Take\_Home\_2 - Mini case 1

patriciatrisno

2025-05-31

## 1 Setting the Scene

### **1.1 Background**

The quiet island nation, once mostly known for its fishing industry, is now seeing a big rise in tourism. This growing interest is mainly because of the increasing popularity of the Oceanus Folk genre, which started on the island and became globally known through its local superstar, Sailor Shift.

Sailor began her music career in 2023 as part of the Oceanus Folk band *Ivy Echoes*. After the group disbanded in 2026, she found solo success in 2028. Her viral single not only hit the top of global charts—becoming the first Oceanus Folk song to do so—but also pushed her into the spotlight as an international star.

Since then, Sailor has released a new album nearly every year, each one doing better than the last. Even though she’s explored other genres like Indie Pop and Indie Folk through collaborations, she always circles back to her Oceanus Folk roots. This genre’s global reach keeps growing, and a big part of that is thanks to Sailor’s work. Her collaborations stand out because she stays focused on two things:

1. Sharing and promoting Oceanus Folk.
2. Helping new artists grow.

Aside from her own career, Sailor’s success also brought attention to her former *Ivy Echoes* bandmates. All of them are still active in the music scene. Together, they’ve helped grow Oceanus Folk’s influence and opened the door for a new wave of rising artists in the genre.

### **1.2 Task**

For these 2025 VAST Challenge (MC1), this project aims to support local journalist Silas Reed in crafting a piece titled “Oceanus Folk: Then-and-Now.” This article will chronicle the ascent of Sailor Shift and the pervasive influence of Oceanus Folk within the music industry. To achieve this, we’re leveraging a comprehensive dataset that includes musical artists, producers, albums, songs, and their intricate influence networks, all organized as a knowledge graph.

Our goal is to transform this rich data into insightful visualizations, revealing new and compelling information about Sailor’s journey to stardom and the broader impact of Oceanus Folk on the music world which summarized in 3 questions:

1. Was the influence intermittent (not continuous, not stable, occurring at irregular intervals) ?
2. What genre and top artist have been most influenced by Oceanus Folk?
3. How has Oceanus Folk changed with the rise of Sailor Shift? From which genre does it draw most of its contemporary inspiration?

## 2 Data Preparation

### 2.1 Package Installation

We start by loading the required R packages using the pacman::p\_load() function. Each package serves a specific purpose for our analysis:

## Code

pacman::p\_load(  
 jsonlite, # for reading JSON files  
 tidyverse, # includes dplyr, tidyr, ggplot2, stringr, etc.  
 lubridate, # for date handling  
 patchwork, # for combining ggplots  
 ggraph, # for graph visualizations  
 tidygraph, # for working with tidy network data  
 igraph, # backend graph structure  
 visNetwork, # for interactive network graphs  
 kableExtra, # for enhanced table formatting (if used in reporting)  
 shiny  
)

### 2.2 Loading Data

In this step, we load the data from the **VAST Challenge 2025 – Mini Challenge 1** dataset.

This dataset contains:

* **17,412 nodes**
* **37,857 edges**

The datas represent musical entities such as **artists**, **songs**, **albums**, **music groups**, and **genres**. which also include the the edges that describe relationships like **collaborations** and **influences**.

Additionally we will also explore the data structure to …..

## Code

# Load Data  
data <- fromJSON("~/Documents/SMU/April Term 2/Visual Analytics/patriciatrisno/ISSS608-VAA/VAST Challenge 2025/MC1/MC1\_graph.json")

## Data Structure

str(data, 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:

The data includes:

* Node types: Person, Song, RecordLabel, Album, MusicalGroup
* Edge types: MemberOf, PerformerOf, ComposerOf, ProducerOf, LyricistOf, InStyleOf, InterpolatesFrom, CoverOf, LyricalReferenceTo, DirectlySamples, RecordedBy, DistributedBy

### 2.3 Extract Edges and Nodes Tables

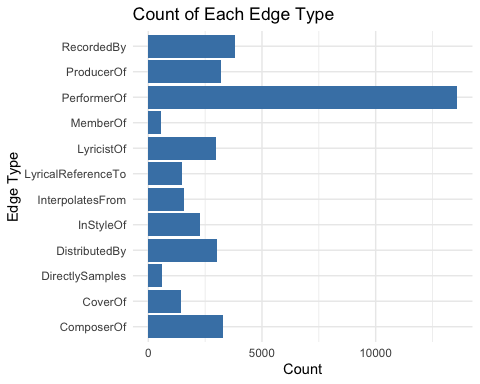
We convert the JSON data into data frames for easier analysis and transformation.

This step helps us **organize the data for analysis**, such as filtering, labeling, and visualizing relationships. We also convert date columns into integers to simplify later operations like filtering and sorting by year.

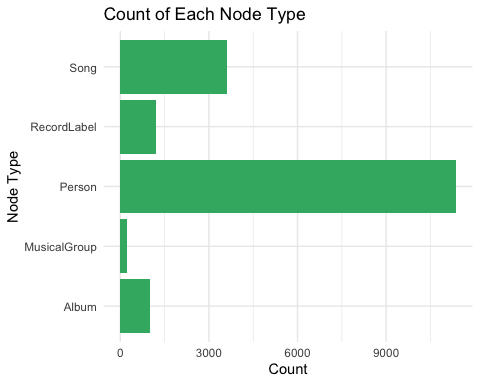
nodes\_tbl <- as\_tibble(data$nodes)  
edges\_tbl <- as\_tibble(data$links)  
  
# Convert Dates to Integers and Clean NA Values  
nodes\_tbl <- nodes\_tbl %>%  
 mutate(across(c(release\_date, notoriety\_date, written\_date),  
 ~as.integer(if\_else(`Node Type` %in% c("Song", "Album"), ., NA\_character\_))))

### 2.4 Initial EDA

ggplot(data = edges\_tbl, aes(y = `Edge Type`)) +  
 geom\_bar(fill = "steelblue") +  
 labs(title = "Count of Each Edge Type", y = "Edge Type", x = "Count") +  
 theme\_minimal()



ggplot(data = nodes\_tbl, aes(y = `Node Type`)) +  
 geom\_bar(fill = "mediumseagreen") +  
 labs(title = "Count of Each Node Type", y = "Node Type", x = "Count") +  
 theme\_minimal()



### 2.5 Labeling the Nodes and Edges

This step is important for **categorizing the data** based on its type and role. It allows us to clearly define what each node and edge represents in the graph.

We group similar edge types into broader categories for clearer analysis:

* **Creator Of**: edges like PerformerOf, ComposerOf, ProducerOf, LyricistOf, etc.
* **Influenced By**: edges like InterpolatesFrom, InStyleOf, CoverOf, etc.
* **Member Of**: edges that indicate group memberships

We also label nodes by type and genre. For instance, nodes in the genre **Oceanus Folk** are marked accordingly.

# Group Similar Edge Types  
edges\_tbl <- edges\_tbl %>%  
 mutate(`Edge Define` = case\_when(  
 `Edge Type` %in% c("PerformerOf", "ComposerOf", "ProducerOf", "LyricistOf") ~ "Creator Of",  
 `Edge Type` %in% c("InStyleOf", "InterpolatesFrom", "CoverOf", "LyricalReferenceTo", "DirectlySamples") ~ "Influenced By",  
 `Edge Type` == "MemberOf" ~ "Member Of",  
 TRUE ~ "Other"  
 ))  
  
# Label Nodes and Adding Column for Further Needs  
nodes\_tbl <- nodes\_tbl %>%  
 mutate(`Node Colour` = case\_when(  
 `Node Type` == "Person" ~ "Person",  
 `Node Type` == "MusicalGroup" ~ "MusicalGroup",  
 `Node Type` == "Song" ~ "Song",  
 `Node Type` == "Album" ~ "Album",  
 `Node Type` == "Record Label" ~ "Record Label",  
 ),  
 `Node Shape` = case\_when(  
 `Node Type` %in% c("Person", "MusicalGroup") ~ "Artist",  
 `Node Type` %in% c("Song", "Album") ~ "Music",  
 TRUE ~ "Record Label"  
 ),  
 label = case\_when(  
 `Node Type` %in% c("Person", "MusicalGroup", "RecordLabel") ~ name,  
 `Node Type` %in% c("Song", "Album") ~ paste0(name, " (", `Node Type`, ")"),  
 TRUE ~ name  
 ),  
 role = `Node Type`,  
 title = paste0("<b>Name:</b> ", name,  
 "<br><b>Role:</b> ", role,  
 ifelse(!is.na(genre), paste0("<br><b>Genre:</b> ", genre), ""),  
 ifelse(!is.na(release\_date), paste0("<br><b>Year:</b> ", release\_date), ""))  
 )

### 2.6 Set the Commonalities

#### 2.6.1 Define the shape

To make the graph clearer, we assign shapes to node categories:

* **Person/Musical Group**: Triangle (▲)
* **Song/Album**: Square (■)
* **Other**: Circle (●)

This helps in visually identifying different node types in the network graph.

# Define Shape Mapping  
shape\_map <- c(  
 "Artist" = 17, # Triangle ▲  
 "Music" = 15, # Square ■  
 "Record Label" = 16 # Circle ●  
)

#### **2.6.2 Color Mapping**

# Define Color Mappings  
node\_colors <- c(  
 "Person" = "deepskyblue",  
 "MusicalGroup" = "purple",  
 "RecordLabel" = "lightpink",  
 "Song" = "darkred",  
 "Album" = "lightblue",  
 "Other Genre" = "gray",  
 "Oceanus Folk" = "gold"  
)  
  
edge\_colors <- c(  
 "Influenced By" = "#FF5733",   
 "Creator Of" = "#8A2BE2",  
 `NA` = "transparent"  
   
)

## 3 Create Knowledge Graph

### 3.1 ID Mapping

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,  
 index = seq\_len(  
 nrow(nodes\_tbl)))  
  
# Map source and target IDs to row indices  
edges\_tbl <- edges\_tbl %>%  
 left\_join(id\_map, by = c("source" = "id")) %>%  
 rename(from = index) %>%  
 left\_join(id\_map, by = c("target" = "id")) %>%  
 rename(to = index)  
  
# Filter out any unmatched (invalid) edges  
edges\_tbl <- edges\_tbl %>%  
 filter(!is.na(from), !is.na(to))

### **3.2 Creating tidygraph**

tbl\_graph() creates the graph structure in R. It connects **nodes** and **edges** based on relationships (influences, creator of songs, etc.). We specify directed = TRUE because influence flows in a direction.

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

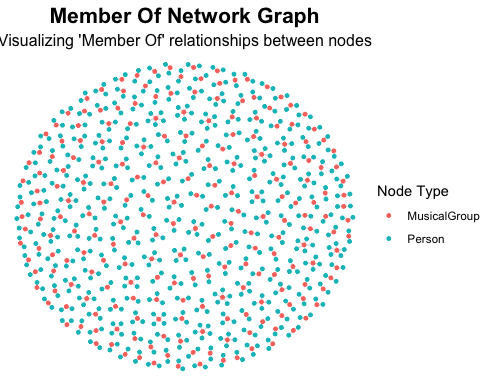
## **4 Visualizing Knowledge Graph - Part 1**

This part we will visualize the whole graph with the data set from nodes\_tbl and edges\_tbl

set.seed(1234)

### 4.1 Sub-Graph - Member Of

# Filter only Member Of  
graph\_member\_of <- graph %>%  
 activate(edges) %>%  
 filter(`Edge Type` == "MemberOf")  
  
#Extract Connected Nodes Only  
used\_node\_indices <- graph\_member\_of %>%  
 activate(edges) %>%  
 as\_tibble() %>%  
 select(from, to) %>%  
 unlist() %>%  
 unique()  
  
# Keep Only the Choosen Nodes  
graph\_member\_of <- graph\_member\_of %>%  
 activate(nodes) %>%  
 mutate(row\_id = row\_number()) %>%  
 filter(row\_id %in% used\_node\_indices) %>%  
 select(-row\_id) # optional cleanup  
  
# Plotting   
ggraph(graph\_member\_of, 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() +  
 labs(  
 title = "Member Of Network Graph",  
 subtitle = "Visualizing 'Member Of' relationships between nodes"  
 ) +  
 theme(  
 plot.title = element\_text(size = 16, face = "bold", hjust = 0.5),  
 plot.subtitle = element\_text(size = 12, hjust = 0.5)  
 )



This Graph can give rough idea on how many artist and musical group are listed, while for more details we also able to see to which musical group each artist part of

## 5 Understand Sailor Shift Influence Networks

Sailor Shift stands out as a central figure in the music industry, particularly within the **Oceanus Folk** genre. Her prominence makes her a key subject for analyzing influence patterns.

In this part we aim to visualize the **influence network** of *Sailor Shift.* This helps us understand how the **Oceanus Folk genre spreads** through *Sailor Shift’s* network. As for this pre visualization we will focusing on:

* Start with *Sailor Shift* (the person node).
* **First-level:** Artists or musical groups directly influencing *Sailor Shift*.
* **Second-level:** Songs or albums created by these artists.

This helps narrow down the graph to **relevant and insightful nodes**, reducing clutter.

We extract only the **relevant nodes and edges** for visualization, focusing on *Sailor Shift’s* influence network within **Oceanus Folk**. This avoids clutter and highlights key relationships.

# Set target: Sailor Shift, get the id  
sailor\_id <- nodes\_tbl %>%  
 filter(name == "Sailor Shift") %>%  
 pull(id)  
  
# 1st Level: Artists Influencing Sailor Shift   
sailor\_first\_edge <- edges\_tbl %>%  
 filter(`Edge Define` == "Influenced By", target %in% sailor\_id)  
  
sailor\_first\_artists <- sailor\_first\_edge %>%  
 pull(source)  
  
  
# 2nd Level: Works Created by 1st-Level Artists  
sailor\_second\_edge <- edges\_tbl %>%  
 filter(`Edge Define` == "Creator Of", source %in% sailor\_first\_artists )  
  
sailor\_second\_works <- sailor\_second\_edge %>%  
 filter(target %in% nodes\_tbl$id[nodes\_tbl$`Node Type` %in% c("Song", "Album")]) %>%  
 pull(target)  
  
# 3rd Level: Artists Influenced by Those Works  
sailor\_third\_edge <- edges\_tbl %>%  
 filter(`Edge Define` == "Influenced By", source %in% sailor\_second\_works)  
  
sailor\_third\_artists <- sailor\_third\_edge %>%  
 pull(target)  
  
# Collect All Relevant Nodes  
relevant\_node\_ids <- unique(c(sailor\_id, sailor\_first\_artists, sailor\_second\_works, sailor\_third\_artists))  
  
nodes\_viz <- nodes\_tbl %>%  
 filter(id %in% relevant\_node\_ids) %>%  
 mutate(id = as.character(id),  
 label = name)  
  
edges\_viz <- edges\_tbl %>%  
 filter(source %in% relevant\_node\_ids & target %in% relevant\_node\_ids) %>%  
 mutate(from = as.character(source),  
 to = as.character(target),  
 arrows = "to",  
 title = paste0("<b>Type:</b> ", `Edge Define`,  
 "<br><b>From:</b> ", from,  
 "<br><b>To:</b> ", to))  
  
nodes\_viz <- nodes\_viz %>%  
 mutate(shape = case\_when(  
 role == "Person" ~ "triangle",  
 genre == "Oceanus Folk" ~ "square",  
 `Node Shape` == "Music" ~ "dot",  
 TRUE ~ "pyramid"  
 ))  
  
group\_colors <- list(  
 "Person" = "deepskyblue",  
 "MusicalGroup" = "purple",  
 "RecordLabel" = "lightpink",  
 "Song" = "darkred",  
 "Album" = "lightblue",  
 "Other Genre" = "gray",  
 "Oceanus Folk" = "gold"  
)  
  
tagList(  
 tags$h2("Sailor Shift’s Extended Influence Network",   
 style = "text-align:center; font-weight:bold; margin-bottom:20px;"),  
   
 visNetwork(nodes\_viz, edges\_viz, width = "100%", height = "700px") %>%  
 visNodes(color = list(highlight = "orange")) %>%  
 visEdges(color = list(color = "#848484", highlight = "red")) %>%  
   
 visGroups(groupname = "Person", color = group\_colors[["Person"]]) %>%  
 visGroups(groupname = "MusicalGroup", color = group\_colors[["MusicalGroup"]]) %>%  
 visGroups(groupname = "RecordLabel", color = group\_colors[["RecordLabel"]]) %>%  
 visGroups(groupname = "Song", color = group\_colors[["Song"]]) %>%  
 visGroups(groupname = "Album", color = group\_colors[["Album"]]) %>%  
 visGroups(groupname = "Oceanus Folk", color = group\_colors[["Oceanus Folk"]]) %>%  
   
 visLegend(addNodes = list(  
 list(label = "Person", shape = "triangle", color = "deepskyblue"),  
 list(label = "Musical Group", shape = "dot", color = "deepskyblue"),  
 list(label = "Record Label", shape = "dot", color = "deepskyblue"),  
 list(label = "Song", shape = "dot", color = "deepskyblue"),  
 list(label = "Album", shape = "dot", color = "deepskyblue"),  
 list(label = "Oceanus Folk", shape = "square", color = "deepskyblue")  
 ),  
 useGroups = FALSE, position = "right", width = 0.2) %>%  
   
 visLayout(randomSeed = 42) %>%  
 visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE)  
)

Since Sailor is also a member of Ivy Echos—a group that played a key role alongside her in shaping the Oceanus Folk genre on the Island—it would be valuable for us to also examine Ivy Echos’ influence.

### Ivy Echos - Sailor’s Musical Group Influence

# Set target: Ivy Echos, get the id  
ivy\_id <- nodes\_tbl %>%  
 filter(name == "Ivy Echos") %>%  
 pull(id)  
  
# 1st Level: Artists Influencing Ivy Echos   
ivy\_first\_edge <- edges\_tbl %>%  
 filter(`Edge Define` == "Influenced By", target %in% ivy\_id)  
  
ivy\_first\_artists <- ivy\_first\_edge %>%  
 pull(source)  
  
# 2nd Level: Works Created by 1st-Level Artists  
ivy\_second\_edge <- edges\_tbl %>%  
 filter(`Edge Define` == "Creator Of", source %in% ivy\_first\_artists)  
  
ivy\_second\_works <- ivy\_second\_edge %>%  
 filter(target %in% nodes\_tbl$id[nodes\_tbl$`Node Type` %in% c("Song", "Album")]) %>%  
 pull(target)  
  
# 3rd Level: Artists Influenced by Those Works  
ivy\_third\_edge <- edges\_tbl %>%  
 filter(`Edge Define` == "Influenced By", source %in% ivy\_second\_works)  
  
ivy\_third\_artists <- ivy\_third\_edge %>%  
 pull(target)  
  
relevant\_node\_ids <- unique(c(ivy\_id, ivy\_first\_artists, ivy\_second\_works, ivy\_third\_artists))  
  
nodes\_viz <- nodes\_tbl %>%  
 filter(id %in% relevant\_node\_ids) %>%  
 mutate(  
 id = as.character(id),  
 label = name  
 )  
  
  
edges\_viz <- edges\_tbl %>%  
 filter(source %in% relevant\_node\_ids & target %in% relevant\_node\_ids) %>%  
 mutate(  
 from = as.character(source),  
 to = as.character(target),  
 arrows = "to",  
 title = paste0("<b>Type:</b> ", `Edge Define`,  
 "<br><b>From:</b> ", from,  
 "<br><b>To:</b> ", to)  
 )  
  
  
nodes\_viz <- nodes\_viz %>%  
 mutate(shape = case\_when(  
 role == "Person" ~ "triangle",  
 genre == "Oceanus Folk" ~ "square",  
 `Node Shape` == "Music" ~ "dot",  
 TRUE ~ "pyramid"  
 ))  
  
group\_colors <- list(  
 "Person" = "deepskyblue",  
 "MusicalGroup" = "purple",  
 "RecordLabel" = "lightpink",  
 "Song" = "darkred",  
 "Album" = "lightblue",  
 "Other Genre" = "gray",  
 "Oceanus Folk" = "gold"  
)  
  
  
tagList(  
 tags$h2("Ivy Echos’ Extended Influence Network",   
 style = "text-align:center; font-weight:bold; margin-bottom:20px;"),  
   
 visNetwork(nodes\_viz, edges\_viz, width = "100%", height = "700px") %>%  
 visNodes(color = list(highlight = "orange")) %>%  
 visEdges(color = list(color = "#848484", highlight = "red")) %>%  
   
 visGroups(groupname = "Person", color = "deepskyblue") %>%  
 visGroups(groupname = "MusicalGroup", color = "deepskyblue") %>%  
 visGroups(groupname = "RecordLabel", color = "deepskyblue") %>%  
 visGroups(groupname = "Song", color = "deepskyblue") %>%  
 visGroups(groupname = "Album", color = "deepskyblue") %>%  
 visGroups(groupname = "Oceanus Folk", color = "deepskyblue") %>%  
   
 visLegend(  
 addNodes = list(  
 list(label = "Person", shape = "triangle", color = "deepskyblue"),  
 list(label = "Musical Group", shape = "dot", color = "deepskyblue"),  
 list(label = "Record Label", shape = "dot", color = "deepskyblue"),  
 list(label = "Song", shape = "dot", color = "deepskyblue"),  
 list(label = "Album", shape = "dot", color = "deepskyblue"),  
 list(label = "Oceanus Folk", shape = "square", color = "deepskyblue")  
 ),  
 useGroups = FALSE,  
 position = "right",  
 width = 0.2  
 ) %>%  
   
 visLayout(randomSeed = 42) %>%  
 visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE)  
)

### **Insights**

The graph reveals Sailor Shift’s strong influence within the Oceanus Folk scene.

Her immediate connections (first-level nodes) include both albums and artists, with a majority being albums (five), and only two direct artist influences. The Oceanus Folk genre appears clustered, indicating that its influence tends to stay within specific genre circles. Sailor Shift’s direct impact seems to be more focused on albums than individual artists. Yellow nodes, representing Oceanus Folk works, are clearly concentrated in certain areas of the network.

In contrast, Ivy Echos—as a group—shows greater influence over musical groups rather than individual artists, some of whom are also known to have been influenced by Sailor Shift. Compared to Sailor, Ivy Echos’ direct connections demonstrate stronger influence over artists who actively produce works within the Oceanus Folk genre.

As we delve deeper into the network, several important questions begin to emerge: How did these influences evolve over time? What shifts occurred in the structure and spread of Oceanus Folk? And more importantly, which individuals or groups were most significantly shaped by the genre?

Understanding these aspects is essential to grasp not only the **temporal dynamics** of influence—such as whether it surged in particular years or plateaued—but also to uncover the **pathways through which Oceanus Folk expanded** its reach. This includes examining whether its growth was driven more by key individuals like Sailor Shift or collaborative collectives like Ivy Echos, and how these actors shaped the genre’s identity.

To systematically explore these dimensions, we will structure our analysis around the following three core questions:

1. **How has influence changed over time?**
2. **What structural changes occurred in the influence network?**
3. **Who are the most impacted individuals or groups within the network?**

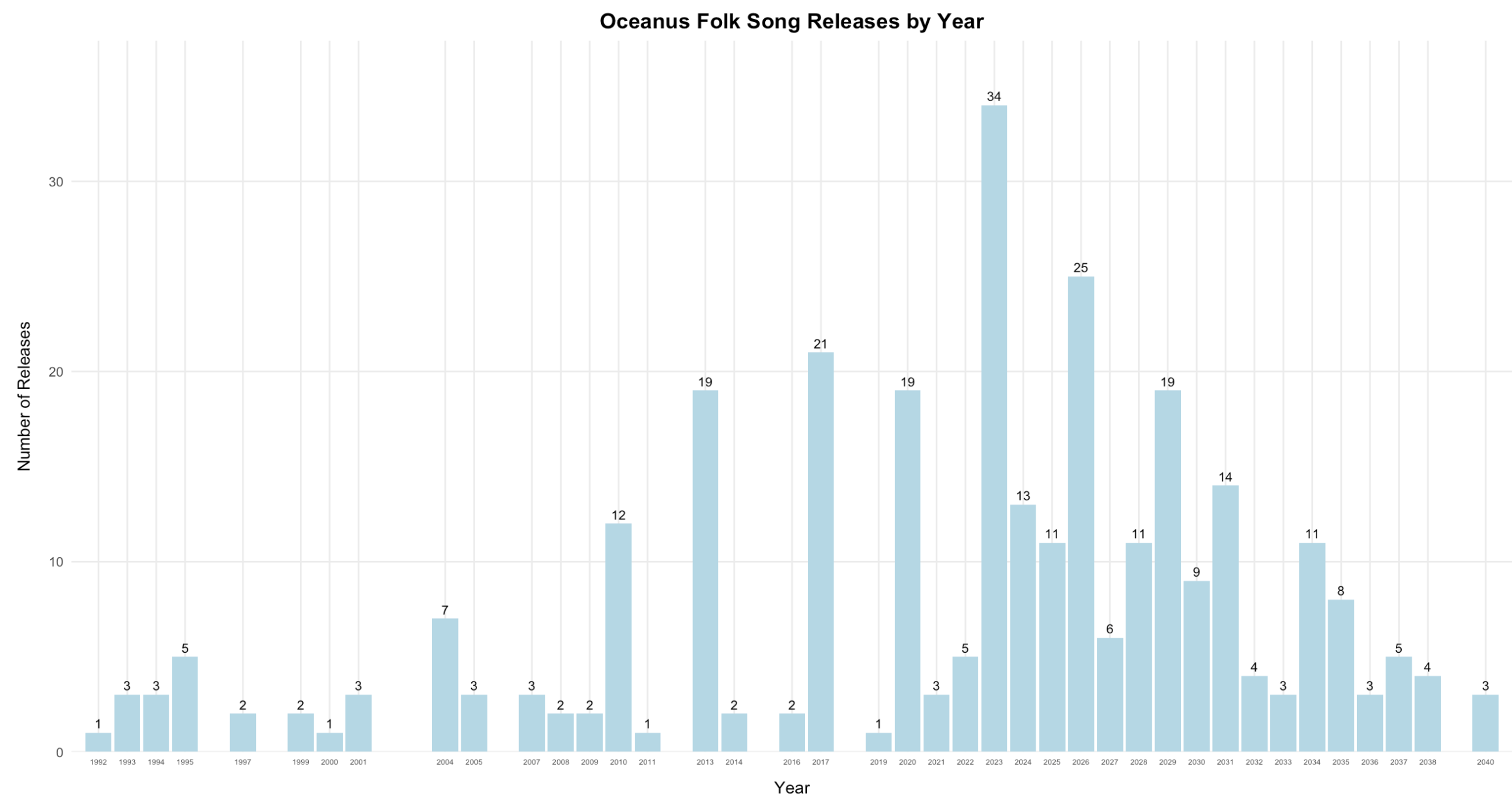
This three-part focus will help reveal not just the presence of influence, but its *depth, direction, and development over time*.

## 6 **Was the influence of Oceanus Folk intermittent or did it have a gradual rise?**

In this part we will incorporate ggplot2 to better show the trends in

### 6.1 Visualizing Trend of Oceanus Folk by their Number of Release Overtime

# Filter song and album with Oceanus Folk genre, and omit those without release date  
oceanus\_musics <- nodes\_tbl %>%  
 filter(genre == "Oceanus Folk",  
 `Node Type` %in% c("Album", "Song"),  
 !is.na(release\_date))  
  
# Count releases by year  
yearly\_counts <- oceanus\_musics %>%  
 count(release\_date) %>%  
 rename(Year = release\_date, Releases = n)  
  
# Visualize the bar chart  
ggplot(yearly\_counts, aes(x = Year, y = Releases)) +  
 geom\_col(fill = "lightblue") +  
 geom\_text(aes(label = Releases), vjust = -0.5, size = 3) +   
 scale\_x\_continuous(breaks = yearly\_counts$Year, expand = expansion(mult = c(0.01, 0.01))) +  
 scale\_y\_continuous(expand = expansion(mult = c(0, 0.1))) +  
 labs(  
 title = "Oceanus Folk Song Releases by Year",  
 x = "Year",  
 y = "Number of Releases"  
 ) +  
 theme\_minimal() +  
 theme(  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 14),  
 axis.title.x = element\_text(margin = margin(t = 10)),  
 axis.title.y = element\_text(margin = margin(r = 10)),  
 axis.text.x = element\_text(size = 5),  
 panel.grid.minor = element\_blank()   
 )



**Insights**

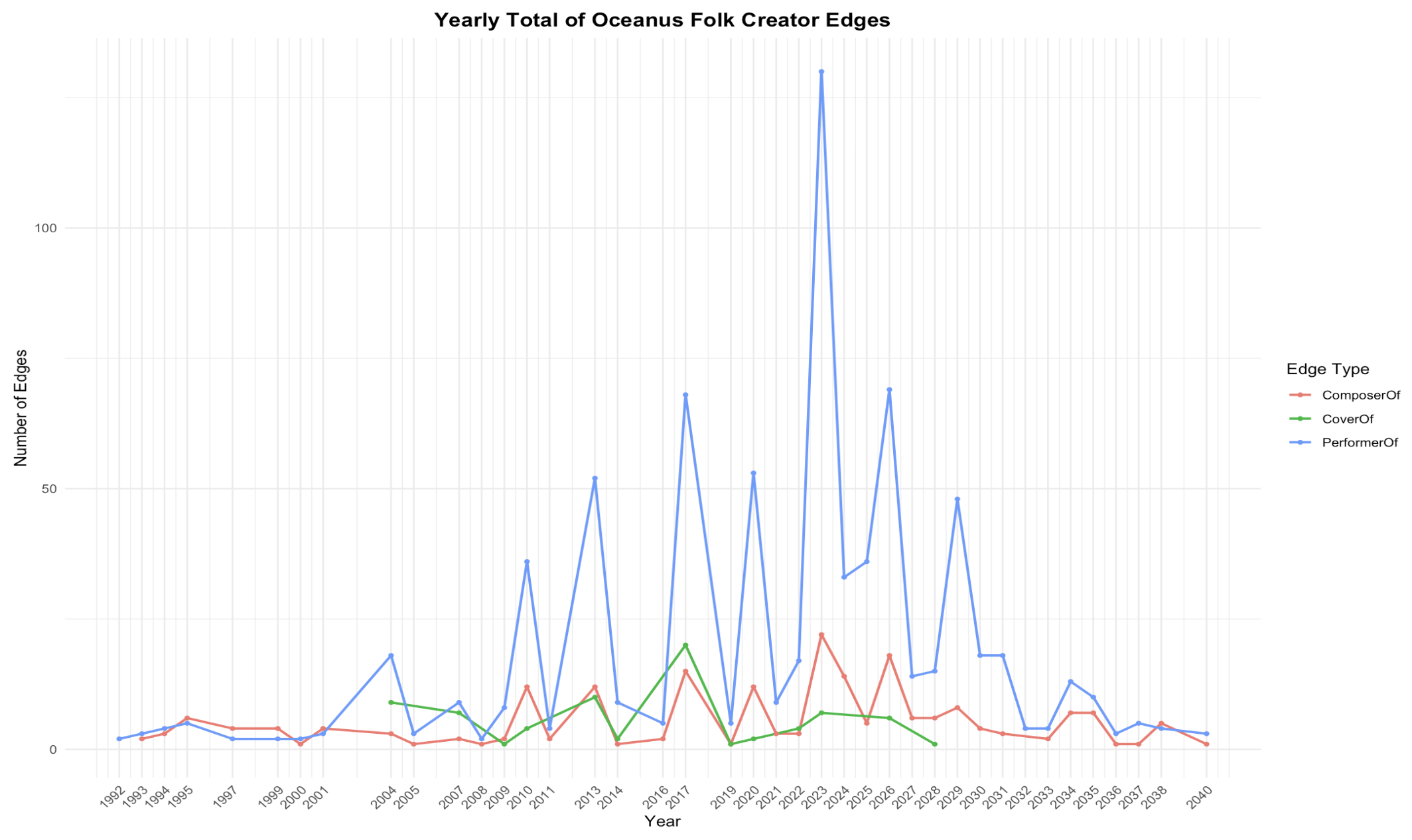
This graph shows the number of songs and albums released under the Oceanus Folk genre over time. We observe that in 2023, there is a significant peak with 34 releases, the highest across all years.

Overall, the trend after 2023 appears to be higher than before 2023, though the difference is not dramatically large. However, it is evident that the release numbers fluctuate from year to year, indicating an unstable pattern. For instance, while 2023 saw the highest release count, other years like 2021, 2018, and 2014 also showed relatively high activity, but there are also years with minimal or no releases.

### 6.2 Visualizing the Popularity of Oceanus Folk Genre

To understand it’s popularity we can also take a look at number of artist or musical group that adopt Oceanus Folk song. Here we consider artist that perform, compose and covering the song with Oceanus Folk genre

# get the id of song and album with oceanus folk genre  
oceanus\_musics\_ids <- oceanus\_musics$id  
  
oceanus\_creators\_summary <- edges\_tbl %>%  
 # Filter edges where target is Oceanus Folk  
 semi\_join(  
 nodes\_tbl %>% filter(genre == "Oceanus Folk") %>% select(id, release\_date),  
 by = c("target" = "id")  
 ) %>%  
 #Filter for specific edge types  
 filter(`Edge Type` %in% c("PerformerOf", "ComposerOf", "CoverOf")) %>%  
 #Add release year  
 left\_join(  
 nodes\_tbl %>% select(id, release\_date),  
 by = c("target" = "id")  
 ) %>%  
 mutate(year = str\_extract(release\_date, "^\\d{4}") %>% as.integer()) %>%  
 #Group and count  
 group\_by(year, `Edge Type`) %>%  
 summarise(total = n(), .groups = "drop") %>%  
 arrange(year, `Edge Type`)  
  
  
# Visualize the Trend  
ggplot(oceanus\_creators\_summary, aes(x = year, y = total, color = `Edge Type`)) +  
 geom\_line(size = 0.8) +  
 geom\_point(size = 1) +  
 labs(  
 title = "Yearly Total of Oceanus Folk Creator Edges",  
 x = "Year",  
 y = "Number of Edges",  
 color = "Edge Type"  
 ) +  
 theme\_minimal() +  
 scale\_x\_continuous(breaks = unique(oceanus\_creators\_summary$year)) +  
 theme(  
 axis.text.x = element\_text(angle = 45, hjust = 1),  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 14) # 👈 bold and center  
 )



**Insights**

The overall trend indicates that artist adoption of the Oceanus Folk genre began to grow steadily from the 2010s, reaching its peak in 2023 before showing signs of decline. The number of cover versions remains relatively low compared to performances of Oceanus Folk works, which may suggest that artists are more engaged in original interpretations rather than reinterpreting existing pieces.

While the number of composers contributing to Oceanus Folk was slightly higher between the 2010s and 2026, the difference is not particularly substantial.

## 7 **What genres and top artists have been most influenced by Oceanus Folk?**

### 7.1 Visualizing Top Genres that have been most influenced by Oceanus Folk

#### 7.1.1 Overview of songs network

This network visualization aims to explore how Oceanus Folk has influenced other songs and genres through direct and indirect musical connections. By starting from Oceanus Folk works and expanding two levels outward, we can identify the spread of its stylistic influence over time and across different genres.

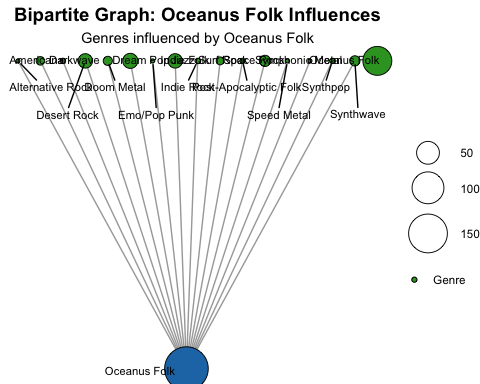
# Filter: Influence Edges Only  
influence\_types <- c("InStyleOf", "InterpolatesFrom", "CoverOf", "LyricalReferenceTo", "DirectlySamples")  
  
edges\_influence <- edges\_tbl %>%  
 filter(`Edge Type` %in% influence\_types)  
  
# Get id of Center Node: Oceanus Folk  
center\_nodes <- nodes\_tbl %>%  
 filter(genre == "Oceanus Folk", `Node Type` %in% c("Song", "Album")) %>%  
 pull(id)  
  
# First degree  
edges\_lvl1 <- edges\_influence %>%  
 filter(target %in% center\_nodes)  
  
ids\_lvl1 <- unique(c(edges\_lvl1$source))  
  
# Second degree  
edges\_lvl2 <- edges\_influence %>%  
 filter(target %in% ids\_lvl1)  
  
ids\_lvl2 <- unique(c(edges\_lvl2$source))  
  
# Combine all relevant node IDs  
all\_ids <- unique(c(center\_nodes, ids\_lvl1, ids\_lvl2))  
  
# Filter Final Nodes and Edges for the Graph  
  
nodes\_viz <- nodes\_tbl %>%  
 filter(id %in% all\_ids) %>%  
 mutate(id = as.character(id)) # visNetwork needs character IDs  
  
edges\_viz <- edges\_influence %>%  
 filter(source %in% all\_ids & target %in% all\_ids) %>%  
 mutate(from = as.character(source),  
 to = as.character(target),  
 color = case\_when(  
 `Edge Type` == "InStyleOf" ~ "#66c2a5",  
 `Edge Type` == "InterpolatesFrom" ~ "#ffd92f",  
 `Edge Type` == "CoverOf" ~ "#fc8d62",  
 `Edge Type` == "LyricalReferenceTo" ~ "#e78ac3",  
 `Edge Type` == "DirectlySamples" ~ "#8da0cb"  
 ),  
 title = paste0("<b>Type:</b> ", `Edge Type`,  
 "<br><b>From:</b> ", from,  
 "<br><b>To:</b> ", to))  
  
  
tagList(  
 tags$h2("Overview of Songs Influenced by Oceanus Folk",   
 style = "text-align:center; font-weight:bold; margin-bottom:20px;"),  
   
 visNetwork(nodes\_viz, edges\_viz, width = "100%") %>%  
 visNodes(color = list(  
 background = node\_colors[nodes\_viz$`Node Colour`],  
 border = "black"  
 )) %>%  
 visEdges(smooth = FALSE) %>%  
 visOptions(highlightNearest = list(enabled = TRUE, degree = 1),  
 nodesIdSelection = TRUE) %>%  
 visIgraphLayout(layout = "layout\_with\_fr") %>%  
 visLegend(useGroups = FALSE, addEdges = data.frame(  
 label = names(edge\_colors),  
 color = unname(edge\_colors)  
 ))  
)

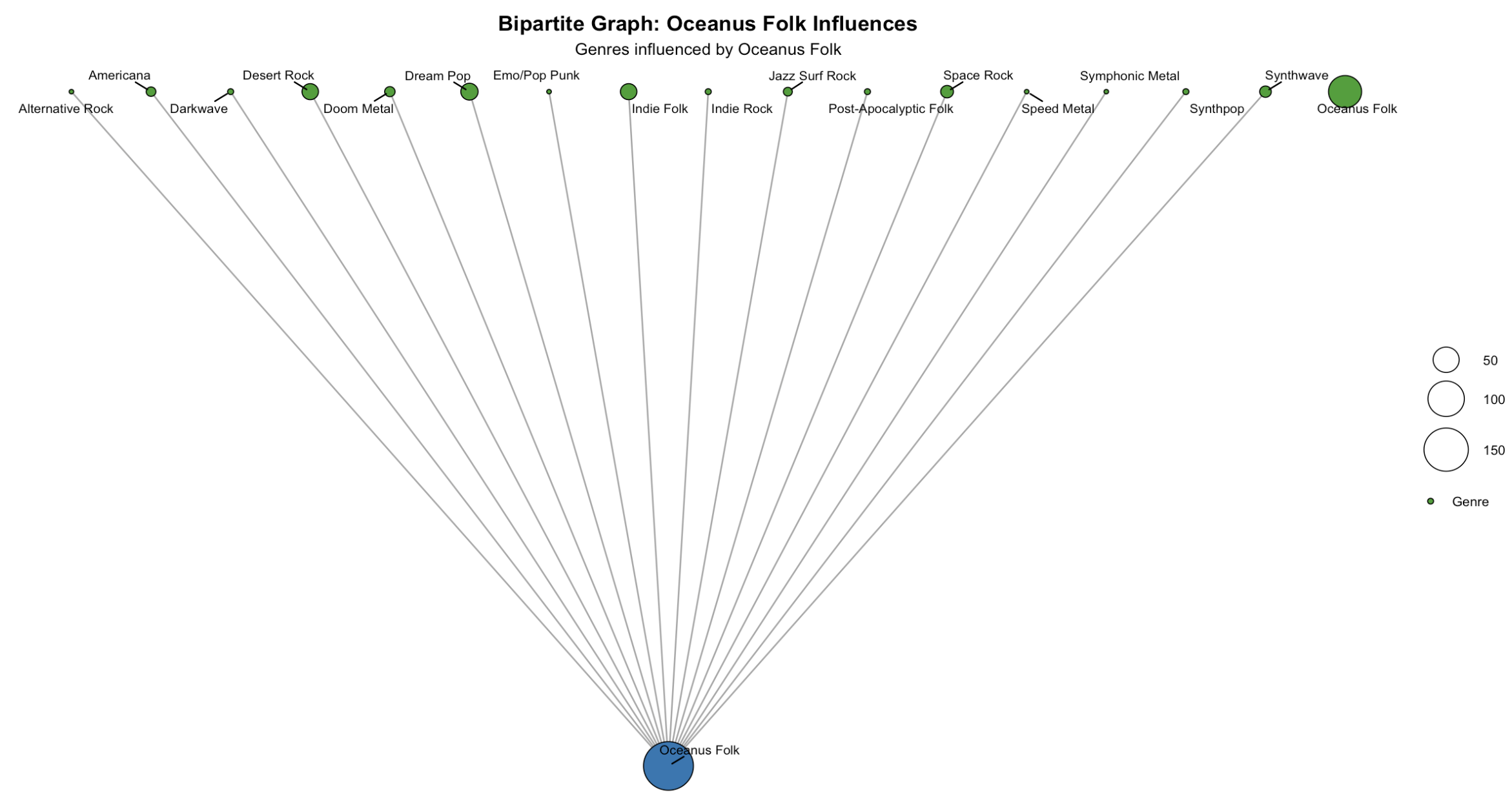
### User Guide

Each node represents a song or album, while edges show the type of influence—such as interpolation, stylistic inspiration, or sampling. You can hover over or click on any node to see its connections, or use the dropdown to search for a specific work. The arrows indicate the direction of influence: who influenced whom. This helps reveal both the reach and nature of Oceanus Folk’s impact across the musical landscape.

#### 7.1.2 Visualizing amount of genre (song) influenced by Oceanus folk

influence\_types <- c("InStyleOf", "InterpolatesFrom", "CoverOf", "LyricalReferenceTo", "DirectlySamples")  
  
# Filter Influences to Oceanus Folk Songs  
edges\_labeled <- edges\_tbl %>%  
 rename(relation = `Edge Type`)  
  
edges\_to\_oceanus <- edges\_labeled %>%  
 filter(relation %in% influence\_types) %>%  
 semi\_join(  
 nodes\_tbl %>%  
 filter(`Node Type` == "Song", genre == "Oceanus Folk") %>%  
 select(id),  
 by = c("target" = "id")  
 )  
  
# Get influenced songs and their genres  
songs\_influenced\_by\_oceanus <- nodes\_tbl %>%  
 filter(id %in% edges\_to\_oceanus$source, `Node Type` == "Song") %>%  
 select(song\_id = id, genre)  
  
# Count genre influence  
genre\_counts <- songs\_influenced\_by\_oceanus %>%  
 count(genre, name = "count")  
  
# Nodes: Oceanus Folk + genres with logical 'type' column  
nodes\_bipartite <- bind\_rows(  
 tibble(id = "Oceanus Folk", type = TRUE, role = "central", count = sum(genre\_counts$count)),  
 tibble(id = genre\_counts$genre, type = FALSE, role = "genre", count = genre\_counts$count)  
)  
  
# Edges: from Oceanus Folk to genres  
edges\_bipartite <- tibble(  
 from = "Oceanus Folk",  
 to = genre\_counts$genre  
)  
  
# Create and plot the bipartite graph  
graph\_bipartite <- tbl\_graph(nodes = nodes\_bipartite, edges = edges\_bipartite, directed = TRUE)  
  
ggraph(graph\_bipartite, layout = "bipartite") +  
 geom\_edge\_link(alpha = 0.7, color = "gray50") +  
 scale\_edge\_width(range = c(0.5, 2)) +  
 geom\_node\_point(aes(size = count, fill = role), shape = 21, color = "black") +  
 scale\_fill\_manual(  
 values = c(central = "#1f78b4", genre = "#33a02c"),  
 breaks = c("genre"), # 👈 Only show "genre" in legend  
 labels = c("Genre")  
 ) +  
 geom\_node\_text(aes(label = id),repel = TRUE,size = 3, force = 2, max.overlaps = Inf,   
 box.padding = 0.5, point.padding = 0.3  
)+  
 scale\_size\_area(max\_size = 15) +  
 theme\_void() +  
 labs(  
 title = "Bipartite Graph: Oceanus Folk Influences",  
 subtitle = "Genres influenced by Oceanus Folk"  
 ) +  
 theme(  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 14),  
 plot.subtitle = element\_text(hjust = 0.5),  
 legend.title = element\_blank()  
 )

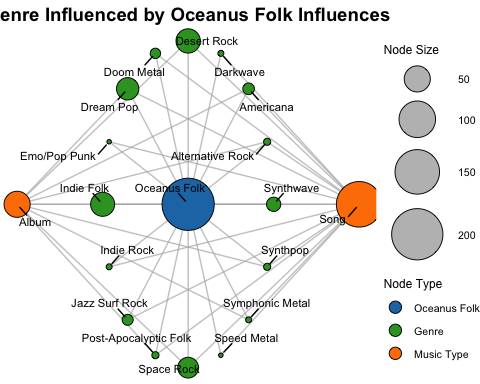




**Insights:**

The graph reveals that a few genres—such as Dessert Rock, Emo/Pop Punk, Indie Folk, Space Rock, and Americana—stand out as the primary recipients of Oceanus Folk’s influence. In contrast, most other genres exhibit relatively consistent and lower levels of influence.

influenced\_info <- edges\_to\_oceanus %>%  
 left\_join(nodes\_tbl %>% select(id, genre, `Node Type`), by = c("source" = "id")) %>%  
 filter(!is.na(genre), genre != "Oceanus Folk", !is.na(`Node Type`)) %>%  
 group\_by(genre, `Node Type`) %>%  
 summarise(count = n(), .groups = "drop")  
  
  
node\_data <- bind\_rows(  
 tibble(id = "Oceanus Folk", type = "center", count = sum(influenced\_info$count)),  
 influenced\_info %>% distinct(genre) %>% mutate(id = genre, type = "genre") %>%  
 left\_join(influenced\_info %>% group\_by(genre) %>% summarise(count = sum(count)), by = "genre"),  
 influenced\_info %>% distinct(`Node Type`) %>% rename(id = `Node Type`) %>%  
 mutate(type = "node\_type") %>%  
 left\_join(influenced\_info %>% group\_by(`Node Type`) %>% summarise(count = sum(count)), by = c("id" = "Node Type"))  
)  
  
edges\_level1 <- influenced\_info %>%  
 group\_by(genre) %>%  
 summarise(weight = sum(count), .groups = "drop") %>%  
 transmute(from = "Oceanus Folk", to = genre, weight = weight)  
  
edges\_level2 <- influenced\_info %>%  
 transmute(from = genre, to = `Node Type`, weight = count)  
  
all\_edges <- bind\_rows(edges\_level1, edges\_level2)  
  
  
genres <- influenced\_info$genre %>% unique()  
node\_types <- influenced\_info$`Node Type` %>% unique()  
  
angle\_genres <- seq(0, 2\*pi, length.out = length(genres)+1)[-1]  
angle\_types <- seq(0, 2\*pi, length.out = length(node\_types)+1)[-1]  
  
layout\_positions <- tibble(  
 id = c("Oceanus Folk", genres, node\_types),  
 x = c(0, cos(angle\_genres), 2 \* cos(angle\_types)),  
 y = c(0, sin(angle\_genres), 2 \* sin(angle\_types))  
)  
  
network\_graph <- tbl\_graph(nodes = node\_data, edges = all\_edges, directed = FALSE)  
layout\_custom <- create\_layout(network\_graph, layout = "manual", x = layout\_positions$x, y = layout\_positions$y)  
  
  
ggraph(layout\_custom) +  
 geom\_edge\_link(color = "gray70", alpha = 0.7) +  
 scale\_edge\_width(range = c(0.3, 2)) +  
   
 geom\_node\_point(aes(size = count, fill = type), shape = 21, color = "black") +  
 scale\_fill\_manual(  
 values = c(center = "#1f78b4", genre = "#33a02c", node\_type = "#ff7f00"),  
 labels = c("Oceanus Folk", "Genre", "Music Type"),  
 breaks = c("center", "genre", "node\_type"),  
 guide = guide\_legend(override.aes = list(size = 4), title = "Node Type", title.position = "top")  
 ) +  
   
 scale\_size\_area(max\_size = 18, guide = guide\_legend(title = "Node Size", title.position = "top", override.aes = list(fill = "gray"))) +  
  
 geom\_node\_text(  
 aes(label = id),  
 repel = TRUE,  
 size = 3,  
 force = 2,  
 box.padding = 0.6,  
 point.padding = 0.3  
 ) +  
   
 labs(  
 title = "Genre Influenced by Oceanus Folk Influences"  
 ) +  
   
 theme\_void() +  
 theme(  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 14),  
 plot.subtitle = element\_text(hjust = 0.5, size = 11),  
 legend.title = element\_text(size = 9),  
 legend.text = element\_text(size = 8),  
 legend.position = "right"  
 )



**Insights:**

Both graphs convey similar insights; however, when examining the types of musical works, it becomes evident that **songs are influenced more frequently than albums**, as reflected in their higher counts.

### 7.2 Visualizing Top artist that have been most influenced by Oceanus Folk

oceanus\_songs <- nodes\_tbl %>%  
 filter(`Node Type` == "Song", genre == "Oceanus Folk") %>%  
 pull(id) %>%  
 as.character()  
  
# Find Songs Influenced by Oceanus Folk Songs  
influenced\_edges <- edges\_tbl %>%  
 filter(`Edge Type` %in% influence\_types, target %in% oceanus\_songs)  
  
influenced\_songs <- influenced\_edges %>%  
 pull(source) %>%  
 unique() %>%  
 as.character()  
  
# Filter for Notable Influenced Songs  
notable\_songs <- nodes\_tbl %>%  
 filter(id %in% influenced\_songs, notable == TRUE, `Node Type` == "Song") %>%  
 pull(id) %>%  
 as.character()  
  
# Link Notable Songs to Producers  
creator\_edges <- edges\_tbl %>%  
 filter(`Edge Type` == "ProducerOf", target %in% notable\_songs)  
  
# Count Influenced Notable Songs per Artist  
artist\_scores <- creator\_edges %>%  
 group\_by(source) %>%  
 summarise(notable\_influenced\_songs = n(), .groups = "drop") %>%  
 arrange(desc(notable\_influenced\_songs)) %>%  
 slice\_head(n = 10)  
  
artist\_scores <- artist\_scores %>%  
 mutate(source = as.character(source))  
  
# Get Artist Node Info  
artist\_nodes <- nodes\_tbl %>%  
 mutate(id = as.character(id)) %>%  
 filter(id %in% artist\_scores$source) %>%  
 select(id, name, role, genre)  
  
# Build Node Table  
node\_data <- bind\_rows(  
 tibble(  
 id = "0",  
 name = "Oceanus Folk",  
 count = sum(artist\_scores$notable\_influenced\_songs),  
 role = NA,  
 genre = NA  
 ),  
 artist\_nodes %>%  
 left\_join(artist\_scores, by = c("id" = "source")) %>%  
 mutate(type = "artist", count = notable\_influenced\_songs)  
)  
  
   
# Build Edge Table (Oceanus Folk ➝ Artist)  
edge\_data <- artist\_scores %>%  
 transmute(  
 from = "0",  
 to = source,  
 weight = notable\_influenced\_songs  
 ) %>%  
 filter(to %in% node\_data$id)   
  
nodes\_data <- node\_data %>%  
 select(id, name, count, role)  
  
nodes\_data <- nodes\_data %>%  
 mutate(id = as.integer(id))  
  
nodes\_data <- nodes\_data %>%  
 mutate(`Node Shape` = ifelse(role == "Person", "triangle", "dot"))  
  
edge\_data <- edge\_data%>%  
 mutate(from = as.integer(from))  
  
nodes\_vis <- nodes\_data %>%  
 mutate(  
 label = name, # 👈 this is what will be shown on the graph  
 value = count, # node size  
 shape = ifelse(`Node Shape` == "triangle", "triangle", "dot") # optional shape logic  
 ) %>%  
 select(id, label, value, shape)  
  
  
  
# Edge table (same edge\_data)  
edges\_vis <- edge\_data %>%  
 select(from, to, weight) %>%  
 mutate(arrows = "to")  
  
# ️ Visualize  
visNetwork(nodes\_vis, edges\_vis) %>%  
 visNodes(shape = "dot", scaling = list(min = 10, max = 30)) %>%  
 visEdges(smooth = FALSE, color = list(color = "gray")) %>%  
 visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE) %>%  
 visLayout(randomSeed = 42)

**Insights**

To begin with, we define **top artists** as those whose songs have been marked as **notable**—indicating high relevance or influence. From this set, we identify the **top 10 artists** based on the number of notable songs they’ve produced.

These artists span **different time periods**, suggesting that Oceanus Folk’s influence is **not concentrated in a single era**, but rather **persisted and evolved across decades**. For example:

* **Leishen (2020–2030)** and **Xiuying Mo (2023–2029)** represent the **recent wave** of Oceanus Folk adopters.
* **Xia Zhu (2017–2030)** bridges both mid and recent phases, with notable activity peaking post-2023.
* In contrast, **Fillipo Pelli (2004–2017)** and **Nicola Bernetti (2004–2013)** reflect the **earlier generation**.
* **Guiying Jin (2010–2028)** illustrates **long-term engagement**, with influence sustained over nearly two decades.

With this finding, we can conclude that:

* **Oceanus Folk’s influence is intergenerational**: notable artists emerged in the early 2000s, 2010s, and continue into the 2030s.
* The genre has a **long-lasting impact**, attracting new artists while remaining relevant to established ones.
* Recent peaks (e.g. post-2020) may indicate a **revival or re-emergence** of Oceanus Folk influence, possibly tied to new stylistic movements or cultural shifts.

## 8 On **the converse, how has Oceanus Folk changed with the rise of Sailor Shift? From which genres does it draw most of its contemporary inspiration?**

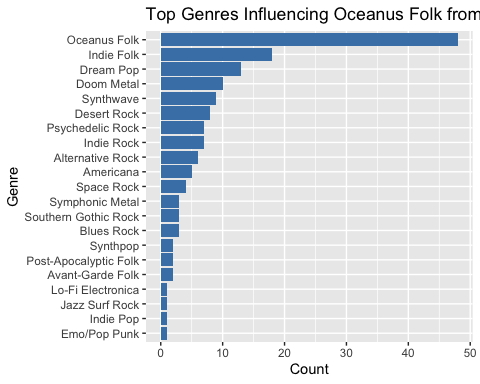
for this part i will set they year to 2021 which Sailor start her music journey with her team and continue with her solo in 2026

This section focuses on **how the *style, themes, and network* of Oceanus Folk evolved** after Sailor Shift entered the scene. Here are key ideas to explore:

### 8.1 Genre that bring contemporary inspiration to Oceanus Folk

In this first section, we explore the genres that have influenced **Oceanus Folk** from 2023 (year of the rise of Ivy Echos)

#| code-fold: true  
  
  
# Filter for Oceanus Folk songs/albums created in the last 2 years  
recent\_oceanus <- nodes\_tbl %>%  
 filter(genre == "Oceanus Folk", `Node Type` %in% c("Song", "Album")) %>%  
 filter(release\_date >= 2023) %>%  
 pull(id)  
  
# Find Influences: who influenced these recent works  
influences <- edges\_tbl %>%  
 filter(`Edge Define` %in% c("Influenced By"), source %in% recent\_oceanus)  
  
# Get the source nodes (influencers)  
influencer\_ids <- influences$target  
  
# Match influencer genres  
influencer\_genres <- nodes\_tbl %>%  
 filter(id %in% influencer\_ids, !is.na(genre)) %>%  
 count(genre, sort = TRUE)  
  
# Visualize  
ggplot(influencer\_genres, aes(x = reorder(genre, n), y = n)) +  
 geom\_col(fill = "steelblue") +  
 coord\_flip() +  
 labs(title = "Top Genres Influencing Oceanus Folk from 2023 Onwards", x = "Genre", y = "Count")

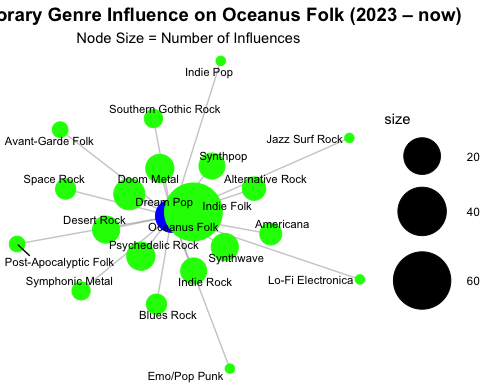


**Insights:**

While Oceanus Folk continues to influence itself through internal evolution, several other genres have also played a significant role in shaping its recent direction. Notably, **Indie Folk**, **Dream Pop**, **Doom Metal**, **Synthwave**, and **Desert Rock** have emerged as key contributors—highlighting a blend of ambient, alternative, and heavy stylistic elements that now inform the contemporary sound of Oceanus Folk.

To better understand how **recent Oceanus Folk works (2023 onward)** have been influenced by other genres, this visualization provides a **clear, centralized network** showing the volume of genre-level influence. Rather than just listing genre counts, this format makes it easier to **see which genres have the strongest connections** to Oceanus Folk and how influence spreads visually. The node size gives immediate feedback on which genres had the most impact.

#| code-fold: true  
  
# Get recent Oceanus Folk works (2023+)  
oceanus\_recent <- nodes\_tbl %>%  
 filter(genre == "Oceanus Folk", `Node Type` %in% c("Song", "Album"), release\_date >= 2023)  
  
# Get influence edges using musical influence types  
influenced\_types <- c("InStyleOf", "InterpolatesFrom", "CoverOf", "LyricalReferenceTo", "DirectlySamples")  
  
influenced\_edges <- edges\_tbl %>%  
 filter(`Edge Type` %in% influenced\_types, source %in% oceanus\_recent$id)  
  
# Join with influencer genres  
influenced\_with\_genre <- influenced\_edges %>%  
 left\_join(nodes\_tbl %>% select(id, genre), by = c("target" = "id")) %>%  
 filter(!is.na(genre))  
  
# Count number of influences per genre  
genre\_counts <- influenced\_with\_genre %>%  
 filter(genre != "Oceanus Folk") %>%   
 count(genre, name = "influence\_count")  
  
  
# Define nodes  
central\_node <- tibble(id = "Oceanus Folk", label = "Oceanus Folk", layer = 1, color = "blue", size = 15)  
  
genre\_nodes <- genre\_counts %>%  
 mutate(  
 id = genre,  
 label = genre,  
 layer = 2,  
 color = "green",  
 size = influence\_count   
 )  
  
# Build node and edge tables  
all\_nodes <- bind\_rows(central\_node, genre\_nodes) %>%  
 distinct(id, .keep\_all = TRUE)  
  
edges <- genre\_counts %>%  
 transmute(from = "Oceanus Folk", to = genre, weight = influence\_count)  
  
# Plot the graph  
library(tidygraph)  
library(ggraph)  
  
graph <- tbl\_graph(nodes = all\_nodes, edges = edges, directed = TRUE)  
  
ggraph(graph, layout = 'fr') +  
 geom\_edge\_link(color = "gray60", alpha = 0.5, width = 0.5) +  
 geom\_node\_point(aes(color = color, size = size)) +  
 geom\_node\_text(aes(label = label), repel = TRUE, size = 3) +  
 scale\_color\_identity() +  
 scale\_size(range = c(3, 20)) +  
 theme\_void() +  
 labs(  
 title = "Contemporary Genre Influence on Oceanus Folk (2023 – now)",  
 subtitle = "Node Size = Number of Influences"  
 ) +  
 theme(  
 plot.title = element\_text(hjust = 0.5, face = "bold", size = 14),  
 plot.subtitle = element\_text(hjust = 0.5, size = 11)  
 )



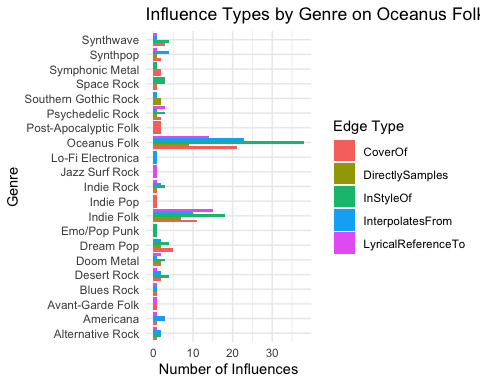
Insights:

It is immediately apparent that **Indie Folk** is the most prominent genre influencing Oceanus Folk in the recent period, followed by **Dream Pop**, **Doom Metal**, **Synthwave**, and **Desert Rock**. These genres collectively reflect a mix of atmospheric, experimental, and emotionally expressive styles, suggesting that **contemporary Oceanus Folk is evolving by absorbing textures from ambient, rock-adjacent, and niche subcultures**.

While Oceanus Folk also draws from itself (self-reinforcing evolution), the diversity of surrounding genres highlights its **openness to cross-genre inspiration** in recent years.

### 8.2 Oceanus Folk Influencing Types following the rise of Sailor shift

#| code-fold: true  
  
  
# Filter Oceanus Folk works from 2023 onward  
oceanus\_recent <- nodes\_tbl %>%  
 filter(genre == "Oceanus Folk", `Node Type` %in% c("Song", "Album"), release\_date >= 2023)  
  
# Filter relevant influence edges from Oceanus Folk works  
influence\_types <- c("InStyleOf", "InterpolatesFrom", "CoverOf", "LyricalReferenceTo", "DirectlySamples")  
  
influenced\_edges <- edges\_tbl %>%  
 filter(`Edge Type` %in% influence\_types, source %in% oceanus\_recent$id)  
  
# Join with genre of the \*target\* (influencer)  
influenced\_with\_genre <- influenced\_edges %>%  
 left\_join(nodes\_tbl %>% select(id, genre), by = c("target" = "id")) %>%  
 filter(!is.na(genre))  
  
#Count by genre and Edge Type  
genre\_edge\_summary <- influenced\_with\_genre %>%  
 count(genre, `Edge Type`, name = "influence\_count") %>%  
 arrange(desc(influence\_count))  
  
# reshape for heatmap or stacked bar  
genre\_edge\_wide <- genre\_edge\_summary %>%  
 pivot\_wider(names\_from = `Edge Type`, values\_from = influence\_count, values\_fill = 0)  
  
#Plot grouped bar chart  
ggplot(genre\_edge\_summary, aes(x = genre, y = influence\_count, fill = `Edge Type`)) +  
 geom\_col(position = "dodge") +  
 coord\_flip() +  
 labs(  
 title = "Influence Types by Genre on Oceanus Folk (2023 onwards)",  
 x = "Genre",  
 y = "Number of Influences",  
 fill = "Edge Type"  
 ) +  
 theme\_minimal()



genre\_edge\_summary <- influenced\_with\_genre %>%  
 count(genre, `Edge Type`, name = "influence\_count") %>%  
 arrange(desc(influence\_count))  
  
  
genre\_edge\_wide <- genre\_edge\_summary %>%  
 pivot\_wider(  
 names\_from = `Edge Type`,  
 values\_from = influence\_count,  
 values\_fill = 0 # fill missing combinations with 0  
 )  
  
genre\_edge\_wide

# A tibble: 21 × 6  
 genre InStyleOf InterpolatesFrom CoverOf LyricalReferenceTo DirectlySamples  
 <chr> <int> <int> <int> <int> <int>  
 1 Oceanu… 38 23 21 14 9  
 2 Indie … 18 10 11 15 7  
 3 Dream … 4 2 5 0 2  
 4 Desert… 4 2 2 1 0  
 5 Synthp… 0 4 2 1 1  
 6 Synthw… 4 1 3 1 0  
 7 Americ… 0 3 1 1 0  
 8 Doom M… 3 1 2 2 2  
 9 Indie … 3 2 1 1 1  
10 Psyche… 3 1 2 3 1  
# ℹ 11 more rows

**Insights**

* **Oceanus Folk influences itself** the most, especially through InStyleOf and InterpolatesFrom, showing internal evolution.
* **Indie Folk** is the strongest external influence, contributing across multiple edge types.
* **Dream Pop, Doom Metal, Synthwave, and Desert Rock** also play key roles, suggesting recent Oceanus Folk draws from ambient, heavy, and experimental styles.
* Many other genres (e.g., **Jazz Surf Rock**, **Lo-Fi Electronica**) contribute lightly but reflect **diverse stylistic inspiration**.
* The **variety of influence types** shows how different genres shape Oceanus Folk in distinct ways—whether through reinterpretation (CoverOf) or stylistic borrowing (InterpolatesFrom, LyricalReferenceTo).

## 9 Summary Findings

As part of our support for journalist Silas Reed’s feature article *“Oceanus Folk: Then-and-Now,”* our comprehensive analysis highlights how the genre evolved from a niche island sound into a global cultural force. At the heart of this transformation is Sailor Shift, whose solo career and foundational role in *Ivy Echos* catalyzed the rise of Oceanus Folk. Through network visualizations, we reveal how Sailor’s influence spread across songs, artists, and albums, with her impact especially strong on album production within the genre. Ivy Echos also played a pivotal role, fostering genre continuity and nurturing talent. Together, they helped Oceanus Folk grow—not just in popularity, but in structural depth, as shown by the genre’s extensive and recurring creative connections in recent years.

The genre’s influence has been both widespread and dynamic. Notably, genres like Indie Folk, Dream Pop, Doom Metal, and Synthwave have increasingly shaped the newer sound of Oceanus Folk since 2023, reflecting a shift toward atmospheric and emotionally layered expressions. While the genre continues to evolve internally—frequently influencing itself through stylistic borrowing—our visual and temporal analyses show a clear expansion of Oceanus Folk’s reach. This insight is vital for readers to understand not just *who* influenced and was influenced by the genre, but *how* it developed over time, across styles, and through different generations of artists.