R Markdown (Figure Merging)

Emma Chapman-Banks

2025-01-17

Part 1: Merging Figures 1 and 4

Load necessary packages

```r
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(janitor)
```

```
##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

First, we will read, clean, and save publication data. This is from Paula's original code

```r
# Get Publication data
df <- read.csv("490k_141224_1234_dedup.csv") #Or change file path to wherever files are located
df <- clean_names(df)

# Remove all excluded records
df <- df |>
  filter(decision == "Include")

# Calc publication records by year
```

```
df_annual <- df |>
  group_by(year) |>
  summarize(n = n()) |>
  rename(n_pub = n)

# Remove records prior to 1900 or after 2025
df_annual <- df_annual |>
  filter(year < 2025) |>
  filter(year > 1900)

df_annual_save <- df_annual
```

Now, we will load in data for policy citations. Again, this is Paula's original code

```
pc <- read.csv("data/overton_results_expanded.csv")
pc <- clean_names(pc)

# Transform date of policy documents to year get year
pc$published_on <- as.Date(pc$published_on)
pc$year <- format(as.Date(pc$published_on, format="%Y-%m/%d"),"%Y")

# Calc policy citations by year
pc_annual <- pc |>
  group_by(year) |>
  summarize(n = n()) |>
  rename(n_cit = n)

# Calc policy citations by org type and subtype
pc_subtype <- pc |>
                group_by(type, subtype) |>
                summarize(n = n()) |>
                rename(n_cit = n)
```

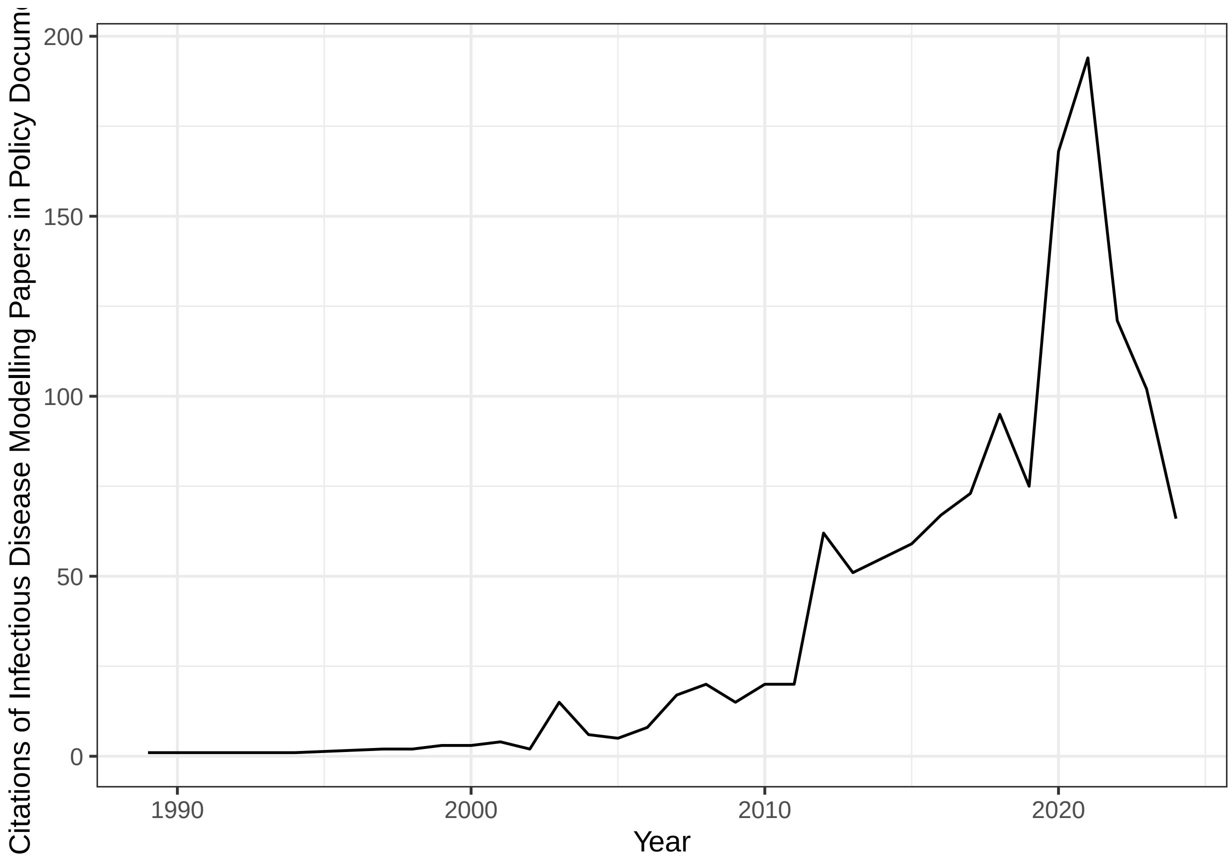
```
## `summarise()` has grouped output by 'type'. You can override using the
## `.groups` argument.
```

```
write.csv(pc_subtype,
          "results/ec_aggregation_by_sub_type.csv",
          row.names = FALSE)
```

Let's first plot the plots separately to see what we are working with. Still Paula's code.

```
#Plot IDM in Policy Documents
ggplot(pc_annual, aes(x = as.numeric(year), y = n_cit)) +
  geom_line() +
  labs(x = "Year", y = "Citations of Infectious Disease Modelling Papers in Policy Documents") +
  theme_bw()
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_line()`).
```



```
# Check the structure and summary of the data
str(pc_annual)
```

```
## tibble [33 x 2] (S3: tbl_df/tbl/data.frame)
##  $ year : chr [1:33] "1989" "1991" "1992" "1994" ...
##  $ n_cit: int [1:33] 1 1 1 1 2 2 3 3 4 2 ...
```

```
summary(pc_annual)
```

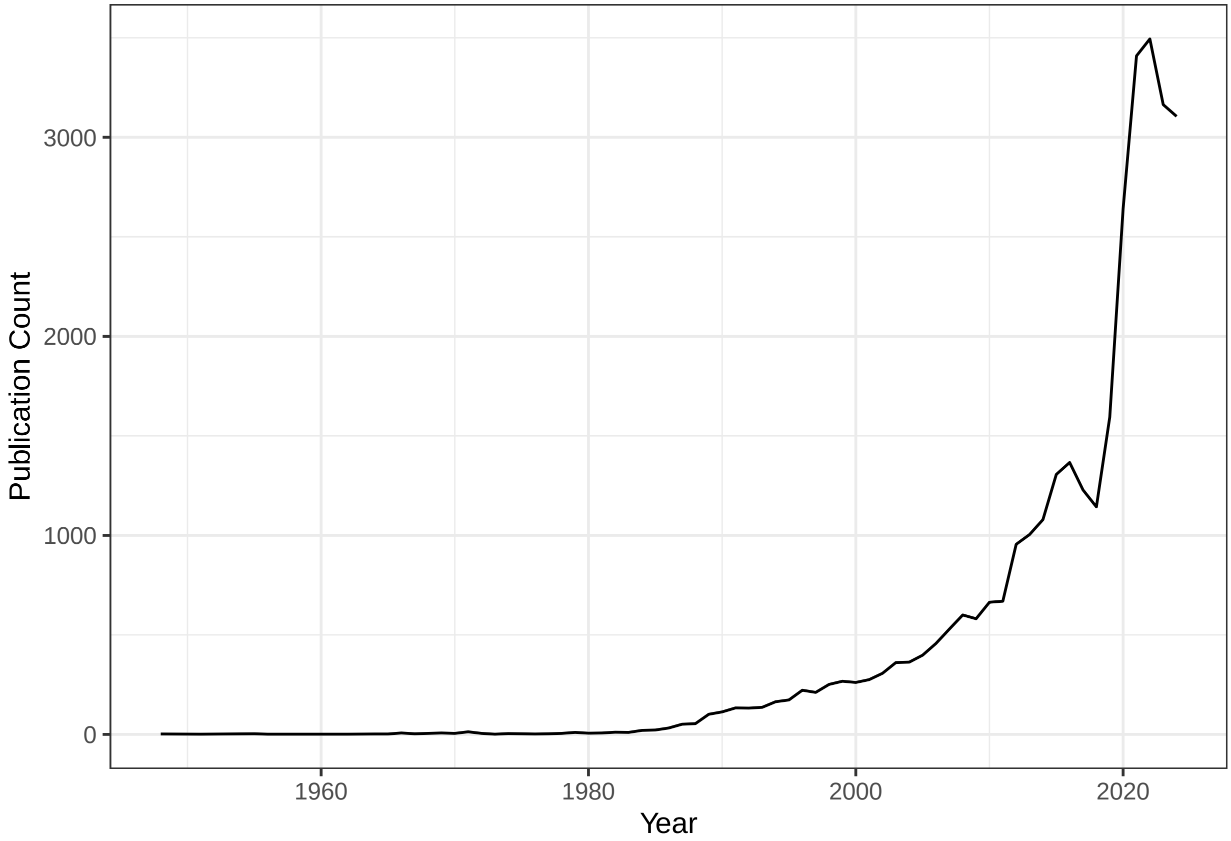
```
##      year               n_cit
##  Length:33          Min.   :  1.00
##  Class :character   1st Qu.:  3.00
##  Mode  :character   Median : 17.00
##                     Mean   : 40.52
##                     3rd Qu.: 66.00
##                     Max.   :194.00
```

```
# Find rows with missing or invalid values
pc_annual |> filter(is.na(n_cit) | n_cit <= 0)
```

```
## # A tibble: 0 x 2
## # i 2 variables: year <chr>, n_cit <int>
```

Plot publication data.

```
#Plot Publication Count
ggplot(df_annual, aes(x = year, y = n_pub)) +
  geom_line() +
  labs(x = "Year", y = "Publication Count") +
  theme_bw()
```



Let's double check the structure of the dataset for publication count too.

```
# Check the structure and summary of the data
str(df_annual)
```

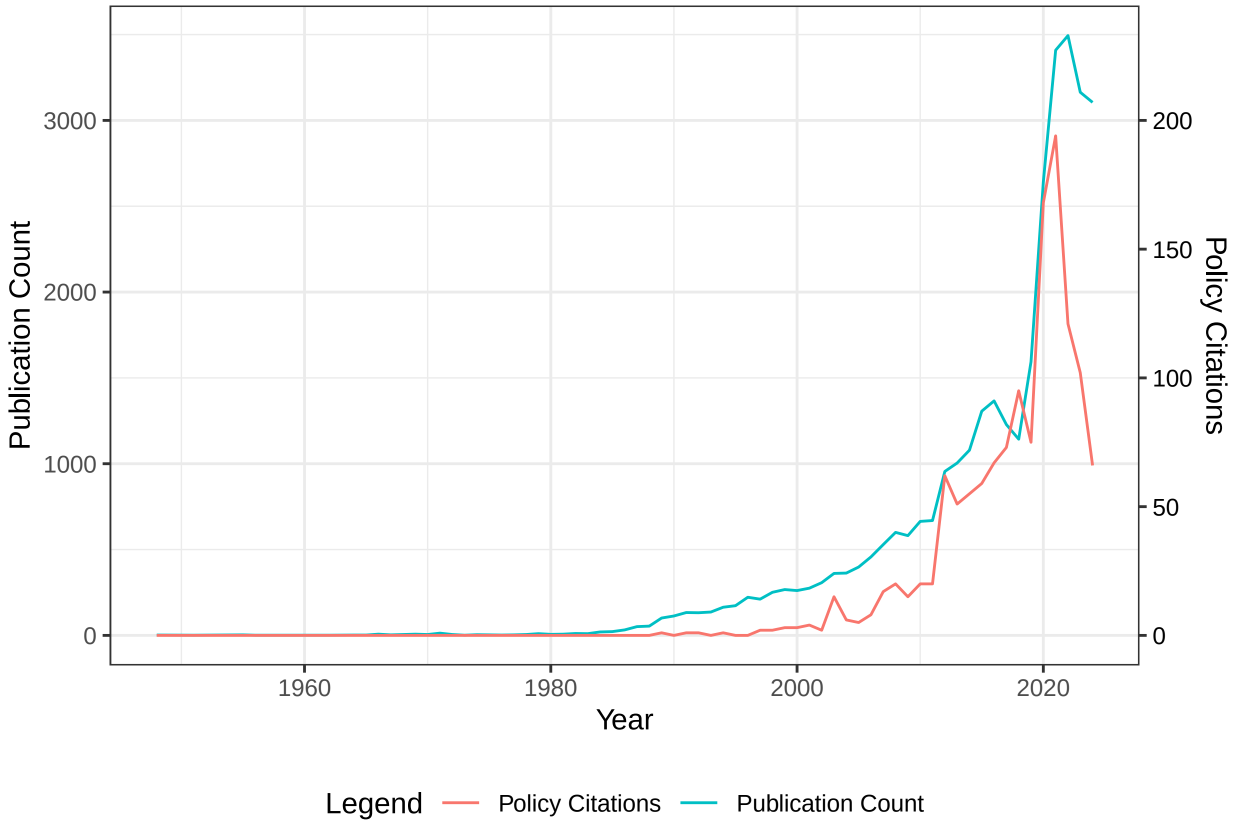
```
## tibble [67 x 2] (S3: tbl_df/tbl/data.frame)
##  $ year : int [1:67] 1948 1951 1955 1956 1961 1962 1964 1965 1966 1967 ...
##  $ n_pub: int [1:67] 2 1 3 1 1 1 2 2 7 3 ...
```

```
summary(df_annual)
```

```
##       year          n_pub       
##  Min.   :1948   Min.   :   1.0  
##  1st Qu.:1974   1st Qu.:   5.0  
##  Median :1991   Median : 132.0  
##  Mean   :1991   Mean   : 494.9  
##  3rd Qu.:2008   3rd Qu.: 555.0  
##  Max.   :2024   Max.   :3494.0  
```

It seems that pc_annual starts from 1989 and df_annual starts from 1948. This can create issues when merging so we must align the data ranges to ensure RStudio can read this correctly.

```r
# Merge datasets, keeping all years from df_annual
combined_df <- merge(df_annual, pc_annual, by = "year", all.x = TRUE)

# Replace missing policy citation values (n_cit) with 0 (as pc_annual doesn't begin until 1989)
combined_df$n_cit[is.na(combined_df$n_cit)] <- 0

# Ensure n_pub is numeric
combined_df$n_pub <- as.numeric(combined_df$n_pub)

# Plot with secondary y-axis
ggplot(combined_df, aes(x = year)) +
  geom_line(aes(y = n_pub, color = "Publication Count")) +
  geom_line(aes(y = n_cit * 15, color = "Policy Citations")) +  # Scale factor for secondary axis
  scale_y_continuous(
    name = "Publication Count",
    sec.axis = sec_axis(~ . / 15, name = "Policy Citations")
  ) +
  labs(x = "Year", color = "Legend") +
  theme_bw() +
  theme(
    axis.title.y.right = element_text(color = "black"),
    axis.text.y.right = element_text(color = "black"),
    legend.position = "bottom"
  )
```



Part 2: Merging all three plots together

Now, we will “superimpose” the publication count/policy citations with the outbreak timeline.

But first, lets load the outbreak timeline data (Paula’s code)

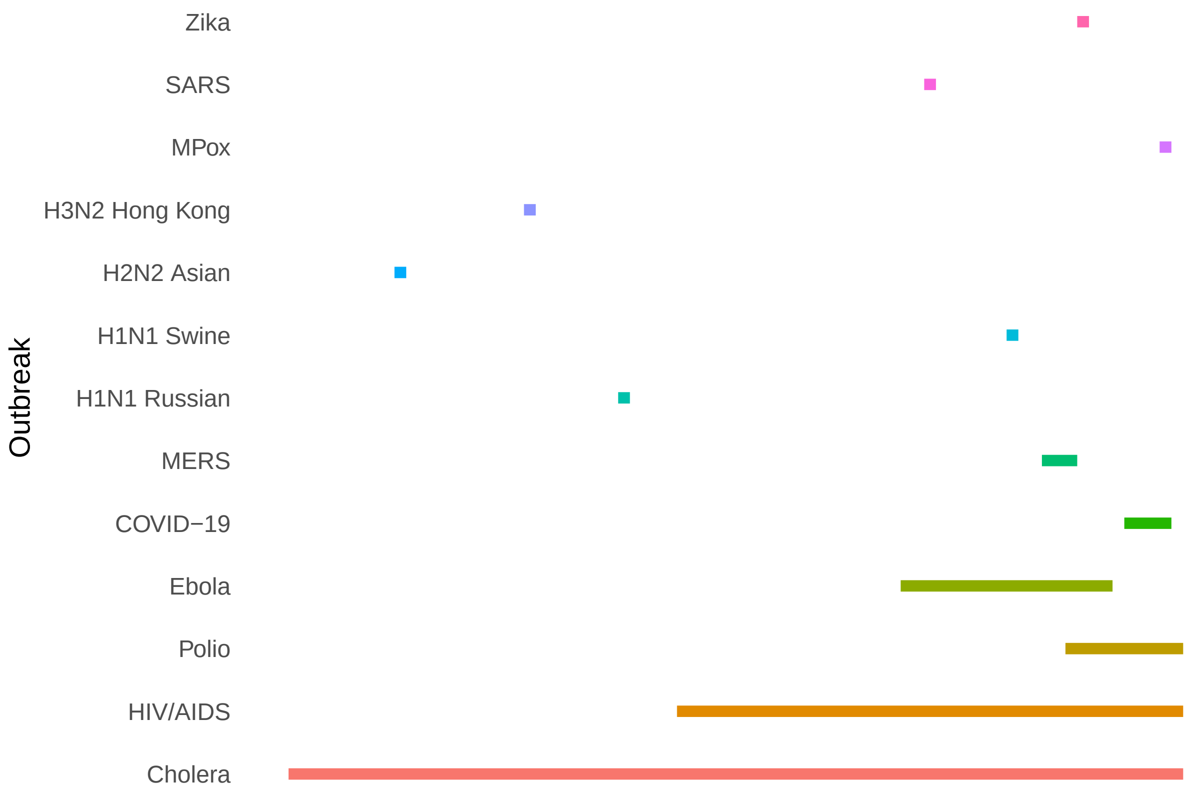
```
# Get Outbreaks timeline data
e_timeline <- readxl::read_excel("data/Epidemics Timeline.xlsx",
                                 sheet = 2)
```

```
## New names:
## * ‘‘ -> ‘...1‘
```

```
names(e_timeline)[1] <- c("year")
e_timeline <- clean_names(e_timeline)

# Reshape data to long format for ggplot2
data_long <- tidyr::pivot_longer(e_timeline,
                                 -year,
                                 names_to = "pathogen",
                                 values_to = "presence")
data_long <- data_long |> filter(presence > 0)
data_long <- data_long |> filter(year >= min(df_annual$year))
```

```r
# Renaming pathogens. This is done to help easier visualisation on the plot
rename_mapping <- c(
  "zika" = "Zika",
  "swine_flu_h1n1_pandemic" = "H1N1 Swine",
  "severe_acute_respiratory_syndrome_sars_coronavirus" = "SARS",
  "russian_flu_pandemic_h1n1" = "H1N1 Russian",
  "m_pox" = "MPox",
  "hong_kong_flu_pandemic_h3n2" = "H3N2 Hong Kong",
  "asian_flu_pandemic_h2n2" = "H2N2 Asian",
  "mers" = "MERS",
  "covid_19" = "COVID-19",
  "ebola" = "Ebola",
  "uptick_in_polio" = "Polio",
  "hiv_aids_pandemic" = "HIV/AIDS",
  "cholera" = "Cholera"
)

data_long$pathogen <- dplyr::recode(data_long$pathogen, !!!rename_mapping)

# Reorder factor levels by frequency
data_long$pathogen <- factor(data_long$pathogen,
                             levels = names(sort(table(data_long$pathogen),
                                                 decreasing = TRUE)))

# Order dataframe by year
data_long <- data_long[order(data_long$year), ]

# Plot Outbreak Timeline
ggplot(data_long, aes(x = year, y = pathogen, color = pathogen)) +
  geom_line(stat = "identity", size = 2) +
  scale_x_continuous(breaks = seq(min(e_timeline$year), max(e_timeline$year), by = 2)) +
  labs(y = "Outbreak") +
  theme_minimal() +
  theme(legend.position = "none",
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.text.x = element_blank(),
        axis.title.x = element_blank())
```

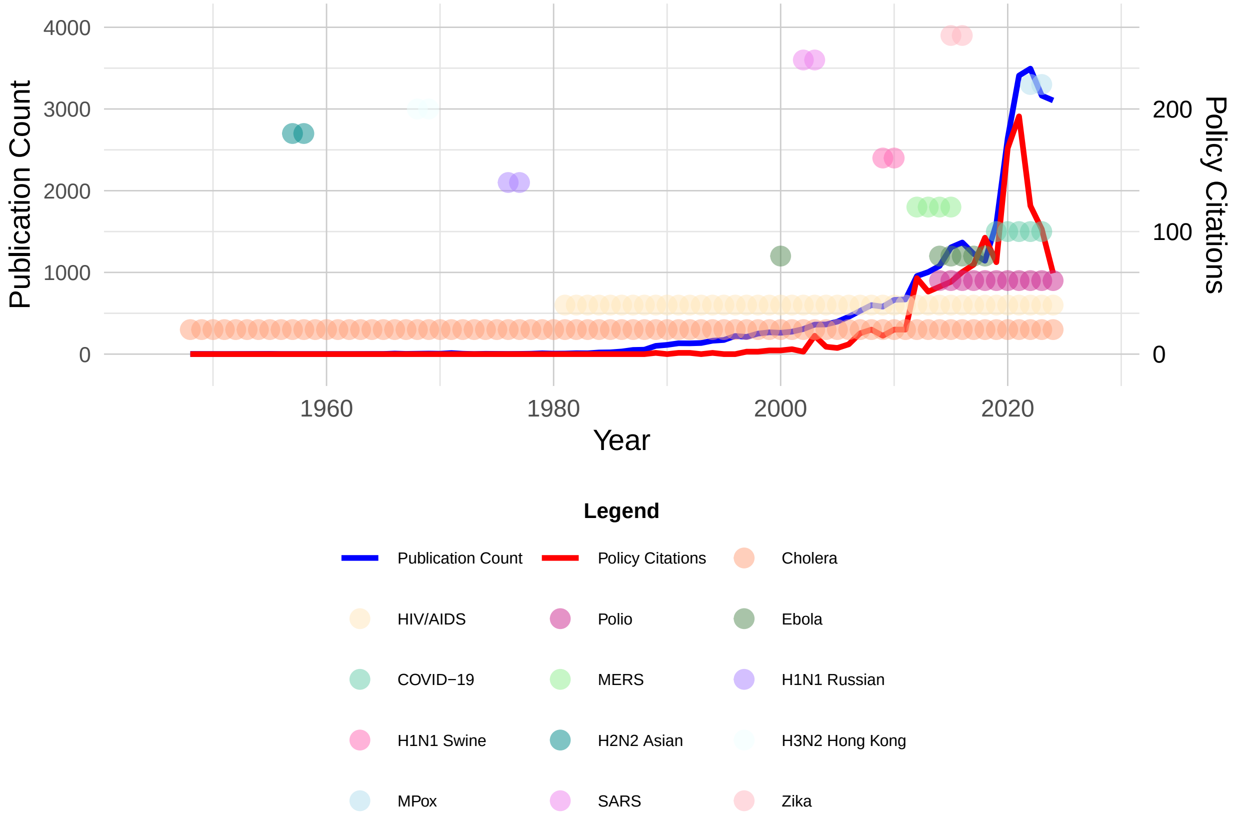
```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```



We will now add the outbreak plot onto the publication/policy plot.

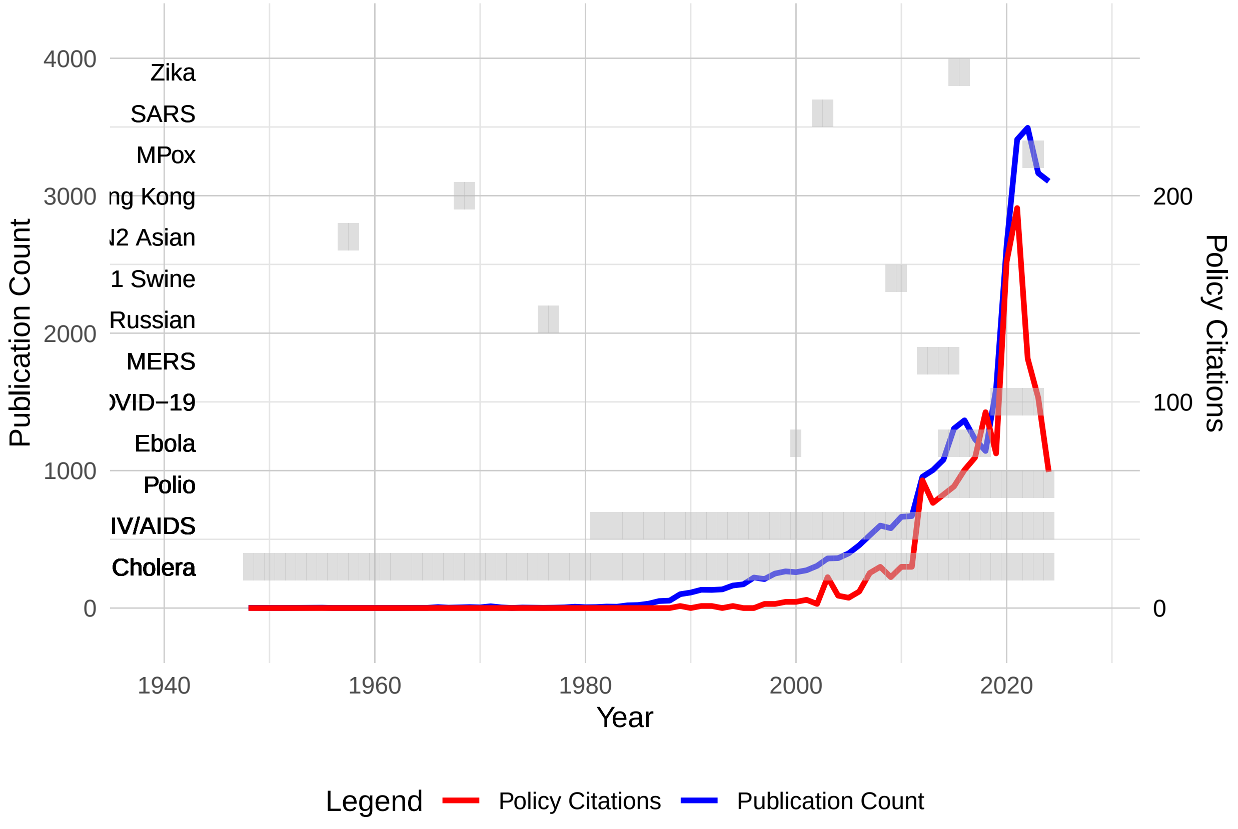
This is the old plot (no disease y-axis with circle dot points)

```r
#Because I had issues with creating the legend. I needed to convert the legend labels into a factor (fo
data_long$pathogen <- factor(data_long$pathogen)
pathogen_breaks <- c("Publication Count", "Policy Citations", levels(data_long$pathogen))

# Create the plot
ggplot(combined_df, aes(x = year)) +
  geom_line(aes(y = n_pub, color = "Publication Count"), size = 1) +
  geom_line(aes(y = n_cit * 15, color = "Policy Citations"), size = 1) +
  geom_point(
    data = data_long,
    aes(x = year, y = as.numeric(as.factor(pathogen)) * 300, color = pathogen),
    size = 3, alpha = 0.5
  ) +
  # Scales
  scale_y_continuous(
    name = "Publication Count",
    sec.axis = sec_axis(~ . / 15, name = "Policy Citations"),
    expand = expansion(mult = c(0.1, 0.1))
  ) +
  scale_x_continuous(expand = expansion(mult = c(0.1, 0.1))) +
  scale_color_manual(
    name = "Legend",
    values = c(
      "Publication Count" = "blue",
      "Policy Citations" = "red",
      "Zika" = "lightpink",
      "H1N1 Swine" = "hotpink",
      "SARS" = "violet",
      "H1N1 Russian" = "mediumpurple1",
      "MPox" = "lightblue2",
      "H3N2 Hong Kong" = "azure",
      "H2N2 Asian" = "cyan4",
      "MERS" = "lightgreen",
      "COVID-19" = "mediumaquamarine",
      "Ebola" = "palegreen4",
      "Polio" = "maroon3",
      "HIV/AIDS" = "wheat1",
      "Cholera" = "lightsalmon"
    ),
    breaks = pathogen_breaks
  ) +
  labs(x = "Year", color = "Legend", y = NULL) +
  theme_minimal() +
  theme(
    legend.position = "bottom",
    legend.title = element_text(size = 8, face = "bold"),
    legend.text = element_text(size = 6),
    axis.title.y.right = element_text(color = "black"),
    axis.text.y.right = element_text(color = "black"),
    axis.text.y.left = element_text(size = 8),
    panel.grid.major = element_line(color = "grey80", linewidth = 0.25),
    panel.grid.minor = element_line(color = "grey90", linewidth = 0.25)
  ) +
  guides(
    color = guide_legend(
      override.aes = list(size = 3),
      nrow = 5,
      byrow = TRUE,
      title.position = "top",
      title.hjust = 0.5
    )
  )
```



This is the updated plot (January 23rd 2025)

```
#Because I had issues with creating the legend. I needed to convert the legend labels into a factor (fo
data_long <- data_long %>%
  mutate(outbreak_y = as.numeric(as.factor(pathogen)))  # Create a unique numeric value for each pathog

ggplot(combined_df, aes(x = year)) +
  geom_line(aes(y = n_pub, color = "Publication Count"), size = 1) +
  geom_line(aes(y = n_cit * 15, color = "Policy Citations"), size = 1) +
  geom_tile(
    data = data_long,
    aes(x = year, y = outbreak_y * 300),
    width = 1, height = 200,
    fill = "grey", alpha = 0.5
  ) +
  # Add outbreak labels with fixed colors (excluded from legend with the inhereit.aes = FALSE)
  geom_text(
    data = data_long,
    aes(x = min(combined_df$year) - 5, y = outbreak_y * 300, label = pathogen),
    hjust = 1, size = 3, inherit.aes = FALSE,
    color = "black"
  ) +
  # Scales and labels
  scale_y_continuous(
    name = "Publication Count",
    sec.axis = sec_axis(~ . / 15, name = "Policy Citations"),
    expand = expansion(mult = c(0.1, 0.1))
) +
scale_x_continuous(expand = expansion(mult = c(0.1, 0.1))) +
scale_color_manual(
  values = c(
    "Publication Count" = "blue",
    "Policy Citations" = "red"
  )
) +
labs(x = "Year", color = "Legend", y = NULL) +
theme_minimal() +
theme(
  legend.position = "bottom",
  axis.title.y.right = element_text(color = "black"),
  axis.text.y.right = element_text(color = "black"),
  panel.grid.major = element_line(color = "grey80", linewidth = 0.25),
  panel.grid.minor = element_line(color = "grey90", linewidth = 0.25)
)
```


Publication Count
Zika
SARS
MPox
ng Kong
N2 Asian
1 Swine
Russian
MERS
OVID−19
Ebola
Polio
IV/AIDS
Cholera
Policy Citations
Year
Legend
Policy Citations
Publication Count