Take Home Exercise 2

yang yayong

2025-05-21

# VAST Challenge 2024 Mini-Challenge 1

# Overview

In this exercise, I will address **Question 1** and **Question 2** individually.

1. Design and develop visualizations and visual analytic tools that will allow Silas to explore and understand the profile of Sailor Shift’s career
   1. Who has she been most influenced by over time?
   2. Who has she collaborated with and directly or indirectly influenced?
   3. How has she influenced collaborators of the broader Oceanus Folk community?

## Getting Started

For the purpose of this exercise, four R packages will be used. They are tidyverse, jsonlite, tidygraph and ggraph.

In the code chunk below, p\_load() of **pacman** package is used to load the R packages into R environemnt.

pacman::p\_load(tidyverse, jsonlite,  
 tidygraph, ggraph,ggplot2,SmartEDA,igraph,visNetwork,DiagrammeR)

## Importing Kownledge Graph Data

For the purpose of this exercise, *MC1\_graph.json* file will be used. Before getting started, you should have the data set in the **data** sub-folder.

In the code chunk below, fromJSON() of **jsonlite** package is used to import *MC1\_graph.json* file into R and save the output object

kg <- fromJSON("data/MC1\_graph.json")

What are the components of kg data?

glimpse(kg)

List of 5  
 $ directed : logi TRUE  
 $ multigraph: logi TRUE  
 $ graph :List of 2  
 ..$ node\_default: Named list()  
 ..$ edge\_default: Named list()  
 $ nodes :'data.frame': 17412 obs. of 10 variables:  
 ..$ Node Type : chr [1:17412] "Song" "Person" "Person" "Person" ...  
 ..$ name : chr [1:17412] "Breaking These Chains" "Carlos Duffy" "Min Qin" "Xiuying Xie" ...  
 ..$ single : logi [1:17412] TRUE NA NA NA NA FALSE ...  
 ..$ release\_date : chr [1:17412] "2017" NA NA NA ...  
 ..$ genre : chr [1:17412] "Oceanus Folk" NA NA NA ...  
 ..$ notable : logi [1:17412] TRUE NA NA NA NA TRUE ...  
 ..$ id : int [1:17412] 0 1 2 3 4 5 6 7 8 9 ...  
 ..$ written\_date : chr [1:17412] NA NA NA NA ...  
 ..$ stage\_name : chr [1:17412] NA NA NA NA ...  
 ..$ notoriety\_date: chr [1:17412] NA NA NA NA ...  
 $ links :'data.frame': 37857 obs. of 4 variables:  
 ..$ Edge Type: chr [1:37857] "InterpolatesFrom" "RecordedBy" "PerformerOf" "ComposerOf" ...  
 ..$ source : int [1:37857] 0 0 1 1 2 2 3 5 5 5 ...  
 ..$ target : int [1:37857] 1841 4 0 16180 0 16180 0 5088 14332 11677 ...  
 ..$ key : int [1:37857] 0 0 0 0 0 0 0 0 0 0 ...

The result shows a typical **knowledge graph (KG)** data structure, the detail explanation is as followed:

1.The overall structure is a list of 5 elements:

| Element Name | Meaning |
| --- | --- |
| directed: TRUE | Indicates the graph is **directed** (i.e., relationships have direction, like “someone performed a song”) |
| multigraph: TRUE | Indicates this is a **multigraph**, meaning multiple edges (relationships) can exist between the same pair of nodes (e.g., a person can be both “composer” and “performer”) |
| graph | Contains default attributes for nodes and edges (currently empty named lists) |
| nodes | A data.frame, each row represents a **node** (e.g., a person or a song) |
| links | A data.frame, each row represents an **edge** (i.e., a relationship between two nodes) |

2.nodes table (node metadata)

There are **17,412 nodes**, with the following variables:

| Column Name | Example / Meaning |
| --- | --- |
| Node Type | Node type, such as "Song" or "Person" |
| name | Name of the node (e.g., a person or song title) |
| single | Whether it is a single (TRUE / FALSE) |
| release\_date | Release date (e.g., "2017") |
| genre | Genre (e.g., "Oceanus Folk") |
| notable | Whether it’s notable (TRUE / FALSE) |
| id | Unique ID for the node |
| written\_date | Date the song was written |
| stage\_name | Stage name (if any) |
| notoriety\_date | Date the artist became known (if any) |

3.links table (edge/relationship information)

There are **37,857 edges**, with the following columns:

| Column Name | Meaning / Example |
| --- | --- |
| Edge Type | Type of relationship (e.g., "PerformerOf", "ComposerOf", "InterpolateFrom") |
| source | ID of the source node |
| target | ID of the target node |
| key | Differentiates multiple edges between the same nodes |

**Inspect structure**

Before preparing the data, it is always a good practice to examine the structure of *kg* object.

In the code chunk below str() is used to reveal the structure of *kg* object.

str(kg, max.level = 1)

List of 5  
 $ directed : logi TRUE  
 $ multigraph: logi TRUE  
 $ graph :List of 2  
 $ nodes :'data.frame': 17412 obs. of 10 variables:  
 $ links :'data.frame': 37857 obs. of 4 variables:

## Extracting the edges and nodes tables

Next, as\_tibble() of **tibble** package package is used to **extract the nodes and links tibble data frames** from *kg* object into two separate tibble data frames called *nodes\_tbl* and *edges\_tbl* respectively.

nodes\_tbl <- as\_tibble(kg$nodes)  
edges\_tbl <- as\_tibble(kg$links)

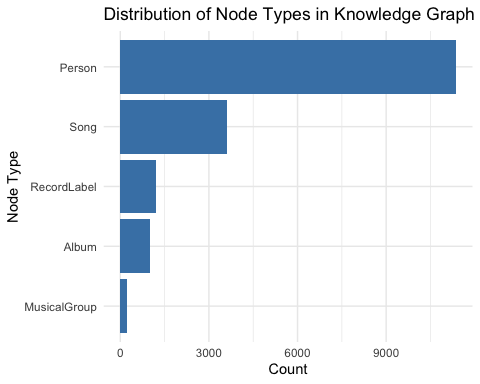
### Initial EDA

It is time for us to apply appropriate EDA methods to examine the data.

In this code chunk below, ggplot2 functions are used the reveal the frequency distribution of *Edge Type* field of *edges\_tbl*.

## Nodes

nodes\_tbl %>%  
 count(`Node Type`) %>%  
 ggplot(aes(x = n, y = reorder(`Node Type`, n))) +  
 geom\_col(fill = "steelblue") +  
 labs(title = "Distribution of Node Types in Knowledge Graph",  
 x = "Count", y = "Node Type") +  
 theme\_minimal()



**Most nodes are of type Person**

* **Explanation**: Individuals (singers, composers, producers, etc.) form the majority of nodes.
* **Insight**: The knowledge graph is **person-centric**, emphasizing individual contributions in the music ecosystem.

**Song is the second most frequent node type**

* **Explanation**: Songs are heavily represented, as expected in a music-related knowledge graph.
* **Insight**: This reflects the graph’s role in **mapping who is connected to which songs**, enabling analysis of performance, authorship, and influence.

**Other node types (e.g., RecordLabel, Album, MusicalGroup) are far fewer**

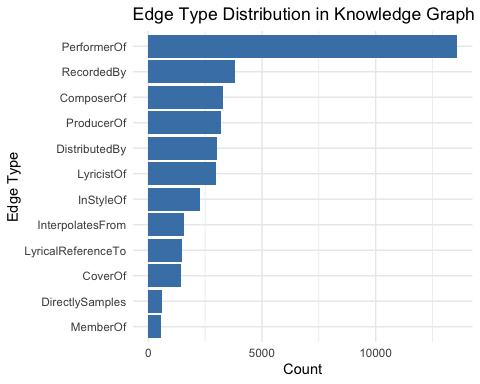
* **Explanation**: Entities like music labels, albums, and groups are present but much less prominent.
* **Insight**: These nodes play a **supporting role** in contextualizing the people-song relationships, potentially useful for advanced analyses (e.g., influence of labels, collaborations within groups).

**MusicalGroup nodes are very rare**

* **Insight**: Either the graph focuses more on individual artists rather than bands/groups, or group membership data may be under-represented.

## Edges

edges\_tbl %>%  
 count(`Edge Type`) %>%  
 ggplot(aes(x = n, y = reorder(`Edge Type`, n))) +  
 geom\_bar(stat = "identity", fill = "steelblue") +  
 labs(title = "Edge Type Distribution in Knowledge Graph",  
 x = "Count", y = "Edge Type") +  
 theme\_minimal()



### **“PerformerOf” is the most frequent edge type**

* **Explanation:** The majority of edges connect a **person** to a **song** via a performance relationship.
* **Insight:** This indicates that the relationship between performers and songs is a core structure in the knowledge graph, showing that the graph places a strong emphasis on **who performed what**.

**“RecordedBy”, “ComposerOf”, and “ProducerOf” are also common**

* **Explanation:** Besides performers, the graph also documents many **creators** of the music, including **composers, producers**, and **recording personnel**.
* **Insight:** This suggests the graph goes beyond surface-level links, capturing multiple roles across the **music creation pipeline**.

**Edge types like “DirectlySamples”, “CoverOf”, “MemberOf”, and “LyricalReferenceTo” are less frequent**

* **Explanation:** These edges represent more complex or less common musical relationships, such as:
* **DirectlySamples**: relates to copyright and musical inspiration
* **CoverOf**: shows the propagation of classic works through cover versions
* **MemberOf**: describes group or band membership
* **Insight:** Although less common, these edge types may hold higher value in studying **musical influence** and **cultural transmission**.

On the other hands, code chunk below uses ggplot2 functions to reveal the frequency distribution of *Node Type* field of *nodes\_tbl*.

## Creating Knowledge Graph

### Mapping from node id to row index

Before we can go ahead to build the tidygraph object, it is important for us to ensures each id from the node list is mapped to the correct row number. This requirement can be achive by using the code chunk below.

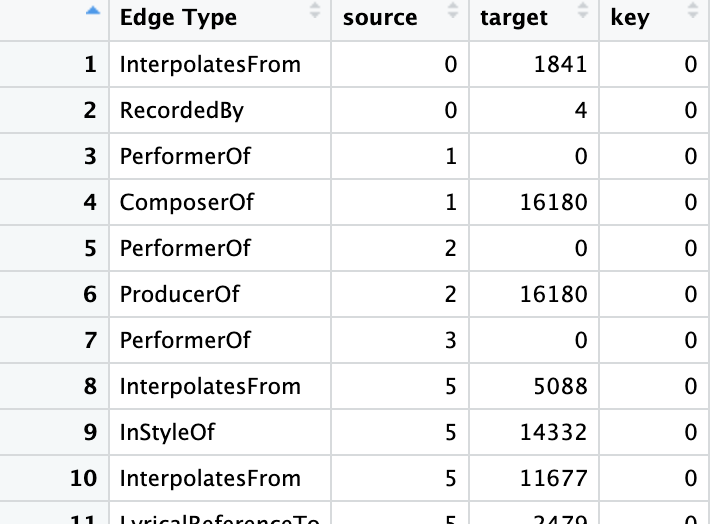
id\_map <- tibble(id = nodes\_tbl$id, #Retrieve the ID column of each row node  
 index = seq\_len(  
 nrow(nodes\_tbl))) #Generate a line number sequence from 1 to n

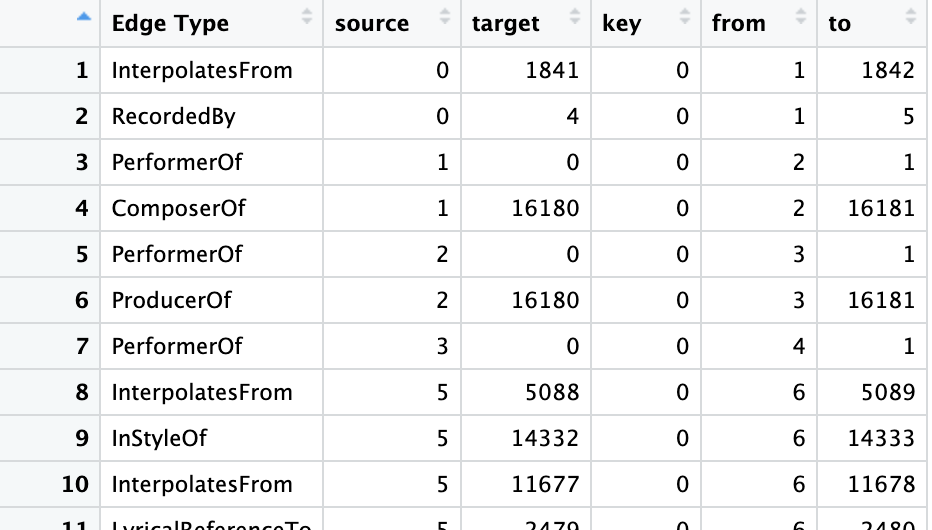
### Map source and target IDs to row indices

Next, we will map the source and the target IDs to row indices by using the code chunk below.

edges\_tbl <- edges\_tbl %>%  
 left\_join(id\_map, by = c("source" = "id")) %>% # source id → from index  
 rename(from = index) %>%   
 left\_join(id\_map, by = c("target" = "id")) %>% # target id → to index  
 rename(to = index)

|  |
| --- |
| Note |
| To better understand the changes before and after the process, it is to take a screenshot of *edges\_tbl* before and after this process and examine the differences. |





Now we can see the difference between before and after the *edges\_tbl.*

table(edges\_tbl$`Edge Type`)

ComposerOf CoverOf DirectlySamples DistributedBy   
 3290 1429 619 3013   
 InStyleOf InterpolatesFrom LyricalReferenceTo LyricistOf   
 2289 1574 1496 2985   
 MemberOf PerformerOf ProducerOf RecordedBy   
 568 13587 3209 3798

### Filter out any unmatched (invalid) edges

Lastly, the code chunk below will be used to exclude the unmatch edges.

edges\_tbl <- edges\_tbl %>%  
 filter(!is.na(from), !is.na(to))

### Creating tidygraph

Lastly, tbl\_graph() is used to create tidygraph’s graph object by using the code chunk below.

graph <- tbl\_graph(nodes = nodes\_tbl,   
 edges = edges\_tbl,   
 directed = kg$directed)

graph

# A tbl\_graph: 17412 nodes and 37857 edges  
#  
# A directed multigraph with 16 components  
#  
# Node Data: 17,412 × 10 (active)  
 `Node Type` name single release\_date genre notable id written\_date  
 <chr> <chr> <lgl> <chr> <chr> <lgl> <int> <chr>   
 1 Song Breaking Th… TRUE 2017 Ocea… TRUE 0 <NA>   
 2 Person Carlos Duffy NA <NA> <NA> NA 1 <NA>   
 3 Person Min Qin NA <NA> <NA> NA 2 <NA>   
 4 Person Xiuying Xie NA <NA> <NA> NA 3 <NA>   
 5 RecordLabel Nautical Mi… NA <NA> <NA> NA 4 <NA>   
 6 Song Unshackled … FALSE 2026 Lo-F… TRUE 5 <NA>   
 7 Person Luke Payne NA <NA> <NA> NA 6 <NA>   
 8 Person Xiulan Zeng NA <NA> <NA> NA 7 <NA>   
 9 Person David Frank… NA <NA> <NA> NA 8 <NA>   
10 RecordLabel Colline-Cas… NA <NA> <NA> NA 9 <NA>   
# ℹ 17,402 more rows  
# ℹ 2 more variables: stage\_name <chr>, notoriety\_date <chr>  
#  
# Edge Data: 37,857 × 6  
 from to `Edge Type` source target key  
 <int> <int> <chr> <int> <int> <int>  
1 1 1842 InterpolatesFrom 0 1841 0  
2 1 5 RecordedBy 0 4 0  
3 2 1 PerformerOf 1 0 0  
# ℹ 37,854 more rows

we might want to confirm the output object is indeed in tidygraph format by using the code chunk below.

class(graph)

[1] "tbl\_graph" "igraph"

## Visualising the knowledge graph

In this section, we will use ggraph’s functions to visualise and analyse the graph object.

|  |
| --- |
| Warning |
| The two examples below are not model answers, they are examples to show you how to use the mantra of **Overview first, details on demand** of visual investigation. |

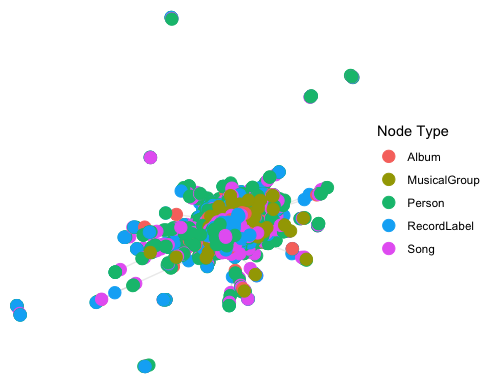
Several of the ggraph layouts involve randomisation. In order to ensure reproducibility, it is necessary to set the seed value before plotting by using the code chunk below.

set.seed(1234)

### **Visualising the whole graph**

In the code chunk below, ggraph functions are used to visualise the whole graph.

ggraph(graph, layout = "fr") +  
 geom\_edge\_link(alpha = 0.3,   
 colour = "gray") +  
 geom\_node\_point(aes(color = `Node Type`),   
 size = 4) +  
 geom\_node\_text(aes(label = name),   
 repel = TRUE,   
 size = 2.5) +  
 theme\_void()



Notice that the whole graph is very messy and we can hardy discover any useful patterns. This is always the case in graph visualisation and analysis. In order to gain meaningful visual discovery, it is always useful for us to looking into the details, for example by plotting sub-graphs.

### **Visualising the sub-graph**

In this section, we are interested to create a sub-graph base on *MemberOf* value in *Edge Type* column of the *edges* data frame.

#### **Filtering edges to only “MemberOf”**

graph\_memberof <- graph %>%  
 activate(edges) %>% ##  
 filter(`Edge Type` == "MemberOf")

#### **Extracting only connected nodes (i.e., used in these edges)**

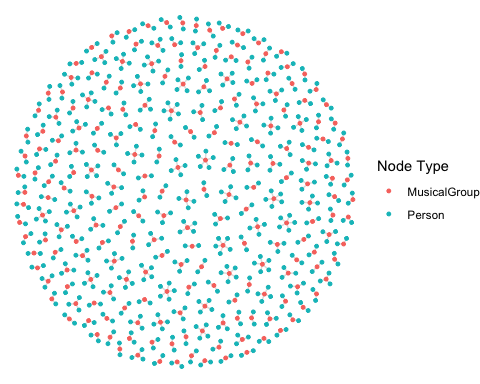
used\_node\_indices <- graph\_memberof %>%  
 activate(edges) %>%  
 as\_tibble() %>%  
 select(from, to) %>%  
 unlist() %>%  
 unique()

#### **Keeping only those nodes**

graph\_memberof <- graph\_memberof %>%  
 activate(nodes) %>%  
 mutate(row\_id = row\_number()) %>%  
 filter(row\_id %in% used\_node\_indices) %>%  
 select(-row\_id) # optional cleanup

#### **Plotting the sub-graph**

ggraph(graph\_memberof,   
 layout = "fr") +  
 geom\_edge\_link(alpha = 0.5,   
 colour = "gray") +  
 geom\_node\_point(aes(color = `Node Type`),   
 size = 1) +  
 geom\_node\_text(aes(label = name),   
 repel = TRUE,   
 size = 2.5) +  
 theme\_void()



Notice that the sub-graph above is very clear and the relationship between musical group and person can be visualise easily.

## Proceed to answer the questions for Mini-Challenge 1

Firstly, I created an extract\_subnetwork function to simplify the process of extracting network data. The methodology was referenced from a [senior’s work.](https://isss608-kjcpaas.netlify.app/take-home_exs/ex3/take-home_ex3#mini-challenge-3) In this case, I will also add a direction parameter to allow for further filtering.

extract\_subnetwork <- function(graph, node\_name,   
 distance = NULL,   
 direction = c("all", "in", "out"),  
 edge\_types = NULL,  
 node\_types = NULL) {  
 direction <- match.arg(direction)  
 node <- which(V(graph)$name == node\_name)  
 if (length(node) == 0) stop("Node name not found in graph.")  
 distance <- ifelse(is.null(distance), length(graph), distance)  
  
 mode <- switch(direction,  
 all = "all",  
 `in` = "in",  
 out = "out")  
  
 igraph\_subgraph <- induced\_subgraph(graph, vids = ego(graph, node, order = distance, mode = mode)[[1]])  
  
 nodes\_df <- igraph::as\_data\_frame(igraph\_subgraph, what = "vertices")  
 edges\_df <- igraph::as\_data\_frame(igraph\_subgraph, what = "edges")  
  
 if (!is.null(edge\_types)) {  
 edges\_df <- edges\_df %>% dplyr::filter(`Edge Type` %in% edge\_types)  
 }  
  
 if (!is.null(node\_types)) {  
 nodes\_df <- nodes\_df %>% dplyr::filter(`Node Type` %in% node\_types)  
 }  
  
 used\_node\_ids <- unique(c(edges\_df$from, edges\_df$to))  
 nodes\_df <- nodes\_df %>% dplyr::filter(name %in% used\_node\_ids)  
  
 tidygraph::tbl\_graph(nodes = nodes\_df, edges = edges\_df, directed = igraph::is\_directed(graph))  
}

**Mini-Challenge 1 consists of three questions. Following a group discussion, I selected Question 1, and the steps I took to address the three sub-questions are described below.**

1. **Who has she been most influenced by over time?**

To address the first question, we considered both direct and indirect influences on Sailor Shift, as well as the temporal dynamics of these influences. The process was divided into three main steps:

**1.1** Identify the individuals who directly influenced her.

**1.2** Examine the works created by Sailor Shift, and trace any indirect influences on these works from others.

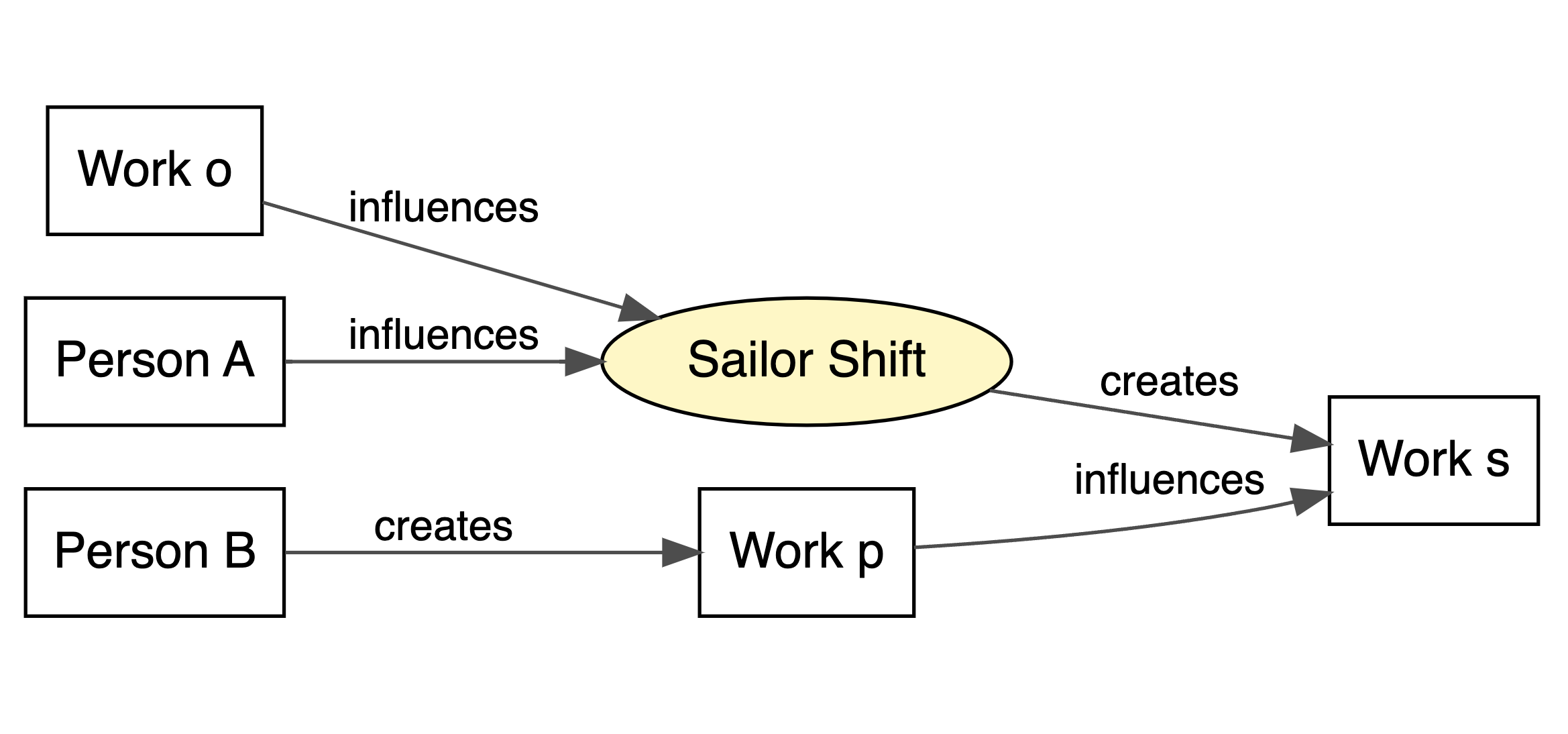
**1.3** Apply a timeline to analyze how these influences evolved over time and observe any trends in their impact.

Sailor Shift <– Person

Sailor Shift –> work <– Person

Sailor Shift –> work\_s <– work\_p <– Person

grViz("  
digraph complex\_influence {  
 graph [layout = dot, rankdir = LR, splines = true]  
  
 node [fontname = Helvetica, style = filled, fillcolor = \"white\", shape = box]  
 edge [style = italic, fontname = Helvetica, fontsize = 12, labeldistance = 1.2, labelloc = c, color = gray30]  
  
  
 A [label = 'Person A']  
 B [label = 'Person B']  
 SS [label = 'Sailor Shift', shape = ellipse, fillcolor = \"#fff7c0\"]  
 W [label = 'Work s']  
 W1 [label = 'Work p']  
 WO [label = 'Work o']   
  
 A -> SS [label = 'influences']  
 SS -> W [label = 'creates']  
 B -> W1 [label = 'creates']  
 W1 -> W [label = 'influences']  
 WO -> SS [label = 'influences']   
}  
")



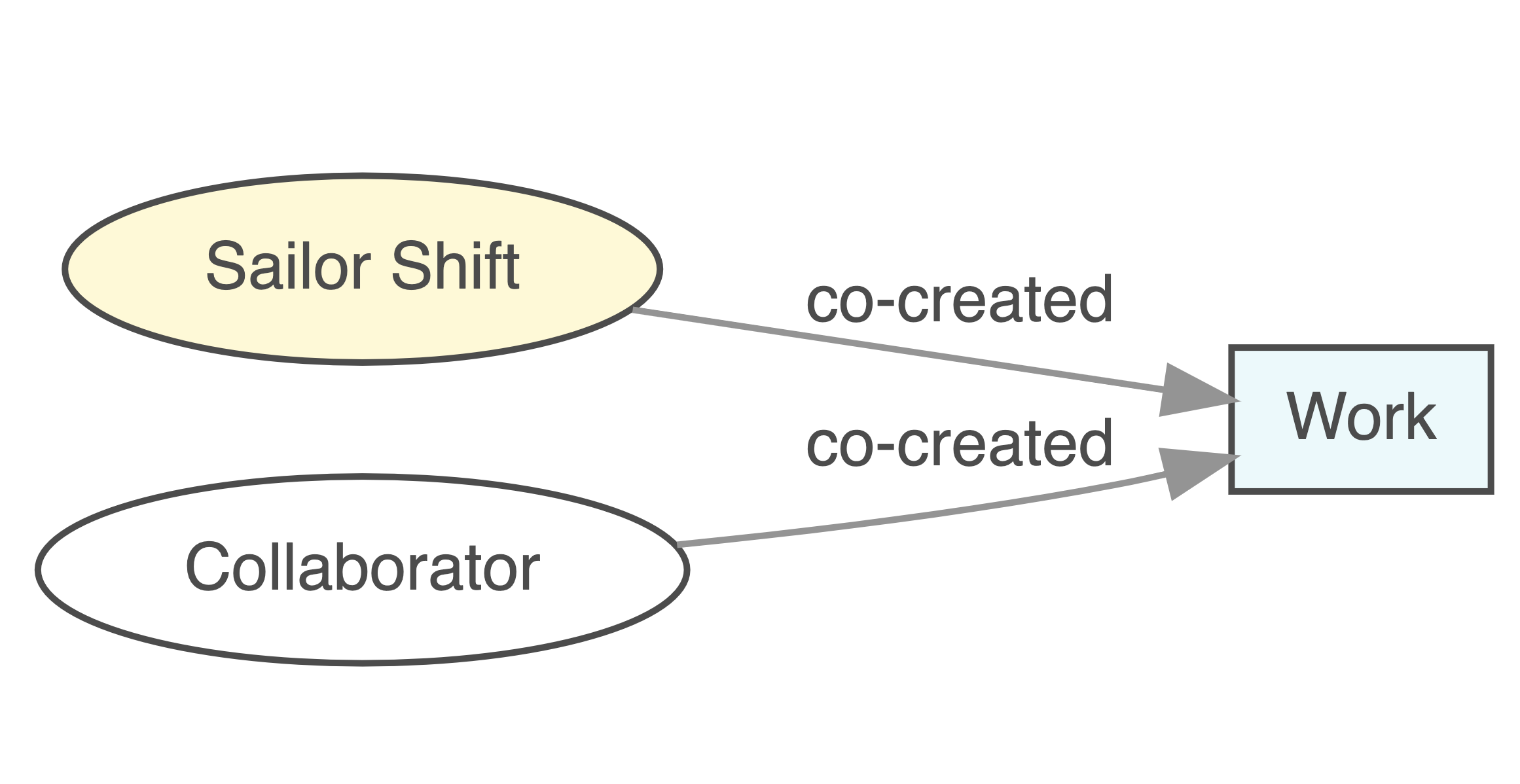
1. **Who has she collaborated with and directly or indirectly influenced?**

To address the second question, we examined both direct and indirect influences involving Sailor Shift, placing her at the center of the analysis.The process was divided into three main steps:

**2.1** Identify who she has collaborated with

Sailor Shift–>work <– Person

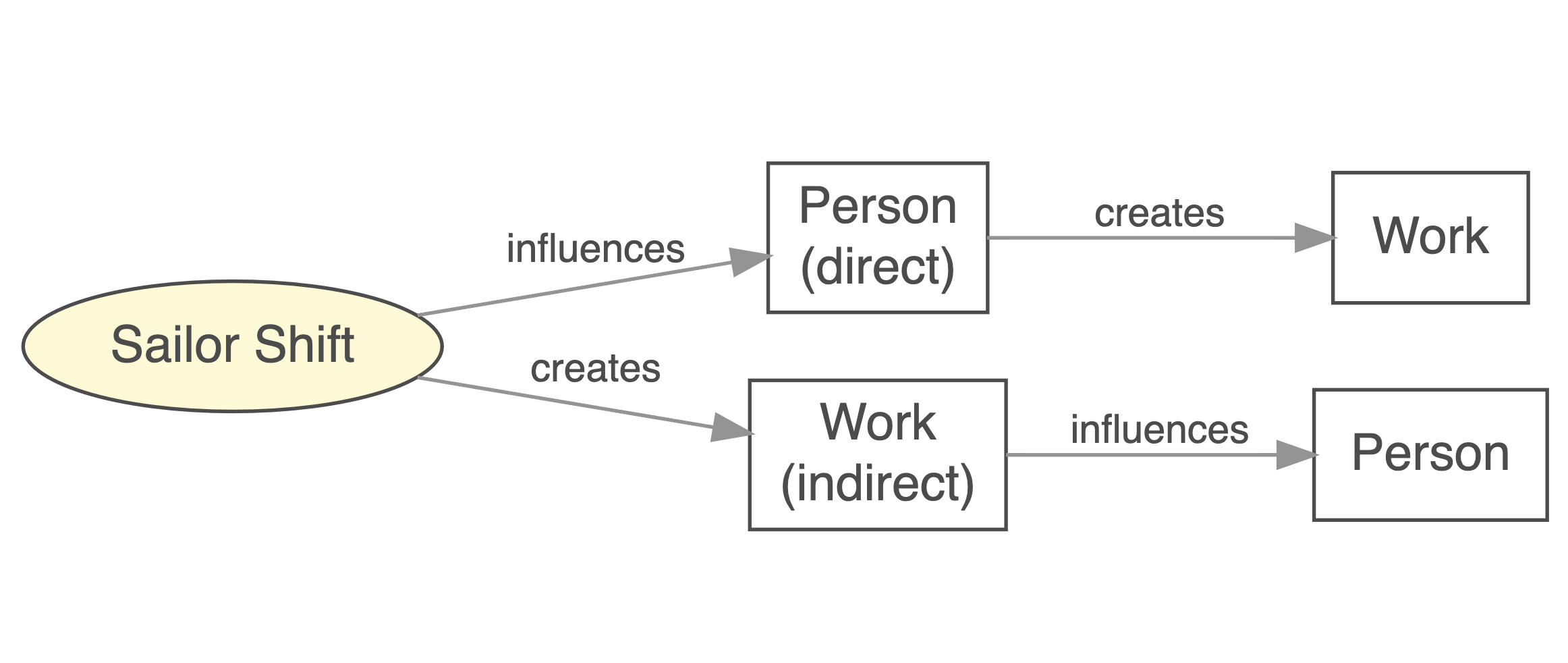
grViz("  
digraph sailor\_collab {  
 graph [layout = dot, rankdir = LR]  
  
 node [fontname = Helvetica, style = filled, fillcolor = \"#ffffff\", shape = box,fontsize = 10, width = 0.5, height = 0.3]  
 edge [fontname = Helvetica, fontsize = 10, color = gray40]  
  
 Sailor [label = 'Sailor Shift', shape = ellipse, fillcolor = '#fff7c0']  
 Work [label = 'Work', shape = box, fillcolor = '#e0f7fa']  
 Person [label = 'Collaborator', shape = ellipse]  
  
 Sailor -> Work [label = 'co-created']  
 Person -> Work [label = 'co-created']  
}  
")

  
**2.2** Determine who she has directly or indirectly influenced

Sailor Shift –> Person –> work (direct)

Sailor Shift–> work –> Person (indirect)

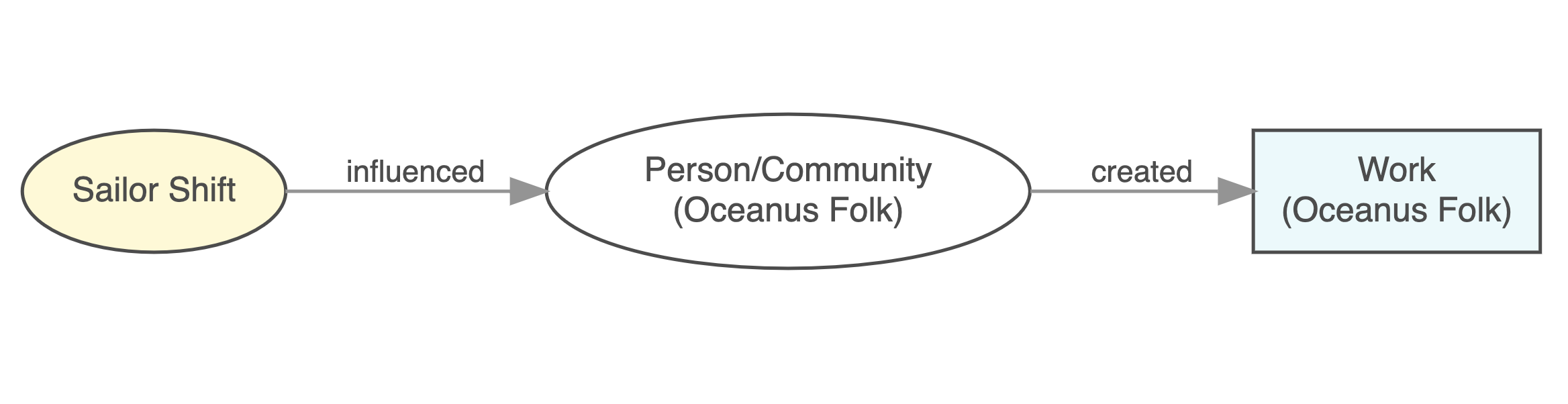
grViz("  
digraph sailor\_influence {  
 graph [layout = dot, rankdir = LR]  
  
 node [fontname = Helvetica, style = filled, fillcolor = \"#ffffff\", shape = box]  
 edge [fontname = Helvetica, fontsize = 11, labelfontsize = 11, color = gray40]  
  
 SS [label = 'Sailor Shift', shape = ellipse, fillcolor = '#fff7c0']  
 P1 [label = 'Person\n(direct)']  
 W1 [label = 'Work']  
 W2 [label = 'Work\n(indirect)']  
 P2 [label = 'Person ']  
  
 SS -> P1 [label = 'influences']  
 P1 -> W1 [label = 'creates']  
  
 SS -> W2 [label = 'creates']  
 W2 -> P2 [label = 'influences']  
}  
")



**3.How has she influenced collaborators of the broader Oceanus Folk community?**

Sailor Shift–> Person/community(Oceanus Folk) –> Work(Oceanus Folk)

grViz("  
digraph oceanus\_influence {  
 graph [layout = dot, rankdir = LR]  
  
 node [fontname = Helvetica, style = filled, fillcolor = \"#ffffff\", shape = box, fontsize = 10]  
 edge [fontname = Helvetica, fontsize = 9, color = gray40]  
  
 Sailor [label = 'Sailor Shift', shape = ellipse, fillcolor = '#fff7c0']  
 Community [label = 'Person/Community\\n(Oceanus Folk)', shape = ellipse]  
 Work [label = 'Work\\n(Oceanus Folk)', shape = box, fillcolor = '#e0f7fa']  
  
 Sailor -> Community [label = 'influenced']  
 Community -> Work [label = 'created']  
}  
")



|  |
| --- |
| Caution |
| It is important to be especially careful with the direction of relationships such as **“PerformerOf,” “ComposerOf,” “LyricistOf,” “ProducerOf,”** as well as **“DirectlySamples,” “InStyleOf,” “CoverOf,” “InterpolatesFrom,”** and **“LyricalReferenceTo.”**  Special attention must be paid to accurately identifying the **source** and **target** of each edge to preserve the correct semantic meaning. |

## Let’s begin.

### **1.Who has she been most influenced by over time?**

The following is the thinking process:

Sailor Shift <– Person

Sailor Shift–> work <– Person

Sailor Shift–> work\_s <– work\_p <– Person

From the initial (though imperfect) visNetwork visualization, we first examined what she had produced, and then proceeded to explore further connections.

subgraph\_in <- extract\_subnetwork(  
 graph,   
 node\_name = "Sailor Shift",   
 distance = 5,   
 direction = "out"  
)  
  
subgraph\_igraph <- as.igraph(subgraph\_in)  
  
nodes\_all <- igraph::as\_data\_frame(subgraph\_igraph, what = "vertices")  
edges\_all <- igraph::as\_data\_frame(subgraph\_igraph, what = "edges")  
  
sailor\_id <- nodes\_all$name[nodes\_all$name == "Sailor Shift"]  
edges\_1st <- edges\_all %>% filter(from == "Sailor Shift")  
  
valid\_targets <- edges\_1st %>%  
 left\_join(nodes\_all, by = c("to" = "name")) %>%  
 filter(`Node Type` %in% c("MusicalGroup", "Song", "Album","RecordLabel")) %>%  
 pull(to)  
  
valid\_node\_ids <- unique(c("Sailor Shift", valid\_targets))  
edges\_vn <- edges\_all %>%   
 filter(from %in% valid\_node\_ids | to %in% valid\_node\_ids)  
  
used\_nodes <- unique(c(edges\_vn$from, edges\_vn$to))  
nodes\_vn <- nodes\_all %>%  
 filter(name %in% used\_nodes) %>%  
 mutate(id = name, label = name, group = `Node Type`)  
  
edges\_vn <- edges\_vn %>%  
 mutate(  
 label = `Edge Type`,   
 title = paste0("Edge Type: ", `Edge Type`)   
 )

visNetwork(nodes\_vn, edges\_vn, height = "800px", width = "100%") %>%  
 visEdges(  
 arrows = "to",  
 font = list(  
 size = 12,   
 align = "middle",   
 color = "black"   
 )  
 ) %>%   
 visOptions(  
 highlightNearest = list(enabled = TRUE, degree = 4, hover = TRUE),  
 nodesIdSelection = TRUE,  
 selectedBy = "group"  
 ) %>%  
 visLegend() %>%  
 visLayout(randomSeed = 1234) %>%  
 visInteraction(navigationButtons = TRUE)

#### **First Layer** Exploration

We started with all nodes directly connected from Sailor Shift, and added the type information of each corresponding from-node.

edges\_1st\_full\_named <- edges\_1st %>%  
 left\_join(nodes\_all %>% select(name, `Node Type`) %>%  
 rename(from = name, `From Type` = `Node Type`), by = "from") %>%  
   
 left\_join(nodes\_all %>% select(name, `Node Type`, release\_date, genre, notable, notoriety\_date) %>%  
 rename(to = name, `To Type` = `Node Type`, `To Release` = release\_date,  
 `To Genre` = genre, `To Notable` = notable, `To Notoriety Date` = notoriety\_date),  
 by = "to") %>%  
   
 select(from, `From Type`, to, `To Type`, `Edge Type`, `To Release`, `To Genre`, `To Notable`, `To Notoriety Date`)

knitr::kable(edges\_1st\_full\_named)

| from | From Type | to | To Type | Edge Type | To Release | To Genre | To Notable | To Notoriety Date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sailor Shift | Person | Neon Heartbeat | Album | LyricistOf | 2031 | Synthwave | FALSE | NA |
| Sailor Shift | Person | Ballads for the End of Time | Album | LyricistOf | 2033 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Melancholy Circuitry | Album | LyricistOf | 2033 | Americana | TRUE | NA |
| Sailor Shift | Person | Drifting Between the Stars and the Sea | Album | LyricistOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Artificial Sunsets | Album | LyricistOf | 2035 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Electric Reverie | Album | LyricistOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Ballads for the Low Tide | Album | LyricistOf | 2037 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Fiddle & The Fjord | MusicalGroup | InStyleOf | NA | NA | NA | NA |
| Sailor Shift | Person | Tides of Echos | Album | LyricistOf | 2029 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Hidden Depths | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Ivy Echos | MusicalGroup | DirectlySamples | NA | NA | NA | NA |
| Sailor Shift | Person | Ivy Echos | MusicalGroup | MemberOf | NA | NA | NA | NA |
| Sailor Shift | Person | The Kelp Forest Canticles | Album | LyricistOf | 2024 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Luminescent Tides | Album | LyricistOf | 2025 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Shoreline Sonnets | Album | LyricistOf | 2026 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tidal Pop Waves | Album | LyricistOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tidal Pop Waves | Album | PerformerOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salty Dreams | Album | LyricistOf | 2030 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salty Dreams | Album | PerformerOf | 2030 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Current & The Chord | Album | LyricistOf | 2032 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Current & The Chord | Album | PerformerOf | 2032 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Coral Beats | Album | LyricistOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Coral Beats | Album | PerformerOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tides & Ballads | Album | LyricistOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tides & Ballads | Album | PerformerOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Oceanbound | Album | LyricistOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Oceanbound | Album | PerformerOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Echoes of the Deep | Album | LyricistOf | 2040 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Echoes of the Deep | Album | PerformerOf | 2040 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | High Tide Heartbeat | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Electric Eel Love | Song | PerformerOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Sun-Drenched Daydream | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Chord of the Deep | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Heart of the Habitat | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Reef Rhythm | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Driftwood Lullaby | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Saltwater Hymn | Song | PerformerOf | 2032 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Moon Over the Tide | Song | PerformerOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Into the Current | Song | PerformerOf | 2034 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Barnacle Heart | Song | PerformerOf | 2034 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Fog & Fiddle | Song | PerformerOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Fisherman’s Prayer | Song | PerformerOf | 2036 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Stormsong | Song | PerformerOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salt in My Veins | Song | PerformerOf | 2040 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | The Last Mariner | Song | PerformerOf | 2040 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | The Saltwater Weavers | MusicalGroup | InStyleOf | NA | NA | NA | NA |
| Sailor Shift | Person | Drowned Harbor | MusicalGroup | LyricalReferenceTo | NA | NA | NA | NA |
| Sailor Shift | Person | Tidesworn Ballads | Album | PerformerOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tidesworn Ballads | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Submerged Sonatas | Album | PerformerOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Submerged Sonatas | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Seashell Serenade | Song | PerformerOf | 2030 | Oceanus Folk | TRUE | 2030 |

Below is all the information on the **First** layer, and then remove if there is no other influenced layer

## DirectlySamples #

selected\_types <- c("DirectlySamples")  
  
edges\_selected <- edges\_1st\_full\_named %>%  
 filter(`Edge Type` %in% selected\_types)  
  
knitr::kable(edges\_selected)

| from | From Type | to | To Type | Edge Type | To Release | To Genre | To Notable | To Notoriety Date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sailor Shift | Person | Ivy Echos | MusicalGroup | DirectlySamples | NA | NA | NA | NA |

## InStyleOf #

selected\_types <- c("InStyleOf")  
  
edges\_selected <- edges\_1st\_full\_named %>%  
 filter(`Edge Type` %in% selected\_types)  
  
knitr::kable(edges\_selected)

| from | From Type | to | To Type | Edge Type | To Release | To Genre | To Notable | To Notoriety Date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sailor Shift | Person | The Fiddle & The Fjord | MusicalGroup | InStyleOf | NA | NA | NA | NA |
| Sailor Shift | Person | The Saltwater Weavers | MusicalGroup | InStyleOf | NA | NA | NA | NA |

## LyricalReferenceTo #

selected\_types <- c("LyricalReferenceTo")  
  
edges\_selected\_LyricalReferenceTo <- edges\_1st\_full\_named %>%  
 filter(`Edge Type` %in% selected\_types)  
  
knitr::kable(edges\_selected\_LyricalReferenceTo)

| from | From Type | to | To Type | Edge Type | To Release | To Genre | To Notable | To Notoriety Date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sailor Shift | Person | Drowned Harbor | MusicalGroup | LyricalReferenceTo | NA | NA | NA | NA |

## LyricistOf

selected\_types <- c("LyricistOf")  
  
edges\_selected <- edges\_1st\_full\_named %>%  
 filter(`Edge Type` %in% selected\_types)  
  
knitr::kable(edges\_selected)

| from | From Type | to | To Type | Edge Type | To Release | To Genre | To Notable | To Notoriety Date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sailor Shift | Person | Neon Heartbeat | Album | LyricistOf | 2031 | Synthwave | FALSE | NA |
| Sailor Shift | Person | Ballads for the End of Time | Album | LyricistOf | 2033 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Melancholy Circuitry | Album | LyricistOf | 2033 | Americana | TRUE | NA |
| Sailor Shift | Person | Drifting Between the Stars and the Sea | Album | LyricistOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Artificial Sunsets | Album | LyricistOf | 2035 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Electric Reverie | Album | LyricistOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Ballads for the Low Tide | Album | LyricistOf | 2037 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tides of Echos | Album | LyricistOf | 2029 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Hidden Depths | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Kelp Forest Canticles | Album | LyricistOf | 2024 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Luminescent Tides | Album | LyricistOf | 2025 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Shoreline Sonnets | Album | LyricistOf | 2026 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tidal Pop Waves | Album | LyricistOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salty Dreams | Album | LyricistOf | 2030 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Current & The Chord | Album | LyricistOf | 2032 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Coral Beats | Album | LyricistOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tides & Ballads | Album | LyricistOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Oceanbound | Album | LyricistOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Echoes of the Deep | Album | LyricistOf | 2040 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tidesworn Ballads | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Submerged Sonatas | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |

## MemberOf #

selected\_types <- c("MemberOf")  
  
edges\_selected <- edges\_1st\_full\_named %>%  
 filter(`Edge Type` %in% selected\_types)  
  
knitr::kable(edges\_selected)

| from | From Type | to | To Type | Edge Type | To Release | To Genre | To Notable | To Notoriety Date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sailor Shift | Person | Ivy Echos | MusicalGroup | MemberOf | NA | NA | NA | NA |

## PerformerOf

selected\_types <- c("PerformerOf")  
  
edges\_selected <- edges\_1st\_full\_named %>%  
 filter(`Edge Type` %in% selected\_types)  
  
knitr::kable(edges\_selected)

| from | From Type | to | To Type | Edge Type | To Release | To Genre | To Notable | To Notoriety Date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sailor Shift | Person | Tidal Pop Waves | Album | PerformerOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salty Dreams | Album | PerformerOf | 2030 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Current & The Chord | Album | PerformerOf | 2032 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Coral Beats | Album | PerformerOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tides & Ballads | Album | PerformerOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Oceanbound | Album | PerformerOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Echoes of the Deep | Album | PerformerOf | 2040 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | High Tide Heartbeat | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Electric Eel Love | Song | PerformerOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Sun-Drenched Daydream | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Chord of the Deep | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Heart of the Habitat | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Reef Rhythm | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Driftwood Lullaby | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Saltwater Hymn | Song | PerformerOf | 2032 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Moon Over the Tide | Song | PerformerOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Into the Current | Song | PerformerOf | 2034 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Barnacle Heart | Song | PerformerOf | 2034 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Fog & Fiddle | Song | PerformerOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Fisherman’s Prayer | Song | PerformerOf | 2036 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Stormsong | Song | PerformerOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salt in My Veins | Song | PerformerOf | 2040 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | The Last Mariner | Song | PerformerOf | 2040 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Tidesworn Ballads | Album | PerformerOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Submerged Sonatas | Album | PerformerOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Seashell Serenade | Song | PerformerOf | 2030 | Oceanus Folk | TRUE | 2030 |

Then we can visual all the connect from Sailor:

edges\_1st\_filtered <- edges\_1st\_full\_named %>%  
 filter(`Edge Type` %in% c("LyricistOf", "PerformerOf", "MemberOf"))  
  
knitr::kable(edges\_1st\_filtered)

| from | From Type | to | To Type | Edge Type | To Release | To Genre | To Notable | To Notoriety Date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sailor Shift | Person | Neon Heartbeat | Album | LyricistOf | 2031 | Synthwave | FALSE | NA |
| Sailor Shift | Person | Ballads for the End of Time | Album | LyricistOf | 2033 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Melancholy Circuitry | Album | LyricistOf | 2033 | Americana | TRUE | NA |
| Sailor Shift | Person | Drifting Between the Stars and the Sea | Album | LyricistOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Artificial Sunsets | Album | LyricistOf | 2035 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Electric Reverie | Album | LyricistOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Ballads for the Low Tide | Album | LyricistOf | 2037 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tides of Echos | Album | LyricistOf | 2029 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Hidden Depths | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Ivy Echos | MusicalGroup | MemberOf | NA | NA | NA | NA |
| Sailor Shift | Person | The Kelp Forest Canticles | Album | LyricistOf | 2024 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Luminescent Tides | Album | LyricistOf | 2025 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Shoreline Sonnets | Album | LyricistOf | 2026 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tidal Pop Waves | Album | LyricistOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tidal Pop Waves | Album | PerformerOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salty Dreams | Album | LyricistOf | 2030 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salty Dreams | Album | PerformerOf | 2030 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Current & The Chord | Album | LyricistOf | 2032 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Current & The Chord | Album | PerformerOf | 2032 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Coral Beats | Album | LyricistOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Coral Beats | Album | PerformerOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tides & Ballads | Album | LyricistOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tides & Ballads | Album | PerformerOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Oceanbound | Album | LyricistOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Oceanbound | Album | PerformerOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Echoes of the Deep | Album | LyricistOf | 2040 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Echoes of the Deep | Album | PerformerOf | 2040 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | High Tide Heartbeat | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Electric Eel Love | Song | PerformerOf | 2028 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Sun-Drenched Daydream | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Chord of the Deep | Song | PerformerOf | 2028 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Heart of the Habitat | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Reef Rhythm | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Driftwood Lullaby | Song | PerformerOf | 2030 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Saltwater Hymn | Song | PerformerOf | 2032 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Moon Over the Tide | Song | PerformerOf | 2034 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Into the Current | Song | PerformerOf | 2034 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Barnacle Heart | Song | PerformerOf | 2034 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Fog & Fiddle | Song | PerformerOf | 2036 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | The Fisherman’s Prayer | Song | PerformerOf | 2036 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Stormsong | Song | PerformerOf | 2038 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Salt in My Veins | Song | PerformerOf | 2040 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | The Last Mariner | Song | PerformerOf | 2040 | Oceanus Folk | FALSE | NA |
| Sailor Shift | Person | Tidesworn Ballads | Album | PerformerOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Tidesworn Ballads | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Submerged Sonatas | Album | PerformerOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Submerged Sonatas | Album | LyricistOf | 2031 | Oceanus Folk | TRUE | NA |
| Sailor Shift | Person | Seashell Serenade | Song | PerformerOf | 2030 | Oceanus Folk | TRUE | 2030 |

edge\_colors <- c(  
 "PerformerOf" = "#2ca02c",  
  
 "LyricistOf" = "#1f77b4",  
   
   
 "MemberOf" = "#d62728"  
)  
  
nodes\_subgraph <- nodes\_all %>%  
 filter(name %in% c("Sailor Shift", edges\_1st\_filtered$to)) %>%  
 transmute(id = name, label = name, group = `Node Type`)  
  
edges\_subgraph <- edges\_1st\_filtered %>%  
 left\_join(  
 nodes\_all %>% select(name, `Node Type`) %>% rename(to = name, to\_type = `Node Type`),  
 by = "to"  
 ) %>%  
 mutate(  
 color = edge\_colors[`Edge Type`],  
 title = paste0("Edge Type: ", `Edge Type`, "<br>To Type: ", to\_type)  
 ) %>%  
 transmute(  
 from = from,  
 to = to,  
 title = title,  
 color = color,  
 arrows = "to"  
 )  
  
edge\_legend <- data.frame(  
 label = names(edge\_colors),  
 color = unname(edge\_colors),  
 arrows = rep("to", length(edge\_colors)),  
 stringsAsFactors = FALSE  
)

visNetwork(nodes\_subgraph, edges\_subgraph, width = "100%", height = "700px") %>%  
 visEdges(smooth = TRUE) %>%  
 visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE) %>%  
 visLegend(  
 addEdges = edge\_legend,  
 useGroups = TRUE,  
 position = "right",  
 ) %>%  
 visPhysics(  
 solver = "forceAtlas2Based",  
 forceAtlas2Based = list(gravitationalConstant = -80),  
 stabilization = list(enabled = TRUE, iterations = 100)  
 ) %>%  
 visLayout(randomSeed = 123)

|  |
| --- |
| What we can find from the network？ |
| **She is a highly prolific creator**   * The dense number of outgoing edges indicates that Sailor Shift has contributed to or performed a large number of works, including both songs and albums. * The number of connected nodes shows she is linked to over 20 works, placing her at the center of the network.   **She plays both performer and lyricist roles**  The presence of many green edges (PerformerOf) and blue edges (LyricistOf) suggests:   * She not only performs but also writes lyrics for her works. * She plays a dual role in the music creation process, showing versatility as both a performer and a songwriter.   **She is not an entirely solo artist**  A red edge (MemberOf) connects her to the red node Ivy Echos, indicating:   * She is a member of this musical group. * Some of her creative output may result from group collaboration. * She maintains a balance between solo and group work.   Sailor Shift is a central figure,skilled in both songwriting and performance, and active in both solo and collaborative projects. Her work likely forms a key part of the community’s creative landscape. |

#### Third Layer Exploration

Next, we aim to explore other’s relationship with Ivy Echos.”

ivy\_groups <- edges\_all %>%  
 filter(to == "Ivy Echos", `Edge Type` == "MemberOf") %>%  
 pull(from)  
ivy\_groups

[1] "Sailor Shift"

group\_members <- edges\_all %>%  
 filter(`Edge Type` == "MemberOf", to %in% ivy\_groups) %>%  
 select(from, to) %>%  
 rename(Member = from, Group = to)  
  
print(group\_members)

[1] Member Group   
<0 rows> (or 0-length row.names)

We attempted to identify the *MemberOf* relationships for Ivy Echos, but the results suggest that this information is not included in the current dataset.”

sailor\_works <- edges\_1st\_full\_named %>%  
 filter(`Edge Type` %in% c("ComposerOf", "PerformerOf", "LyricistOf", "ProducerOf","MemberOf"),  
 `To Type` %in% c("Song", "Album","MusicalGroup","Person","RecordLabel")) %>%  
 pull(to) %>%  
 unique()  
  
edges\_others <- edges\_all %>%  
 filter(  
 to %in% sailor\_works,  
 `Edge Type` %in% c("ComposerOf", "PerformerOf", "LyricistOf", "ProducerOf"),  
 from != "Sailor Shift"   
 )  
  
edges\_others\_full <- edges\_others %>%  
 left\_join(nodes\_all, by = c("from" = "name")) %>%  
 rename(Participant = from, ParticipantType = `Node Type`) %>%  
 select(Participant, ParticipantType, to, `Edge Type`)  
  
knitr::kable(edges\_others\_full)

| Participant | ParticipantType | to | Edge Type |
| --- | --- | --- | --- |
| Ivy Echos | MusicalGroup | The Kelp Forest Canticles | PerformerOf |
| Ivy Echos | MusicalGroup | Luminescent Tides | PerformerOf |
| Ivy Echos | MusicalGroup | Shoreline Sonnets | PerformerOf |

From the result we can see, in this dataset,Ivy Echos has perform three songs.These will late add in the Folk Musci Group for further analysis.

For more convenient exploration, we selected the nodes connected to the third layer in order to observe indirect influences

Get the complete information of the from node

Sailor Shift –(PerformerOf etc.)–> Song A –(InStyleOf etc.)–> Song B

nodes\_tbl <- nodes\_tbl %>% mutate(index = row\_number())  
  
sailor\_index <- nodes\_tbl %>%  
 filter(name == "Sailor Shift") %>%  
 pull(index)  
  
edges\_from\_sailor <- edges\_tbl %>%  
 filter(from == sailor\_index)  
  
first\_layer\_info <- edges\_from\_sailor %>%  
 inner\_join(nodes\_tbl, by = c("to" = "index")) %>%  
 filter(`Node Type` %in% c("Song", "Album", "MusicalGroup", "RecordLabel"))  
  
first\_layer\_targets <- first\_layer\_info$to  
  
influence\_edges <- c("InStyleOf", "LyricalReferenceTo", "InterpolatesFrom", "CoverOf", "DirectlySamples")  
  
edges\_2nd <- edges\_tbl %>%  
 filter(from %in% first\_layer\_targets & `Edge Type` %in% influence\_edges)

In order to find more insights in one table, we have added the “from” information to the data frame

edges\_1st <- edges\_all %>%  
 filter(from == "Sailor Shift")  
  
edges\_1st\_full <- edges\_1st %>%  
 left\_join(nodes\_all, by = c("to" = "name")) %>%  
 select(from, to, `Edge Type`, `Node Type`, release\_date,genre,notable,notoriety\_date)  
  
selected\_types <- c("LyricistOf", "PerformerOf")  
edges\_selected <- edges\_1st\_full %>%  
 filter(`Edge Type` %in% selected\_types)  
  
target\_nodes <- unique(edges\_selected$to)  
  
edges\_2nd <- edges\_all %>%  
 filter(from %in% target\_nodes)  
  
from\_info <- nodes\_all %>%  
 select(name, `Node Type`) %>%  
 rename(from = name, `From Node Type` = `Node Type`)  
  
to\_info <- nodes\_all %>%  
 select(name, `Node Type`, release\_date, genre, notable, notoriety\_date) %>%  
 rename(  
 to = name,  
 `To Node Type` = `Node Type`,  
 to\_release\_date = release\_date,  
 to\_genre = genre,  
 to\_notable = notable,  
 to\_notoriety\_date = notoriety\_date  
 )  
  
edges\_2nd\_full <- edges\_2nd %>%  
 left\_join(from\_info, by = "from") %>%  
 left\_join(to\_info, by = "to") %>%  
 select(  
 from, `From Node Type`, to, `To Node Type`, `Edge Type`,  
 to\_release\_date, to\_genre, to\_notable, to\_notoriety\_date  
 )  
  
  
knitr::kable(edges\_2nd\_full)

| from | From Node Type | to | To Node Type | Edge Type | to\_release\_date | to\_genre | to\_notable | to\_notoriety\_date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Melancholy Circuitry | Album | Twilight’s Threshold | Song | CoverOf | 2007 | Synthwave | TRUE | 2007 |
| Electric Reverie | Album | Reflejo Interior | Song | InterpolatesFrom | 1983 | Americana | TRUE | NA |
| Electric Reverie | Album | Folklore’s Heartbeat | Song | InStyleOf | 2020 | Blues Rock | TRUE | 2033 |
| Tides of Echos | Album | The Long Way Home | Album | InStyleOf | 2023 | Dream Pop | TRUE | NA |
| Tides of Echos | Album | Weathered Miles | Song | InterpolatesFrom | 2023 | Doom Metal | TRUE | NA |
| The Kelp Forest Canticles | Album | Hallowed Transaction | Album | DirectlySamples | 2017 | Doom Metal | TRUE | NA |
| The Kelp Forest Canticles | Album | Für uns alle zusammen | Song | CoverOf | 2017 | Alternative Rock | TRUE | NA |
| Salty Dreams | Album | Susurros de Pasión | Song | InStyleOf | 2004 | Synthwave | TRUE | 2004 |
| Salty Dreams | Album | Partisan’s Lament | Song | CoverOf | 2030 | Indie Pop | TRUE | NA |
| Coral Beats | Album | Sacred Fragments | Song | InStyleOf | 2003 | Synthwave | TRUE | 2003 |
| Coral Beats | Album | Harvest Dance at Ler Valley | Song | CoverOf | 2010 | Psychedelic Rock | TRUE | NA |
| Oceanbound | Album | Bold Without Apology | Song | LyricalReferenceTo | 2010 | Indie Rock | TRUE | NA |
| Oceanbound | Album | Altitude of Mistakes | Song | LyricalReferenceTo | 2015 | Americana | TRUE | 2015 |
| High Tide Heartbeat | Song | Addicted to Your Heartache | Album | InterpolatesFrom | 2004 | Southern Gothic Rock | TRUE | NA |
| Electric Eel Love | Song | Twelve Bells of Augsburg | Song | InStyleOf | 2026 | Psychedelic Rock | TRUE | NA |
| Electric Eel Love | Song | The Crown We Wear | Song | CoverOf | 2022 | Indie Rock | TRUE | NA |
| Driftwood Lullaby | Song | Echoes of Forgotten Light | Song | InStyleOf | 2004 | Alternative Rock | TRUE | NA |
| Saltwater Hymn | Song | Parallel Memories | Song | InStyleOf | 2028 | Dream Pop | TRUE | NA |
| Moon Over the Tide | Song | Divergent Memories | Song | InterpolatesFrom | 2017 | Indie Folk | TRUE | 2017 |
| Moon Over the Tide | Song | Silent Steps in the Forest’s Embrace | Song | LyricalReferenceTo | 2023 | Avant-Garde Folk | TRUE | 2023 |
| Barnacle Heart | Song | Our Inevitable Path | Song | InStyleOf | 2017 | Indie Folk | TRUE | NA |
| Fog & Fiddle | Song | Provence Aria | Song | InStyleOf | 2023 | Indie Rock | TRUE | NA |
| Fog & Fiddle | Song | Coastal Whispers of Biscay | Song | InterpolatesFrom | 2031 | Alternative Rock | TRUE | NA |
| Stormsong | Song | When the Bar Lights Go Up | Song | InStyleOf | 2000 | Emo/Pop Punk | TRUE | NA |
| Stormsong | Song | Exoskeleton Dissolve | Song | DirectlySamples | 2020 | Indie Rock | TRUE | 2020 |
| Tidesworn Ballads | Album | Dreamscape of Judgment | Song | CoverOf | 2023 | Symphonic Metal | TRUE | NA |

from\_info <- nodes\_all %>%  
 select(name, `Node Type`, release\_date, genre, notable, notoriety\_date) %>%  
 rename(  
 from = name,  
 `From Node Type` = `Node Type`,  
 from\_release\_date = release\_date,  
 from\_genre = genre,  
 from\_notable = notable,  
 from\_notoriety\_date = notoriety\_date  
 )  
  
to\_info <- nodes\_all %>%  
 select(name, `Node Type`, release\_date, genre, notable, notoriety\_date) %>%  
 rename(  
 to = name,  
 `To Node Type` = `Node Type`,  
 to\_release\_date = release\_date,  
 to\_genre = genre,  
 to\_notable = notable,  
 to\_notoriety\_date = notoriety\_date  
 )  
  
edges\_2nd\_full <- edges\_2nd %>%  
 left\_join(from\_info, by = "from") %>%  
 left\_join(to\_info, by = "to") %>%  
 select(  
 from, `From Node Type`, from\_release\_date, from\_genre, from\_notable, from\_notoriety\_date,  
 to, `To Node Type`, to\_release\_date, to\_genre, to\_notable, to\_notoriety\_date,  
 `Edge Type`  
 )  
  
knitr::kable(edges\_2nd\_full)

| from | From Node Type | from\_release\_date | from\_genre | from\_notable | from\_notoriety\_date | to | To Node Type | to\_release\_date | to\_genre | to\_notable | to\_notoriety\_date | Edge Type |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Melancholy Circuitry | Album | 2033 | Americana | TRUE | NA | Twilight’s Threshold | Song | 2007 | Synthwave | TRUE | 2007 | CoverOf |
| Electric Reverie | Album | 2038 | Oceanus Folk | TRUE | NA | Reflejo Interior | Song | 1983 | Americana | TRUE | NA | InterpolatesFrom |
| Electric Reverie | Album | 2038 | Oceanus Folk | TRUE | NA | Folklore’s Heartbeat | Song | 2020 | Blues Rock | TRUE | 2033 | InStyleOf |
| Tides of Echos | Album | 2029 | Oceanus Folk | TRUE | NA | The Long Way Home | Album | 2023 | Dream Pop | TRUE | NA | InStyleOf |
| Tides of Echos | Album | 2029 | Oceanus Folk | TRUE | NA | Weathered Miles | Song | 2023 | Doom Metal | TRUE | NA | InterpolatesFrom |
| The Kelp Forest Canticles | Album | 2024 | Oceanus Folk | TRUE | NA | Hallowed Transaction | Album | 2017 | Doom Metal | TRUE | NA | DirectlySamples |
| The Kelp Forest Canticles | Album | 2024 | Oceanus Folk | TRUE | NA | Für uns alle zusammen | Song | 2017 | Alternative Rock | TRUE | NA | CoverOf |
| Salty Dreams | Album | 2030 | Oceanus Folk | TRUE | NA | Susurros de Pasión | Song | 2004 | Synthwave | TRUE | 2004 | InStyleOf |
| Salty Dreams | Album | 2030 | Oceanus Folk | TRUE | NA | Partisan’s Lament | Song | 2030 | Indie Pop | TRUE | NA | CoverOf |
| Coral Beats | Album | 2034 | Oceanus Folk | TRUE | NA | Sacred Fragments | Song | 2003 | Synthwave | TRUE | 2003 | InStyleOf |
| Coral Beats | Album | 2034 | Oceanus Folk | TRUE | NA | Harvest Dance at Ler Valley | Song | 2010 | Psychedelic Rock | TRUE | NA | CoverOf |
| Oceanbound | Album | 2038 | Oceanus Folk | TRUE | NA | Bold Without Apology | Song | 2010 | Indie Rock | TRUE | NA | LyricalReferenceTo |
| Oceanbound | Album | 2038 | Oceanus Folk | TRUE | NA | Altitude of Mistakes | Song | 2015 | Americana | TRUE | 2015 | LyricalReferenceTo |
| High Tide Heartbeat | Song | 2028 | Oceanus Folk | FALSE | NA | Addicted to Your Heartache | Album | 2004 | Southern Gothic Rock | TRUE | NA | InterpolatesFrom |
| Electric Eel Love | Song | 2028 | Oceanus Folk | TRUE | NA | Twelve Bells of Augsburg | Song | 2026 | Psychedelic Rock | TRUE | NA | InStyleOf |
| Electric Eel Love | Song | 2028 | Oceanus Folk | TRUE | NA | The Crown We Wear | Song | 2022 | Indie Rock | TRUE | NA | CoverOf |
| Driftwood Lullaby | Song | 2030 | Oceanus Folk | FALSE | NA | Echoes of Forgotten Light | Song | 2004 | Alternative Rock | TRUE | NA | InStyleOf |
| Saltwater Hymn | Song | 2032 | Oceanus Folk | FALSE | NA | Parallel Memories | Song | 2028 | Dream Pop | TRUE | NA | InStyleOf |
| Moon Over the Tide | Song | 2034 | Oceanus Folk | TRUE | NA | Divergent Memories | Song | 2017 | Indie Folk | TRUE | 2017 | InterpolatesFrom |
| Moon Over the Tide | Song | 2034 | Oceanus Folk | TRUE | NA | Silent Steps in the Forest’s Embrace | Song | 2023 | Avant-Garde Folk | TRUE | 2023 | LyricalReferenceTo |
| Barnacle Heart | Song | 2034 | Oceanus Folk | FALSE | NA | Our Inevitable Path | Song | 2017 | Indie Folk | TRUE | NA | InStyleOf |
| Fog & Fiddle | Song | 2036 | Oceanus Folk | TRUE | NA | Provence Aria | Song | 2023 | Indie Rock | TRUE | NA | InStyleOf |
| Fog & Fiddle | Song | 2036 | Oceanus Folk | TRUE | NA | Coastal Whispers of Biscay | Song | 2031 | Alternative Rock | TRUE | NA | InterpolatesFrom |
| Stormsong | Song | 2038 | Oceanus Folk | TRUE | NA | When the Bar Lights Go Up | Song | 2000 | Emo/Pop Punk | TRUE | NA | InStyleOf |
| Stormsong | Song | 2038 | Oceanus Folk | TRUE | NA | Exoskeleton Dissolve | Song | 2020 | Indie Rock | TRUE | 2020 | DirectlySamples |
| Tidesworn Ballads | Album | 2031 | Oceanus Folk | TRUE | NA | Dreamscape of Judgment | Song | 2023 | Symphonic Metal | TRUE | NA | CoverOf |

Now we can observe the next layer — identifying which songs and related works have influenced Sailor Shift. At the same time, we also aim to incorporate the time dimension into this analysis.

#### **Construct edges\_time\_ready data**

edges\_time\_ready <- edges\_2nd\_full %>%  
 mutate(  
 from\_release\_date = as.numeric(from\_release\_date),  
 to\_release\_date = as.numeric(to\_release\_date)  
 ) %>%  
 filter(!is.na(from\_release\_date) & !is.na(to\_release\_date)) %>%  
 mutate(  
 influencer\_point = to,  
 sailor\_point = from,  
 work\_label = paste0(from, " (Sailor)"),  
 y\_axis\_label = paste0(from, " (Sailor)")   
 ) %>%  
 tidyr::pivot\_longer(  
 cols = c(to\_release\_date, from\_release\_date),  
 names\_to = "timepoint",  
 values\_to = "year"  
 ) %>%  
 mutate(  
 point\_type = ifelse(timepoint == "from\_release\_date", "Sailor", "Influencer"),  
 label = ifelse(point\_type == "Sailor", from, to),  
 label\_display = ifelse(point\_type == "Sailor", paste0(label, " (Sailor)"), label),  
 genre = ifelse(point\_type == "Sailor", from\_genre, to\_genre),  
 notable = ifelse(point\_type == "Sailor", from\_notable, to\_notable)  
 )

p <- ggplot(edges\_time\_ready, aes(x = year, y = y\_axis\_label, group = paste(from, to))) +  
 geom\_line(  
 aes(text = paste("Influence:", to, "→", from)),  
 color = "grey40", linewidth = 0.4, alpha = 0.6  
 ) +  
 geom\_point(  
 aes(  
 text = paste0(  
 "Year: ", year,  
 "<br>Work: ", label\_display,  
 "<br>Type: ", point\_type,  
 "<br>Genre: ", genre,  
 "<br>Notable: ", ifelse(notable == TRUE, "Yes", "No"),  
 "<br>Edge Type: ", `Edge Type`  
 ),  
 color = point\_type  
 ),  
 size = 1.8, alpha = 0.9  
 ) +  
 scale\_color\_manual(values = c("Influencer" = "#1F78B4", "Sailor" = "#E41A1C")) +  
 labs(  
 title = "Influence Timeline of Sailor Shift's Works",  
 subtitle = "Lines point to Sailor's works (fixed y-axis)",  
 x = "Release Year", y = "Sailor's Work", color = "Point Type"  
 ) +  
 scale\_x\_continuous(breaks = pretty(edges\_time\_ready$year, n = 10)) +  
 theme\_minimal() +  
 theme(  
 axis.text.y = element\_text(size = 7),  
 plot.title = element\_text(hjust = 0, face = "bold", size = 16),  
 panel.grid.minor = element\_blank(),  
 legend.position = "bottom",  
   
 )  
  
interactive\_plot <- plotly::ggplotly(p, tooltip = "text")  
interactive\_plot

|  |
| --- |
| When was she influenced, by whom, and in what way? |
| Timing and Type of Influence   * Sailor Shift’s compositions were inspired by a wide temporal range of influencer works from the 1980s to the 2020s. * Notable classics that shaped her music include: * Reflejo Interior (1983, Americana) → Influenced *Electric Reverie* * Twilight’s Threshold (2007, Synthwave) → Influenced *Tides of Echos* * Hallowed Transaction (2017, Doom Metal) and *Divergent Memories (2017, Indie Folk)* → Contributed to her emerging Indie and Metal sounds * From a genre perspective, Sailor draws from Synthwave, Doom Metal, Dream Pop, Indie Rock, and Psychedelic Rock, blending them into her distinctive Oceanus Folk style.   Time Distribution & Creative Rhythm   * 2015–2020 marks the *starting phase of stylistic absorption*. * Songs like *Melancholy Circuitry* and *Stormsong* show early influence connections — indicating the onset of external musical borrowing. * 2023–2035 is her most active creative and stylistic integration period. * The visualization shows a dense cluster of red points (Sailor’s works) and blue points (influencer works), suggesting heavy external inspiration and high creative output during this time.   Creative Maturity vs. Density of Inspiration   * Songs such as: * *Tidesworn Ballads*, *Coral Beats*, *Driftwood Lullaby*, *Electric Reverie* * are influenced by multiple sources — identifying them as “convergent inspiration” pieces reflecting diverse stylistic integration. * Data also reveals: * Most of Sailor’s songs are of the Oceanus Folk genre, while the influencers span Synthwave, Southern Gothic, Indie Pop, Symphonic Metal, and more. * Her most notable works (marked as Notable = TRUE) tend to be those that were heavily influenced — indicating that *inspiration led to impact*. |

We analyze the ‘Other Person → Other Work → Sailor’ connections that contribute to the influence on Sailor Shift’s second-layer works.

nodes\_tbl <- nodes\_tbl %>% mutate(index = row\_number())  
  
sailor\_index <- nodes\_tbl %>%  
 filter(name == "Sailor Shift") %>%  
 pull(index)  
  
edges\_from\_sailor <- edges\_tbl %>%  
 filter(from == sailor\_index)  
  
first\_layer\_info <- edges\_from\_sailor %>%  
 inner\_join(nodes\_tbl, by = c("to" = "index")) %>%  
 filter(`Node Type` %in% c("Song", "Album", "MusicalGroup", "RecordLabel"))  
  
first\_layer\_targets <- first\_layer\_info$to  
  
influence\_edges <- c("InStyleOf", "LyricalReferenceTo", "InterpolatesFrom", "CoverOf", "DirectlySamples")  
  
edges\_2nd <- edges\_tbl %>%  
 filter(from %in% first\_layer\_targets & `Edge Type` %in% influence\_edges)  
  
influenced\_works <- edges\_2nd$to  
songs\_with\_outgoing <- unique(edges\_2nd$from)  
  
person\_edge\_types <- c("PerformerOf", "ComposerOf", "LyricistOf", "ProducerOf")  
  
edges\_people\_to\_2nd <- edges\_tbl %>%  
 filter(  
 to %in% influenced\_works,  
 `Edge Type` %in% person\_edge\_types  
 ) %>%  
 left\_join(nodes\_tbl %>% select(index, `Node Type`), by = c("from" = "index")) %>%  
 rename(`From Node Type` = `Node Type`) %>%  
 filter(`From Node Type` %in% c("Person", "RecordLabel")) %>%  
 left\_join(nodes\_tbl %>% select(index, `Node Type`, release\_date), by = c("to" = "index")) %>%  
 rename(`To Node Type` = `Node Type`, release\_date = release\_date)

#### Indirectly influenced Sailor work and Sailor

Here we can clearly see ,who has indirectly influenced Sailor work and Sailor.

edges\_people\_to\_2nd\_tbl <- edges\_people\_to\_2nd %>%  
 left\_join(nodes\_tbl %>% select(index, name), by = c("from" = "index")) %>%  
 rename(`Person Name` = name) %>%  
 left\_join(nodes\_tbl %>% select(index, name), by = c("to" = "index")) %>%  
 rename(`Work Name` = name) %>%  
 select(`Person Name`, `Work Name`, `Edge Type`, `From Node Type`, `To Node Type`, release\_date)  
  
knitr::kable(edges\_people\_to\_2nd\_tbl)

| Person Name | Work Name | Edge Type | From Node Type | To Node Type | release\_date |
| --- | --- | --- | --- | --- | --- |
| Ming Ren | Provence Aria | PerformerOf | Person | Song | 2023 |
| Belinda Knappe | Harvest Dance at Ler Valley | ProducerOf | Person | Song | 2010 |
| Yang Peng | Sacred Fragments | PerformerOf | Person | Song | 2003 |
| Yang Peng | Sacred Fragments | ProducerOf | Person | Song | 2003 |
| Min Cao | Sacred Fragments | PerformerOf | Person | Song | 2003 |
| Qiang Zhou | Bold Without Apology | ComposerOf | Person | Song | 2010 |
| Jing Kang | Bold Without Apology | PerformerOf | Person | Song | 2010 |
| Jing Kang | Bold Without Apology | ProducerOf | Person | Song | 2010 |
| Jun Zhou | Susurros de Pasión | PerformerOf | Person | Song | 2004 |
| Yong Dong | Altitude of Mistakes | PerformerOf | Person | Song | 2015 |
| Urszula Stochmal | Addicted to Your Heartache | PerformerOf | Person | Album | 2004 |
| Urszula Stochmal | Addicted to Your Heartache | LyricistOf | Person | Album | 2004 |
| Urszula Stochmal | Addicted to Your Heartache | ComposerOf | Person | Album | 2004 |
| Ming Long | Twilight’s Threshold | PerformerOf | Person | Song | 2007 |
| Ming Long | Twilight’s Threshold | ProducerOf | Person | Song | 2007 |
| Yong Lu | Twilight’s Threshold | LyricistOf | Person | Song | 2007 |
| Yang Wan | Sacred Fragments | ComposerOf | Person | Song | 2003 |
| Yoko Hashimoto | Hallowed Transaction | ProducerOf | Person | Album | 2017 |
| Kaori Ito | Hallowed Transaction | ComposerOf | Person | Album | 2017 |
| Naoko Sato | Hallowed Transaction | LyricistOf | Person | Album | 2017 |
| Naoko Sato | Hallowed Transaction | PerformerOf | Person | Album | 2017 |
| Yan Tang | Altitude of Mistakes | ComposerOf | Person | Song | 2015 |
| Jeremiah Love | Twelve Bells of Augsburg | PerformerOf | Person | Song | 2026 |
| Jeremiah Love | Twelve Bells of Augsburg | ProducerOf | Person | Song | 2026 |
| Zachary Francis | Twelve Bells of Augsburg | PerformerOf | Person | Song | 2026 |
| Robert Woods | Twelve Bells of Augsburg | LyricistOf | Person | Song | 2026 |
| Robert Woods | Twelve Bells of Augsburg | PerformerOf | Person | Song | 2026 |
| Sean Jones | Partisan’s Lament | ProducerOf | Person | Song | 2030 |
| Min Long | Susurros de Pasión | LyricistOf | Person | Song | 2004 |
| Min Long | Susurros de Pasión | PerformerOf | Person | Song | 2004 |
| Gang Shao | Susurros de Pasión | LyricistOf | Person | Song | 2004 |
| Gang Shao | Susurros de Pasión | PerformerOf | Person | Song | 2004 |
| Qiang Yuan | Susurros de Pasión | PerformerOf | Person | Song | 2004 |
| Wei Zhao | Susurros de Pasión | ProducerOf | Person | Song | 2004 |
| Juan Zhou | Susurros de Pasión | PerformerOf | Person | Song | 2004 |
| Yang Zhao | Susurros de Pasión | PerformerOf | Person | Song | 2004 |
| Xiulan Fang | Altitude of Mistakes | PerformerOf | Person | Song | 2015 |
| Qiang Liu | Altitude of Mistakes | ComposerOf | Person | Song | 2015 |
| Qiang Liu | Altitude of Mistakes | PerformerOf | Person | Song | 2015 |
| Jun Qiao | Altitude of Mistakes | PerformerOf | Person | Song | 2015 |
| Ming Huang | Altitude of Mistakes | PerformerOf | Person | Song | 2015 |
| Samantha Bullock | Partisan’s Lament | PerformerOf | Person | Song | 2030 |
| Shannon Harvey | Our Inevitable Path | PerformerOf | Person | Song | 2017 |
| Shannon Harvey | Our Inevitable Path | LyricistOf | Person | Song | 2017 |
| Shannon Harvey | Our Inevitable Path | ComposerOf | Person | Song | 2017 |
| Lori Massey | When the Bar Lights Go Up | PerformerOf | Person | Song | 2000 |
| Lori Massey | When the Bar Lights Go Up | LyricistOf | Person | Song | 2000 |
| Sheryl Roman | Reflejo Interior | PerformerOf | Person | Song | 1983 |
| John Graves | Reflejo Interior | ComposerOf | Person | Song | 1983 |
| Jason Aguirre | Reflejo Interior | PerformerOf | Person | Song | 1983 |
| Matthew Lane | Reflejo Interior | ProducerOf | Person | Song | 1983 |
| Cipriano Peranda | Exoskeleton Dissolve | ComposerOf | Person | Song | 2020 |
| Igor Dyś | Echoes of Forgotten Light | PerformerOf | Person | Song | 2004 |
| Igor Dyś | Echoes of Forgotten Light | LyricistOf | Person | Song | 2004 |
| Tao Gao | Silent Steps in the Forest’s Embrace | PerformerOf | Person | Song | 2023 |
| Tao Gao | Silent Steps in the Forest’s Embrace | ProducerOf | Person | Song | 2023 |
| Joseph Ponce | Für uns alle zusammen | PerformerOf | Person | Song | 2017 |
| Herbert Mcfarland | Für uns alle zusammen | PerformerOf | Person | Song | 2017 |
| Rachel Montes | Für uns alle zusammen | PerformerOf | Person | Song | 2017 |
| Caitlin Miller | Für uns alle zusammen | LyricistOf | Person | Song | 2017 |
| Rebecca Wise | Für uns alle zusammen | PerformerOf | Person | Song | 2017 |
| Ping Liao | Dreamscape of Judgment | LyricistOf | Person | Song | 2023 |
| Trevor Bass | Dreamscape of Judgment | PerformerOf | Person | Song | 2023 |
| Xiuying Du | Dreamscape of Judgment | PerformerOf | Person | Song | 2023 |
| Xia Yuan | Dreamscape of Judgment | PerformerOf | Person | Song | 2023 |
| Sandra Burke | Dreamscape of Judgment | PerformerOf | Person | Song | 2023 |
| Yan Li | Dreamscape of Judgment | ComposerOf | Person | Song | 2023 |
| Yan Li | Dreamscape of Judgment | PerformerOf | Person | Song | 2023 |
| Ping Jin | Parallel Memories | LyricistOf | Person | Song | 2028 |
| Tao Wen | Parallel Memories | PerformerOf | Person | Song | 2028 |
| Tao Wen | Parallel Memories | ProducerOf | Person | Song | 2028 |
| Guiying Ren | Parallel Memories | PerformerOf | Person | Song | 2028 |
| Lei Shi | Parallel Memories | PerformerOf | Person | Song | 2028 |
| Wei Liang | Parallel Memories | ComposerOf | Person | Song | 2028 |
| Wei Liang | Parallel Memories | PerformerOf | Person | Song | 2028 |
| Joshua Taylor | Divergent Memories | PerformerOf | Person | Song | 2017 |
| Joshua Taylor | Divergent Memories | LyricistOf | Person | Song | 2017 |
| Joshua Taylor | Divergent Memories | ComposerOf | Person | Song | 2017 |
| Lei Fu | Provence Aria | PerformerOf | Person | Song | 2023 |
| Lei Fu | Provence Aria | ProducerOf | Person | Song | 2023 |
| Yan Dai | Provence Aria | LyricistOf | Person | Song | 2023 |
| Yan Dai | Provence Aria | PerformerOf | Person | Song | 2023 |
| Daniel Mccormick | Provence Aria | PerformerOf | Person | Song | 2023 |
| Lei Jin | Provence Aria | PerformerOf | Person | Song | 2023 |
| James Medina | Coastal Whispers of Biscay | PerformerOf | Person | Song | 2031 |
| Matthew Best | Coastal Whispers of Biscay | LyricistOf | Person | Song | 2031 |
| Matthew Best | Coastal Whispers of Biscay | PerformerOf | Person | Song | 2031 |
| Linda Burns | Coastal Whispers of Biscay | PerformerOf | Person | Song | 2031 |
| Brooke Olson | Folklore’s Heartbeat | PerformerOf | Person | Song | 2020 |
| David Reese | Exoskeleton Dissolve | PerformerOf | Person | Song | 2020 |
| Ryan Dunlap | Exoskeleton Dissolve | PerformerOf | Person | Song | 2020 |
| Elmo Calbo | Exoskeleton Dissolve | PerformerOf | Person | Song | 2020 |
| Amico Luciani | Exoskeleton Dissolve | PerformerOf | Person | Song | 2020 |
| Wei Zhao | Silent Steps in the Forest’s Embrace | ComposerOf | Person | Song | 2023 |
| Tao Hu | Folklore’s Heartbeat | ComposerOf | Person | Song | 2020 |
| Tao Hu | Folklore’s Heartbeat | PerformerOf | Person | Song | 2020 |
| Ming Qiao | The Long Way Home | LyricistOf | Person | Album | 2023 |
| Na Ye | The Long Way Home | PerformerOf | Person | Album | 2023 |
| Na Ye | The Long Way Home | ProducerOf | Person | Album | 2023 |
| Debra Graham | Weathered Miles | ComposerOf | Person | Song | 2023 |
| Debra Graham | Weathered Miles | PerformerOf | Person | Song | 2023 |
| Edward Evans | Weathered Miles | LyricistOf | Person | Song | 2023 |
| Dylan Schwartz | Weathered Miles | PerformerOf | Person | Song | 2023 |
| James Henderson | Weathered Miles | PerformerOf | Person | Song | 2023 |
| Amy Powell | Weathered Miles | ProducerOf | Person | Song | 2023 |
| Shawn Johnson | Weathered Miles | ProducerOf | Person | Song | 2023 |
| Joel Long | Weathered Miles | PerformerOf | Person | Song | 2023 |
| Michael Nixon | Weathered Miles | ComposerOf | Person | Song | 2023 |
| Michael Nixon | Weathered Miles | PerformerOf | Person | Song | 2023 |
| Nathan Jones | Weathered Miles | PerformerOf | Person | Song | 2023 |
| Kimberly Estrada | Twilight’s Threshold | PerformerOf | Person | Song | 2007 |
| Fang Ding | Twilight’s Threshold | PerformerOf | Person | Song | 2007 |
| Miss Jennifer Williams | The Crown We Wear | PerformerOf | Person | Song | 2022 |
| William Robinson | The Crown We Wear | PerformerOf | Person | Song | 2022 |
| William Robinson | The Crown We Wear | ProducerOf | Person | Song | 2022 |
| Judy Yang | The Crown We Wear | PerformerOf | Person | Song | 2022 |
| Joanna Avery | The Crown We Wear | PerformerOf | Person | Song | 2022 |
| Molly Gonzalez | The Crown We Wear | PerformerOf | Person | Song | 2022 |
| Juan Shi | Sacred Fragments | PerformerOf | Person | Song | 2003 |
| Ming Zhou | Sacred Fragments | PerformerOf | Person | Song | 2003 |
| Simone Säuberlich | Harvest Dance at Ler Valley | ComposerOf | Person | Song | 2010 |
| Simone Säuberlich | Harvest Dance at Ler Valley | PerformerOf | Person | Song | 2010 |
| Oceanic Records | The Saltwater Weavers | ProducerOf | RecordLabel | MusicalGroup | NA |
| Oceanic Records | The Tide Singer’s Knot | ProducerOf | RecordLabel | MusicalGroup | NA |
| Oceanus Records | The Salty Wakes | ProducerOf | RecordLabel | MusicalGroup | NA |
| Oceanus Records | The Fiddle & The Fjord | ProducerOf | RecordLabel | MusicalGroup | NA |

#### **Final Visualization Summary: Direct and Indirect Influences on Sailor Shift**

* First Layer: Nodes directly connected from Sailor Shift.

Includes:

Songs and albums (these will propagate influence to the next layer)

Bands and record labels (no further propagation needed)

* Second Layer: All works that are influenced by the first-layer songs/albums and are linked to other individuals.
* Third Layer: The creators, producers, and other contributors behind the second-layer works, completing the indirect influence chain.

nodes\_tbl <- nodes\_tbl %>% mutate(index = row\_number())  
  
sailor\_index <- nodes\_tbl %>%  
 filter(name == "Sailor Shift") %>%  
 pull(index)  
  
edges\_from\_sailor <- edges\_tbl %>%  
 filter(from == sailor\_index)  
  
first\_layer\_info <- edges\_from\_sailor %>%  
 inner\_join(nodes\_tbl, by = c("to" = "index")) %>%  
 filter(`Node Type` %in% c("Song", "Album", "MusicalGroup", "RecordLabel"))  
  
first\_layer\_targets <- first\_layer\_info$to  
  
influence\_edges <- c("InStyleOf", "LyricalReferenceTo", "InterpolatesFrom", "CoverOf", "DirectlySamples")  
  
edges\_2nd <- edges\_tbl %>%  
 filter(from %in% first\_layer\_targets & `Edge Type` %in% influence\_edges)  
  
influenced\_works <- edges\_2nd$to  
songs\_with\_outgoing <- unique(edges\_2nd$from)  
  
person\_edge\_types <- c("PerformerOf", "ComposerOf", "LyricistOf", "ProducerOf")  
  
edges\_people\_to\_2nd <- edges\_tbl %>%  
 filter(  
 to %in% influenced\_works,  
 `Edge Type` %in% person\_edge\_types  
 ) %>%  
 left\_join(nodes\_tbl %>% select(index, `Node Type`), by = c("from" = "index")) %>%  
 rename(`From Node Type` = `Node Type`) %>%  
 filter(`From Node Type` %in% c("Person", "RecordLabel")) %>%  
 left\_join(nodes\_tbl %>% select(index, `Node Type`, release\_date), by = c("to" = "index")) %>%  
 rename(`To Node Type` = `Node Type`, release\_date = release\_date)  
  
first\_layer\_filtered <- first\_layer\_info %>%  
 filter(  
 (`Node Type` %in% c("Song", "Album") & to %in% songs\_with\_outgoing) |  
 (`Node Type` %in% c("MusicalGroup", "RecordLabel"))  
 )  
  
edges\_from\_sailor\_filtered <- edges\_from\_sailor %>%  
 semi\_join(first\_layer\_filtered, by = c("to" = "to"))  
  
all\_edges <- bind\_rows(  
 edges\_from\_sailor\_filtered,  
 edges\_2nd,  
 edges\_people\_to\_2nd %>% rename(from = from, to = to)  
)  
  
node\_ids <- unique(c(all\_edges$from, all\_edges$to))  
  
nodes\_subgraph <- nodes\_tbl %>%  
 filter(index %in% node\_ids) %>%  
 transmute(  
 id = index,  
 label = name,  
 group = `Node Type`  
 )  
  
edges\_subgraph <- all\_edges %>%  
 transmute(  
 from = from,  
 to = to,  
 label = `Edge Type`,  
 title = paste0("Edge Type: ", `Edge Type`, "<br>Release: ", release\_date)  
 )

visNetwork(nodes\_subgraph, edges\_subgraph, width = "100%", height = "700px") %>%  
 visEdges(arrows = "to") %>%  
 visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE) %>%  
 visLegend(  
 useGroups = TRUE,  
 position = "right",  
 ) %>%  
 visPhysics(  
 solver = "forceAtlas2Based",  
 forceAtlas2Based = list(gravitationalConstant = -80),  
 stabilization = list(enabled = TRUE, iterations = 100)  
 ) %>%  
 visLayout(randomSeed = 123)

|  |
| --- |
| Who has she been most influenced? |
| The visual network clearly illustrates that Sailor Shift is not an isolated creator, but rather a convergence point within a multi-layered web of artistic influence.  She is directly influenced by several notable works, such as Tides of Echos and Electric Reverie, which themselves are shaped by earlier compositions and artists.  Among the sources of influence, four musical groups play a key role, including her own band Ivy Echos. The remaining groups—such as The Saltwater Weavers, The Fiddle & the Fjord, and The Salty Wakes—are projects she was closely involved in, either as a producer or performer. This dual role highlights her position as both an influencer and a participant in collaborative creation.  In the third layer of the influence network, most of the connections are formed through InStyleOf and InterpolatesFrom edges, indicating that Sailor Shift primarily draws inspiration from the style and structural elements of earlier works. Only a few edges fall under CoverOf, suggesting that her creative process leans more toward reinterpretation and transformation rather than direct replication.  This cascading network of style, melody, and lyrical influence reflects not only her artistic lineage, but also reveals the broader creative transmission ecosystem within the Oceanus Folk community. |

### **2.Who has she collaborated with and directly or indirectly influenced? &**

### **3.How has she influenced collaborators of the broader Oceanus Folk community**

#### First we will first answer ：Who has she influenced ?

And here is the directly neighbours :

nodes\_tbl <- nodes\_tbl %>% mutate(index = row\_number())  
  
sailor\_index <- nodes\_tbl %>%  
 filter(name == "Sailor Shift") %>%  
 pull(index)  
  
influence\_edges <- c("InStyleOf", "CoverOf", "InterpolatesFrom", "LyricalReferenceTo", "DirectlySamples")  
  
edges\_from\_sailor <- edges\_tbl %>%  
 filter(to == sailor\_index & `Edge Type` %in% influence\_edges)  
  
influenced1 <- edges\_from\_sailor %>%  
 left\_join(nodes\_tbl, by = c("from" = "index")) %>%  
 select(name, `Edge Type`, everything())

knitr::kable(influenced1)

| name | Edge Type | source | target | key | from | to | Node Type | single | release\_date | genre | notable | id | written\_date | stage\_name | notoriety\_date |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Cassette Future | InStyleOf | 17022 | 17255 | 0 | 17023 | 17256 | MusicalGroup | NA | NA | NA | NA | 17022 | NA | NA | NA |
| Silver Veil | LyricalReferenceTo | 17032 | 17255 | 0 | 17033 | 17256 | MusicalGroup | NA | NA | NA | NA | 17032 | NA | NA | NA |
| The Phantom Operators | InterpolatesFrom | 17063 | 17255 | 0 | 17064 | 17256 | MusicalGroup | NA | NA | NA | NA | 17063 | NA | NA | NA |
| The Hollow Monarchs | InStyleOf | 17110 | 17255 | 0 | 17111 | 17256 | MusicalGroup | NA | NA | NA | NA | 17110 | NA | NA | NA |
| Cassian Storm | LyricalReferenceTo | 17126 | 17255 | 0 | 17127 | 17256 | Person | NA | NA | NA | NA | 17126 | NA | Seraphina Vale | NA |
| Claire Holmes | InterpolatesFrom | 17155 | 17255 | 0 | 17156 | 17256 | Person | NA | NA | NA | NA | 17155 | NA | Jaye Finch | NA |
| Copper Canyon Ghosts | DirectlySamples | 17361 | 17255 | 0 | 17362 | 17256 | MusicalGroup | NA | NA | NA | NA | 17361 | NA | NA | NA |

subgraph\_in <- extract\_subnetwork(  
 graph,   
 node\_name = "Sailor Shift",   
 distance = 3,   
 direction = "in"  
)  
  
edges\_vn <- igraph::as\_data\_frame(subgraph\_in, what = "edges") %>%  
 filter(`Edge Type` != "ProducerOf") %>%  
 rename(from = from, to = to, label = `Edge Type`)  
  
nodes\_vn <- igraph::as\_data\_frame(subgraph\_in, what = "vertices") %>%  
 mutate(id = name, label = name, group = `Node Type`)  
  
used\_nodes <- unique(c(edges\_vn$from, edges\_vn$to))  
nodes\_vn <- nodes\_vn %>% filter(id %in% used\_nodes)

visNetwork(nodes\_vn, edges\_vn, height = "800px", width = "100%") %>%  
 visEdges(arrows = "to") %>%  
 visOptions(  
 highlightNearest = list(enabled = TRUE, degree = 3, hover = TRUE),  
 nodesIdSelection = TRUE,  
 selectedBy = "group"  
 ) %>%  
 visLegend() %>%  
 visLayout(randomSeed = 1234) %>%  
 visInteraction(navigationButtons = TRUE)

|  |
| --- |
| Who has she influenced directly or indirectly？ |
| These are edges originating from Sailor Shift, indicating how she influences other groups or works:   * InStyleOf: stylistic influence or similar musical style * InterpolatesFrom: portions of her work have been interpolated * LyricalReferenceTo: her name or lyrics are referenced * DirectlySamples: her work has been directly sampled   Sailor Shift has a broad creative influence   * She influences at least 5 different groups (e.g., Cassette Future, The Hollow Monarchs) * The types of influence span style, lyrics, and sampling.   An indirect “sub-network” is formed   * For example, through the path Sailor Shift → Cassette Future → Zara Quinn, her influence extends to band members as well.   Strong structural centrality   * Sailor Shift is the hub of this network, with nearly all edges revolving around her. * This suggests she holds a central position of influence in this music community. |

***We attempted to trace Sailor Shift’s indirect influence through songs and albums to identify impacted individuals. However, this approach yielded no results, so we decided not to pursue this path further.***

Based on the results, Copper Canyon Ghosts is identified as an Oceanus Folk band. To explore the third question—“**How has Sailor Shift influenced collaborators within the broader Oceanus Folk community**?”—we will examine both direct and indirect influence paths .

#### Identify the artists and groups influenced by Sailor Shift in Oceanus Folk Genre

First, we identify the artists and groups influenced by Sailor Shift. Among them, we determine which ones belong to the Oceanus Folk genre, using “genre” as our key search indicator.

nodes\_tbl <- nodes\_tbl %>% mutate(index = row\_number())  
  
creator\_names <- c(  
 "Zara Quinn", "Milo Knight", "Cassette Future", "Eliza Brooks",  
 "Jasper Reed", "Silver Veil", "Juno Ray", "Savannah Teal",  
 "The Phantom Operators", "Chloe Montgomery", "Nathaniel Brooks", "Clara Davis",  
 "The Hollow Monarchs", "Cassian Storm", "Claire Holmes", "Beatrice Albright",  
 "Daniel O'Connell", "Copper Canyon Ghosts"  
)  
  
creator\_ids <- nodes\_tbl %>%  
 filter(name %in% creator\_names) %>%  
 pull(index)  
  
target\_edge\_types <- c("PerformerOf", "ComposerOf", "LyricistOf", "ProducerOf")  
  
edges\_out <- edges\_tbl %>%  
 filter(from %in% creator\_ids, `Edge Type` %in% target\_edge\_types)  
  
edges\_out\_full <- edges\_out %>%  
 left\_join(nodes\_tbl %>% mutate(index = row\_number()), by = c("to" = "index")) %>%  
 select(from, to, `Edge Type`, name, `Node Type`, release\_date,genre,notable,notoriety\_date)  
  
edges\_out\_full <- edges\_out\_full %>%  
 left\_join(nodes\_tbl %>% select(index, creator\_name = name), by = c("from" = "index")) %>%  
 select(creator\_name, `Edge Type`, name, `Node Type`, release\_date,genre,notable,notoriety\_date)

edges\_oceanus <- edges\_out\_full %>%  
 filter(genre == "Oceanus Folk")  
  
knitr::kable(head(edges\_oceanus, 6))

| creator\_name | Edge Type | name | Node Type | release\_date | genre | notable | notoriety\_date |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Beatrice Albright | PerformerOf | Tidesworn Ballads | Album | 2031 | Oceanus Folk | TRUE | NA |
| Beatrice Albright | LyricistOf | Tidesworn Ballads | Album | 2031 | Oceanus Folk | TRUE | NA |
| Beatrice Albright | PerformerOf | Tidesworn Ballads | Album | 2031 | Oceanus Folk | TRUE | NA |
| Beatrice Albright | PerformerOf | The Siren’s Sigh | Song | 2031 | Oceanus Folk | TRUE | 2031 |
| Beatrice Albright | PerformerOf | Whispers on the Jetty Wind | Song | 2031 | Oceanus Folk | TRUE | 2031 |
| Beatrice Albright | PerformerOf | Moonlit Kelp Forest Dance | Song | 2031 | Oceanus Folk | TRUE | 2031 |

This table summarizes the creative contributions of Beatrice Albright, Daniel O’Connell, and Copper Canyon Ghosts to various albums and songs—all of which are part of the Oceanus Folk genre

edges\_sailor <- edges\_all %>%  
 filter(to == "Sailor Shift") %>%  
 filter(`Edge Type` %in% c("InStyleOf", "LyricalReferenceTo", "CoverOf", "InterpolatesFrom", "DirectlySamples")) %>%  
 select(from, to, `Edge Type`)  
  
creator\_work\_nodes <- unique(c(edges\_oceanus$creator\_name, edges\_oceanus$name))  
  
all\_node\_names <- union(  
 creator\_work\_nodes,  
 unique(c(edges\_sailor$from, edges\_sailor$to, "Sailor Shift", "Copper Canyon Ghosts"))   
)  
  
nodes\_vn <- nodes\_tbl %>%  
 filter(name %in% all\_node\_names) %>%  
 mutate(  
 id = index,  
 label = name,  
 group = `Node Type`,  
 color.background = ifelse(name == "Sailor Shift", "blue", NA),   
 shape = ifelse(name == "Sailor Shift", "star", "dot")  
 )  
  
edges\_vn\_creator <- edges\_oceanus %>%  
 rename(from = creator\_name, to = name, label = `Edge Type`) %>%  
 left\_join(nodes\_vn %>% select(name, id), by = c("from" = "name")) %>%  
 rename(from\_id = id) %>%  
 left\_join(nodes\_vn %>% select(name, id), by = c("to" = "name")) %>%  
 rename(to\_id = id) %>%  
 mutate(length = NA) %>%  
 select(from = from\_id, to = to\_id, label, length) %>%  
 filter(!is.na(from) & !is.na(to))  
  
edges\_vn\_sailor <- edges\_sailor %>%  
 left\_join(nodes\_vn %>% select(name, id), by = c("from" = "name")) %>%  
 rename(from\_id = id) %>%  
 left\_join(nodes\_vn %>% select(name, id), by = c("to" = "name")) %>%  
 rename(to\_id = id) %>%  
 mutate(length = NA) %>%  
 select(from = from\_id, to = to\_id, label = `Edge Type`, length) %>%  
 filter(!is.na(from) & !is.na(to))  
  
forced\_edge <- tibble(  
 from = "Copper Canyon Ghosts",  
 to = "Sailor Shift",  
 label = "DirectlySamples",  
 length = 400  
)  
  
forced\_edge\_ids <- forced\_edge %>%  
 left\_join(nodes\_vn %>% select(name, id), by = c("from" = "name")) %>%  
 rename(from\_id = id) %>%  
 left\_join(nodes\_vn %>% select(name, id), by = c("to" = "name")) %>%  
 rename(to\_id = id) %>%  
 select(from = from\_id, to = to\_id, label, length) %>%  
 filter(!is.na(from) & !is.na(to))  
  
edges\_final <- bind\_rows(  
 edges\_vn\_creator,  
 edges\_vn\_sailor,  
 forced\_edge\_ids  
)

visNetwork(nodes\_vn, edges\_final, height = "700px", width = "100%") %>%  
 visEdges(arrows = "to") %>%  
 visOptions(  
 highlightNearest = TRUE,  
 nodesIdSelection = TRUE,  
 selectedBy = "group"  
 ) %>%  
 visLegend() %>%  
 visLayout(randomSeed = 42) %>%  
 visPhysics(  
 solver = "forceAtlas2Based",  
 forceAtlas2Based = list(gravitationalConstant = -80),  
 stabilization = list(enabled = TRUE, iterations = 100)  
 )

|  |
| --- |
| How has Sailor Shift influenced collaborators within the broader Oceanus Folk community? |
| Sailor Shift → influences Copper Canyon Ghosts → performs representative works → connects with Daniel and Beatrice through shared creations  This influence pathway shows that Sailor Shift’s creative style and content are indirectly transmitted to other members of the Oceanus Folk community through collaborative performances and reinterpretations.  The network visualization reveals that Sailor Shift influences key figures in the Oceanus Folk scene both directly and indirectly.  She is sampled by Copper Canyon Ghosts, a band that performs major works composed by Daniel O’Connell and Beatrice Albright.  These shared works form the musical core of the Oceanus Folk genre, creating a multi-layered influence network with Sailor Shift at its source.  Her creative impact extends beyond direct collaboration—spreading throughout the community via the chain of **work → band → collaborators**, shaping the genre’s evolution and collective identity. |

## Summary of Question 1

Sailor Shift is both a cultural inheritor and a creative engine. She draws on decades of musical history, spanning genres from Synthwave to Americana, integrating them into the Oceanus Folk style. Through direct performances and indirect stylistic channels, she influences a wide web of collaborators and emerging artists—serving as a stylistic and structural anchor for the Oceanus Folk scene. Her work demonstrates a rare balance between solo artistry and community participation, making her a pivotal node in the evolution of this genre.