```
#------------------------------------------------------------------------------

# P5 positions on conflict management and resolution
# Coding date: November 2023-January 2024
# Coded by: Andrea Knapp (andrea.knapp@uni-potsdam.de)

#------------------------------------------------------------------------------

# Sections:
# 1) Before the analysis
# 2) Data preparation and pre-processing
# 3) Creation of dictionary
# 4) Latent Semantic Scaling
# 5) Validation of pre-processing and seeding
# 6) Analysis: Difference by speaker
# 7) Analysis: Difference over time
# 8) Analysis: Difference over time by speaker
# 9) Analysis: Difference across target countries
# 10) Analysis: Position of P5 across three periods
# 11) Analysis: Heatmaps across conflicts
# 12) Analysis: Most polarized cases

#------------------------------------------------------------------------------

# 1) Before the analysis

# load packages
rm(list = ls())
library(dplyr)
library(readtext)
library(quanteda)
library(patchwork)
library(dplyr)
library(ggplot2)
library(tidyverse)
library(tidytext)
library(dplyr)
library(quanteda.textstats)
library(geodata)
library(maps)
library(ggpubr)
library(quanteda.textplots)
library(udpipe)
library(gridExtra)
```

```
library(LSX)
library(preText)

# set working directory
setwd("...") # change directory

# load data
load("filtered_dataset")
load("toks_sent")
load("corp_speeches")
load("corp_sent")
load("dfm_sent")


#------------------------------------------------------------------------------

# 2) Data preparation and pre-processing

# load dataset by Schönfeld et al. (2025)
load(".../docs_meta.RData") # change directory
load("/.../docs.RData") # change directory

# merge dataset
meta_speeches$doc_id  <- meta_speeches$filename
merged_dataset <- merge(meta_speeches, raw_docs, by = "filename", all = TRUE)

# filter speakers of interest
filtered_dataset <- merged_dataset %>% filter(country_org %in% c("China", "France",
"Russian Federation",
                "United Kingdom Of Great Britain And Northern Ireland", "United States Of
America"))
filtered_dataset <- filtered_dataset %>%
  mutate(country_org = ifelse(country_org == "United States Of America", "United States",
country_org))
filtered_dataset <- filtered_dataset %>%
  mutate(country_org = ifelse(country_org == "United Kingdom Of Great Britain And
Northern Ireland", "United Kingdom", country_org))
filtered_dataset <- filtered_dataset %>%
  mutate(country_org = ifelse(country_org == "Russian Federation", "Russia", country_org))

# eliminate when P5 were President and did not speak in national capacity
filtered_dataset <- filtered_dataset %>%
  filter(!(participanttype == "The President"))
any(filtered_dataset$role_in_un == "The President")
```

```
save(filtered_dataset, file = "filtered_dataset")

# create corpus
corp_speeches <- corpus(filtered_dataset, text_field = "content")
save(corp_speeches, file = "corp_speeches")

# break down to sentences level for analysis
corp_sent <- corpus_reshape(corp_speeches, to =  "sentences")
save(corp_sent, file = "corp_sent")

# tokenization and pre-processing
toks_sent <- corp_sent %>%
  tokens(remove_punct = TRUE, remove_symbols = TRUE,
       remove_numbers = TRUE, remove_url = TRUE) %>% tokens_remove(stopwords("en"))
%>% tokens_tolower()

# compound multi-words of interest
multiword <- c("united nations", "security council", "azerbaijani republic",
          "bosnia herzegovina", "bosnia-herzegovina", "burkina faso", "central african
republic",
          "cote d'ivoire", "côte d'ivoire", "czech republic", "democratic people's republic
korea",
          "democratic republic congo", "dr congo", "dominican republic", "east timor", "timor
leste",
          "timor-leste", "el salvador", "equatorial guinea", "guinea-bissau", "islamic republic
afghanistan",
          "islamic republic iran", "libyan arab jamahiriya", "new zealand", "northern ireland",
          "north korea", "papua new guinea", "people's republic  china", "people's republic
southern yemen",
          "republic congo", "republic korea", "russian federation", "saudi arabia", "sierra leone",
          "southern rhodesia", "south africa", "south sudan", "sri lanka", "trinidad tobago",
"united arab emirates",
          "union soviet socialist republics", "soviet union", "united arab republic", "united
kingdom",
          "united states", "viet nam", "palestinian state", "central america", "great lakes", "great
lakes region",
          "lake chad basin", "northern uganda", "syrian golan", "golan heights", "latin america",
"mazar-e-sharif",
          "south america", "middle east", "sahel region", "sub-saharan africa", "western
sahara", "eastern aleppo",
          "east jerusalem", "west bank", "former yugoslvia", "gaza strip", "west africa", "central
africa",
          "african union", "al aqsa", "al-qaida", "as qaida", "al-nusra front", "al nusra front", "al
nusrah front",
```

"lord's resistance army", "al nusra", "al-nusra", "al-nusrah", "al-nusra", "al-shabaab", "al shabaab",
"arab league", "boko haram", "council of europe", "european union", "general assembly", "united nations general assembly",
"international court of justice", "international criminal court", "islamic state iraq levant", "islamic state",
"lusaka protocol", "occupied territories", "sahel joint force", "joint force", "secretary general",
"secretary-general", "security council", "united nations security council", "union south american nations",
"peace initiative", "humanitarian affairs", "economic development", "civil society",
"peaceful implementation", "g-5 sahel", "rome statute", "drug trafficking", "human trafficking",
"light weapons", "illegal exploitation", "sanctions regime", "terrorist fighter", "peacekeeping mission",
"early warning", "human rights", "international law", "humanitarian law", "humanitarian assistance",
"peacekeeping operation", "foreign affairs", "civilian population", "terrorist group", "terrorist groups",
"civil society", "chemical weapon", "chemical weapons", "conflict prevention", "territorial integrity",
"two-state solution", "political solution", "regional organization", "regional organizations",
"voluntary repatriation", "child soldiers", "food insecurity", "humanitarian workers", "international partners",
"forced displacement", "civil war", "civil wars", "extremist group", "extremist groups", "multinational force",
"food programme", "integrated stabilization mission", "mass destruction", "humanitarian access",
"multidimensional integration mission", "humanitarian crisis", "hybrid operation",
"sexual violence", "gender-based violence", "stabilization mission", "arms embargo", "good governance",
"small arms", "peace agreement", "terrorist acts", "cease fire", "geneva convention",
"armed conflict", "armed conflicts", "humanitarian aid", "climate change", "natural resources",
"armed forces", "violent extremism", "international peace security", "organized crime",
"neighboring countries", "war crimes", "crimes humanity", "ethnic cleansing", "good governance",
"mass atrocity", "mass atrocities", "cease-fire agreement", "law enforcement",
"peace enforcement", "international community", "abyei area", "displaced persons",
"monitoring team", "sanctions list", "al-qaida sanctions", "advisory committee",
"technical assistance", "political process", "international peace", "host country", "humanitarian personnel",

```
        "economic resources", "sexual exploitation", "military equipment", "political
independence",
        "international tribunal", "election process", "humanitarian organization", "assistance
mission",
        "criminal court", "reconciliation process", "arab territories", "territorial control",
"territory control",
        "international organization", "special committee", "non-governmental organization",
        "humanitarian relief", "military observers", "civilian casualties", "civilian casualty",
"chapter v",
        "chapter vi", "chapter vii", "transition assistance group", "good offices mission",
        "transitional administration", "observer mission", "integrated mission", "preventative
deployment force",
        "protection force", "emergency force", "supervision mission", "political reform",
"political reforms",
        "power sharing", "rule law", "human rights promotion", "institution building",
"responsibility protect")
toks_sent <- tokens_compound(toks_sent, pattern = phrase(multiword))

# stemming
toks_sent <- tokens_wordstem(toks_sent)
save(toks_sent, file = "toks_sent")
dfm_sent <- dfm(tokens(toks_sent))
dfm_sent <- dfm_trim(dfm_sent, min_termfreq = 10)
save(dfm_sent, file = "dfm_sent")

#------------------------------------------------------------------------------

# 3) Creation of dictionary

# mandate dictionary by Hellmüller et al. (2023)
dictionary_intervention <- list(
  resolution = c("civil_societi", "civil-societi", "build capac", "capacity-build", "capac build",
"capac built",
          "dialogu", "reconcili", "econom reform", "reform econom", "good_govern",
"human_right", "human-right",
          "build institut", "institut build", "institut built", "institution-build", "transit author",
          "transit govern", "transit state", "transit administr", "independ media", "judici
reform", "reform judici",
          "rule_law reform", "reform rule_law",  "legisl reform", "reform legisl", "legal reform",
"reform legal",
          "penal system reform", "gender-based_viol", "sexual_viol", "secur sector reform",
"reform secur sector",
          "support state", "support author", "support administr", "support govern", "transit
justic",
```

```
          "women right", "right women", "particip women", "women particip"),
  management = c("coordin donor", "coordin partner", "coordin united_n agenc", "demilitar",
"demin",
          "elimin chemical_weapon", "chemical_weapon programm", "good offic", "mediat",
          "humanitarian_assist", "observ", "monitor", "refuge assist", "assist refuge",
          "intern displaced_person assist", "assist intern displaced_person", "idp assist",
"assist idp",
          "salw", "small_arm light_weapon", "secur environ", "deliv aid", "deliveri aid", "aid
deliv", "aid deliveri"))


#------------------------------------------------------------------------------

# 4) Latent Semantic Scaling

# create seeds
seed_intervention <- as.seedwords(dictionary_intervention)

# LSS
lss_intervention <- textmodel_lss(dfm_sent, seeds = seed_intervention, k = 300, cache =
TRUE)
coefficients_intervention <- coef(lss_intervention)
sorted_coefficients_intervention <- sort(coefficients_intervention, decreasing = TRUE)
head(sorted_coefficients_intervention, 100)
tail(sorted_coefficients_intervention, 100)

# print mostly strongly associated words
plot_data <- data.frame(
  Term = names(sorted_coefficients_intervention),
  Coefficient = sorted_coefficients_intervention)

# top 20 for CR
top_plot <- ggplot(head(plot_data, 20), aes(x = reorder(Term, Coefficient), y = Coefficient))
+
  geom_bar(stat = "identity", fill = "mediumblue", color = "black") +
  labs(title = "Most strongly associated words: CR",
       x = "",
       y = "Coefficients") +
  theme_minimal() + theme(
    axis.text.x = element_text(angle = 45, hjust = 1, size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    plot.title = element_text(size = 15))
```

```
# top 20 for CM
bottom_plot <- ggplot(tail(plot_data, 20), aes(x = reorder(Term, -Coefficient), y =
Coefficient)) +
  geom_bar(stat = "identity", fill = "orangered2", color = "black") +
  labs(title = "Most strongly associated words: CM",
       x = "",
       y = "Coefficients") +
  theme_minimal() + theme(
        axis.text.x = element_text(angle = 45, hjust = 1, size = 11),
        axis.text.y = element_text(size = 11),
        axis.title.x = element_text(size = 13),
        axis.title.y = element_text(size = 13),
        plot.title = element_text(size = 15))

# visualize
grid.arrange(top_plot, bottom_plot, ncol = 1)

# reassemble sentences into documents
dfm_doc_intervention <- dfm_group(dfm_sent)
dfm_intervention <- docvars(dfm_doc_intervention)
dfm_intervention$fit <- predict(lss_intervention, newdata = dfm_doc_intervention)



#------------------------------------------------------------------------------

# 5) Validation of pre-processing and seeding

# validate pre-processing choices (random sample because full corpus is too large)
# attention: depending on the operating system, this procedure can take a few hours
set.seed(123)
sampled_corp_sent <- corp_sent[sample(length(corp_sent), size = 500)]
preprocessed_documents <- factorial_preprocessing(sampled_corp_sent, use_ngrams =
TRUE,
                                  infrequent_term_threshold = 0.1, verbose = FALSE)
preText_results <- preText(preprocessed_documents, dataset_name = "Speeches",
                   distance_method = "cosine", num_comparisons = 50, verbose = TRUE)
coefficients_df <- data.frame(Variable = preText_results$regression_results$Variable,
                    Estimate = preText_results$regression_results$Coefficient,
                    SE = preText_results$regression_results$SE)
coefficients_df <- coefficients_df[coefficients_df$Variable != "Intercept", ]
ggplot(coefficients_df, aes(x = reorder(Variable, Estimate), y = Estimate)) +
  geom_point() +
  geom_errorbar(aes(ymin = Estimate - SE, ymax = Estimate + SE), width = 0.2) +
```

```
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  coord_flip() + labs(title = "", x = "", y = "Coefficient estimate") +
  theme_minimal() + theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13))

# corpus without eliminating infrequent words
toks_sent_all <- corp_sent %>%
  tokens(remove_punct = TRUE, remove_symbols = TRUE,
       remove_numbers = TRUE, remove_url = TRUE) %>% tokens_remove(stopwords("en"))
%>% tokens_tolower()
toks_sent_all <- tokens_compound(toks_sent_all, pattern = phrase(multiword))
toks_sent_all <- tokens_wordstem(toks_sent_all)
dfm_sent_all <- dfm(tokens(toks_sent_all))
lss_intervention_all <- textmodel_lss(dfm_sent_all, seeds = seed_intervention, k = 300,
cache = TRUE)
dfm_doc_intervention_all <- dfm_group(dfm_sent_all)
dfm_intervention_all <- docvars(dfm_doc_intervention_all)
dfm_intervention_all$fit <- predict(lss_intervention_all, newdata =
dfm_doc_intervention_all)
summary_by_country_all <- dfm_intervention_all %>%
  group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE),
          total_documents = n()) %>%
  mutate(country = factor(country_org,
                  levels = c("United States", "United Kingdom",  "Russia", "France", "China")))
summary_by_country_all <- summary_by_country_all %>%
  mutate(lower_ci = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
       upper_ci = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents))
ggplot(summary_by_country_all, aes(x = mean_fit, y = country, label = country_org)) +
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) +
  labs(title = "",
      x = "Polarization (negative values = CM, positive values = CR)", y = "Country names
(alphabetical)") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.x = element_text(size = 13),
```

```
   axis.title.y = element_text(size = 13)) +
  geom_vline(xintercept = 0, linetype = "dashed", color = "red") +
  scale_x_continuous(breaks = seq(-0.4, 0.4, 0.1), limits = c(-0.4, 0.4))

# corpus without eliminating stop words
toks_sent_stop <- corp_sent %>%
  tokens(remove_punct = TRUE, remove_symbols = TRUE,
       remove_numbers = TRUE, remove_url = TRUE) %>% tokens_tolower()
toks_sent_stop <- tokens_compound(toks_sent_stop, pattern = phrase(multiword))
toks_sent_stop <- tokens_wordstem(toks_sent_stop)
dfm_sent_stop <- dfm(tokens(toks_sent_stop))
dfm_sent_stop <- dfm_trim(dfm_sent_stop, min_termfreq = 10)
lss_intervention_stop <- textmodel_lss(dfm_sent_stop, seeds = seed_intervention, k = 300,
cache = TRUE)
dfm_doc_intervention_stop <- dfm_group(dfm_sent_stop)
dfm_intervention_stop <- docvars(dfm_doc_intervention_stop)
dfm_intervention_stop$fit <- predict(lss_intervention_stop, newdata =
dfm_doc_intervention_stop)
summary_by_country_stop <- dfm_intervention_stop %>%
  group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE),
          total_documents = n()) %>%
  mutate(country = factor(country_org,
                  levels = c("United States", "United Kingdom",  "Russia", "France", "China")))
summary_by_country_stop <- summary_by_country_stop %>%
  mutate(lower_ci = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
       upper_ci = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents))
ggplot(summary_by_country_stop, aes(x = mean_fit, y = country, label = country_org)) +
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) +
  labs(title = "",
      x = "Polarization (negative values = CM, positive values = CR)", y = "Country names
(alphabetical)") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13)) +
  geom_vline(xintercept = 0, linetype = "dashed", color = "red") +
  scale_x_continuous(breaks = seq(-0.4, 0.4, 0.1), limits = c(-0.4, 0.4))
```

```
# eliminate place names
country_data <- geodata::country_codes()
countries <- country_data$NAME
cities <- unique(world.cities$name)
regions <- c("Donbas", "Crimea", "Transnistria", "Kosovo", "Vojvodina", "Kaliningrad",
         "Kurdistan", "Palestine", "Gaza", "West Bank", "Rojava", "South Ossetia",
         "Abkhazia", "Donetsk", "Luhansk", "Chechnya", "Ingushetia", "Dagestan",
         "Darfur", "Western Sahara", "Biafra", "Tibet", "Xinjiang", "Kashmir",
         "Taiwan", "Nagorno-Karabakh", "Sinai", "Golan Heights",
         "Somaliland", "Azawad", "Ogaden")
toks_sent_place <- corp_sent %>%
  tokens(remove_punct = TRUE, remove_symbols = TRUE,
       remove_numbers = TRUE, remove_url = TRUE) %>%
  tokens_remove(pattern = countries,  valuetype = "fixed",case_insensitive = TRUE) %>%
  tokens_remove(pattern = cities, valuetype = "fixed", case_insensitive = TRUE) %>%
  tokens_remove(pattern = regions, valuetype = "fixed", case_insensitive = TRUE) %>%
  tokens_remove(stopwords("en"))
toks_sent_place <- tokens_compound(toks_sent_place, pattern = phrase(multiword))
toks_sent_place <- tokens_wordstem(toks_sent_place)
dfm_sent_place <- dfm(tokens(toks_sent_place))
dfm_sent_place <- dfm_trim(dfm_sent_place, min_termfreq = 10)
lss_intervention_place <- textmodel_lss(dfm_sent_place, seeds = seed_intervention, k =
300, cache = TRUE)
dfm_doc_intervention_place <- dfm_group(dfm_sent_place)
dfm_intervention_place <- docvars(dfm_doc_intervention_place)
dfm_intervention_place$fit <- predict(lss_intervention_place, newdata =
dfm_doc_intervention_place)
summary_by_country_place <- dfm_intervention_place %>%
  group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE),
          total_documents = n()) %>%
  mutate(country = factor(country_org,
                    levels = c("United States", "United Kingdom",  "Russia", "France", "China")))
summary_by_country_place <- summary_by_country_place %>%
  mutate(lower_ci = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
       upper_ci = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents))
ggplot(summary_by_country_place, aes(x = mean_fit, y = country, label = country_org)) +
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) +
  labs(title = "",
     x = "Polarization (negative values = CM, positive values = CR)", y = "Country names
(alphabetical)") +
  theme_minimal() +
```

```
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13)) +
  geom_vline(xintercept = 0, linetype = "dashed", color = "red") +
  scale_x_continuous(breaks = seq(-0.4, 0.4, 0.1), limits = c(-0.4, 0.4))

# suggestion reviewer change dictionary
dictionary_intervention_changed <- list(
  resolution = c("civil_societi", "civil-societi", "build capac", "capacity-build", "capac build",
"capac built",
             "dialogu", "reconcili", "econom reform", "reform econom", "good_govern",
"human_right", "human-right",
             "build institut", "institut build", "institut built", "institution-build", "transit author",
             "transit govern", "transit state", "transit administr", "independ media", "judici
reform", "reform judici",
             "rule_law reform", "reform rule_law",  "legisl reform", "reform legisl", "legal reform",
"reform legal",
             "penal system reform", "gender-based_viol", "sexual_viol", "secur sector reform",
"reform secur sector",
             "support state", "support author", "support administr", "support govern", "transit
justic",
             "women right", "right women", "particip women", "women particip", "genocid",
"ethnic_cleans*"),
  management = c("coordin donor", "coordin partner", "coordin united_n agenc", "demilitar",
"demin",
             "elimin chemical_weapon", "chemical_weapon programm", "good offic", "mediat",
             "humanitarian_assist", "observ", "monitor",
             "salw", "small_arm light_weapon", "secur environ", "deliv aid", "deliveri aid", "aid
deliv", "aid deliveri"))
seed_intervention_changed <- as.seedwords(dictionary_intervention_changed)
lss_intervention_changed <- textmodel_lss(dfm_sent, seeds = seed_intervention_changed,
k = 300, cache = TRUE)
dfm_doc_intervention_changed <- dfm_group(dfm_sent)
dfm_intervention_changed <- docvars(dfm_doc_intervention_changed)
dfm_intervention_changed$fit <- predict(lss_intervention_changed, newdata =
dfm_doc_intervention_changed)
summary_by_country_changed <- dfm_intervention_changed %>%
  group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE),
            total_documents = n()) %>%
  mutate(country = factor(country_org,
```

```
                levels = c("United States", "United Kingdom",  "Russia", "France", "China")))
summary_by_country_changed <- summary_by_country_changed %>%
  mutate(lower_ci = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
       upper_ci = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents))
ggplot(summary_by_country_changed, aes(x = mean_fit, y = country, label = country_org))
+
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) +
  labs(title = "",
      x = "Polarization (negative values = CM, positive values = CR)", y = "Country names
(alphabetical)") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13)) +
  geom_vline(xintercept = 0, linetype = "dashed", color = "red") +
  scale_x_continuous(breaks = seq(-0.4, 0.4, 0.1), limits = c(-0.4, 0.4))

# validate seed words
analyze_seed_word_coherence <- function(dfm, seed_dict, max_vectors = 500) {
  coherence_results <- list()
  for (category in names(seed_dict)) {
    seeds <- seed_dict[[category]]
    existing_seeds <- seeds[seeds %in% colnames(dfm)]
    if (length(existing_seeds) < 2) {
      coherence_results[[category]] <- list(
        warning = "Insufficient seeds for analysis")
      next
    }
    if (length(existing_seeds) > max_vectors) {
      existing_seeds <- existing_seeds[1:max_vectors]
      warning(paste("Truncated", category, "seeds to", max_vectors, "vectors"))
    }
    tryCatch({
      seed_vectors <- as.matrix(dfm[, existing_seeds])
      similarity_results <- compute_pairwise_similarities(seed_vectors)
      coherence_results[[category]] <- list(
        mean_similarity = similarity_results$mean_similarity,
        median_similarity = similarity_results$median_similarity,
        similarity_std = similarity_results$similarity_std,
```

```
      seeds_used = existing_seeds,
      similarity_summary = list(
       min = min(similarity_results$similarities, na.rm = TRUE),
       max = max(similarity_results$similarities, na.rm = TRUE),
       quantiles = quantile(similarity_results$similarities, na.rm = TRUE)))
   }, error = function(e) {
    coherence_results[[category]] <- list(
     error = paste("Error processing category:", e$message))
   })
  }
  return(coherence_results)
}
compute_pairwise_similarities <- function(matrix) {
  similarities <- matrix(0, nrow = ncol(matrix), ncol = ncol(matrix))
  for (i in 1:(ncol(matrix)-1)) {
   for (j in (i+1):ncol(matrix)) {
    sim <- sum(matrix[,i] * matrix[,j]) /
     (sqrt(sum(matrix[,i]^2)) * sqrt(sum(matrix[,j]^2)))
    similarities[i,j] <- similarities[j,i] <- sim
   }
  }
  diag(similarities) <- NA
  flat_similarities <- similarities[lower.tri(similarities)]
  list(
   similarities = flat_similarities,
   mean_similarity = mean(flat_similarities, na.rm = TRUE),
   median_similarity = median(flat_similarities, na.rm = TRUE),
   similarity_std = sd(flat_similarities, na.rm = TRUE))
}
tryCatch({
  coherence_results <- analyze_seed_word_coherence(dfm_sent, dictionary_intervention)
  print(coherence_results)
  lapply(coherence_results, function(category) {
   if (!is.null(category$mean_similarity)) {
    list(
     seeds_used = length(category$seeds_used),
     mean_similarity = category$mean_similarity,
     median_similarity = category$median_similarity,
     similarity_std = category$similarity_std,
     similarity_summary = category$similarity_summary)
   } else {
    category
   }
  })
```

```
}, error = function(e) {
  cat("Error in coherence analysis:", e$message, "\n")
  NULL
})
visualize_seed_coherence <- function(coherence_results) {
  names(coherence_results)[names(coherence_results) == "management"] <- "CM"
  names(coherence_results)[names(coherence_results) == "resolution"] <- "CR"
  plot_data <- lapply(names(coherence_results), function(category) {
    if (!is.null(coherence_results[[category]]$similarity_summary)) {
      data.frame(
        Category = category,
        Metric = c("Min", "25th Percentile", "Median", "75th Percentile", "Max"),
        Value = unlist(coherence_results[[category]]$similarity_summary$quantiles))
    }
  })
  plot_df <- do.call(rbind, plot_data)
  p1 <- ggplot(plot_df, aes(x = Metric, y = Value, fill = Category)) +
    geom_bar(stat = "identity", position = "dodge") +
    labs(title = "",
         y = "Cosine similarity",
         x = "Quantile") +
    theme_minimal() +
    theme(axis.text.x = element_text(angle = 0, hjust = 0))
  boxplot_data <- lapply(names(coherence_results), function(category) {
    if (!is.null(coherence_results[[category]]$mean_similarity)) {
      data.frame(
        Category = category,
        Similarity = coherence_results[[category]]$similarity_summary$quantiles)
    }
  })
  boxplot_df <- do.call(rbind, boxplot_data)
  p2 <- ggplot(boxplot_df, aes(x = Category, y = Similarity)) +
    geom_boxplot(aes(fill = Category)) +
    labs(title = "",
         y = "Cosine similarity",
         x = "Category") +
    theme_minimal()
  return(list(quantile_plot = p1, boxplot = p2))
}
plots <- visualize_seed_coherence(coherence_results)
print(plots$quantile_plot)
print(plots$boxplot)

# manual validation of 100 cases
```

```
lss_scores <- as.numeric(dfm_sent %*% lss_intervention$beta)
lss_df <- data.frame(document = docnames(dfm_sent), lss_score = lss_scores)
top_50 <- lss_df[order(-lss_df$lss_score), ][1:50, ]
lss_scores <- as.numeric(dfm_sent %*% lss_intervention$beta)
lss_df <- data.frame(document = docnames(dfm_sent), lss_score = lss_scores)
bottom_50 <- lss_df[order(lss_df$lss_score), ][1:50, ]
bottom_50$text <- texts(corp_sent)[match(bottom_50$document, docnames(corp_sent))]
print(top_50)


#------------------------------------------------------------------------------

# 6) Analysis: Difference by speaker (static)

# filter data
summary_by_country <- dfm_intervention %>%
  group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE),
            total_documents = n()) %>%
  mutate(country = factor(country_org,
                          levels = c("United States", "United Kingdom",  "Russia", "France", "China")))
summary_by_country <- summary_by_country %>%
  mutate(lower_ci = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
         upper_ci = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents))

# visualize
ggplot(summary_by_country, aes(x = mean_fit, y = country, label = country_org)) +
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) +
  labs(title = "",
       x = "Polarization (negative values = CM, positive values = CR)", y = "Country names
(alphabetical)") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13)) +
  geom_vline(xintercept = 0, linetype = "dashed", color = "red") +
  scale_x_continuous(breaks = seq(-0.4, 0.4, 0.1), limits = c(-0.4, 0.4))
ggsave("figure2.png", width = 7, height = 4, dpi = 600)
```

```
ggplot(summary_by_country, aes(x = mean_fit, y = country, label = country_org)) +
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) +
  labs(title = "",
       x = "Polarization (negative values = CM, positive values = CR)",
       y = "Country names (alphabetical)") +
  theme_minimal() +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13)) +
  geom_vline(xintercept = 0, linetype = "dashed", color = "red") +
  scale_x_continuous(breaks = seq(-0.4, 0.4, 0.1), limits = c(-0.4, 0.4)) +
  coord_fixed(ratio = 0.15)


#------------------------------------------------------------------------------

# 7) Analysis: Difference over time

# filter data
summary_by_year <- dfm_intervention %>%
  group_by(year) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE),
            lower_ci = mean_fit - 1.96 * sd(fit, na.rm = TRUE) / sqrt(n()),
            upper_ci = mean_fit + 1.96 * sd(fit, na.rm = TRUE) / sqrt(n()))

# visualize
ggplot(summary_by_year, aes(x = year, y = mean_fit)) +
  geom_line() +
  geom_ribbon(aes(ymin = lower_ci, ymax = upper_ci), alpha = 0.3, fill = "grey") +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  labs(title = "",
       x = "Year", y = "Median fit (negative values = CM, positive values = CR)") +
  theme_minimal() + theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13)) +
  scale_x_continuous(breaks = seq(min(summary_by_year$year),
max(summary_by_year$year), by = 2)) +
```

```
  scale_y_continuous(breaks = seq(-1, 0.5, by = 0.25))


#------------------------------------------------------------------------------

# 8) Analysis: Difference over time by speaker

# filter data
summary_by_country_year <- dfm_intervention %>%
  group_by(country_org, year) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE),
            ci = 1.96 * sd(fit, na.rm = TRUE) / sqrt(n()),
            .groups = 'drop')
summary_by_country_year_all <- summary_by_country_year %>%
  mutate(panel = country_org) %>%
  bind_rows(summary_by_country_year %>% mutate(panel = "All"))

# visualize
ggplot(summary_by_country_year_all, aes(x = year, y = mean_fit, group = country_org, color
= country_org, linetype = country_org)) +
  geom_line(size = 0.8) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  geom_ribbon(aes(ymin = ifelse(panel == "All", NA, mean_fit - ci),
                  ymax = ifelse(panel == "All", NA, mean_fit + ci),
                  fill = country_org), alpha = 0.2, color = NA) +
  labs(title = "",
       x = "Year",
       y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-2, -1, 0, 1, 2), limits = c(-2, 2)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) +
  theme_minimal() +
  facet_wrap(~panel, scales = "free_x", nrow = 2, ncol = 3) +
  theme(axis.text.x = element_text(size = 11),
        axis.text.y = element_text(size = 11),
        axis.title.x = element_text(size = 13),
        axis.title.y = element_text(size = 13),
        strip.text = element_text(size = 13),
        legend.position = "bottom")
ggsave("figure3.png", width = 12, height = 6, dpi = 600)

#------------------------------------------------------------------------------

# 9) Analysis: Difference across target countries
```

```
# filter data
unique_agenda_items <- unique(dfm_intervention$agenda_item_manual)
view(unique_agenda_items)
dfm_theme_intervention <- dfm_intervention %>%
  group_by(agenda_item_manual, country_org, year) %>%
  summarise(mean_fit = median(fit, na.rm = TRUE),
            total_documents = n(),
            ci = 1.96 * IQR(fit, na.rm = TRUE) / sqrt(n()))

# Afghanistan
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Afghanistan")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Africa
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Afghanistan")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
```

```
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Angola
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Angola")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
```

```
   strip.text = element_text(size = 13),
   legend.position = "none")

# Bosnia and Herzegovina
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Bosnia And Herzegovina")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Burundi
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Burundi")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
```

```
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Central African Republic
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Central African Republic")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                x = "Year",
                y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Central African Republic-Chad
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Central African Republic/Chad")) %>%
```

```
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")


# Colombia
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Colombia")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
```

```
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Côte d'Ivoire
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Côte D'ivoire")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Croatia
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Croatia")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                     x = "Year",
```

```
            y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Cyprus
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Cyprus")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")
```

```
# DR Congo
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Democratic Republic Of The Congo")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("El Salvador")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
```

```
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Ethiopia/Eritrea
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Ethiopia/Eritrea")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
           x = "Year",
           y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
           ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
           fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Former Yugoslav Republic of Macedonia
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Former Yugoslav Republic Of Macedonia")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
```

```
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Georgia
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Georgia")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
```

```
   strip.text = element_text(size = 13),
   legend.position = "none")

# Great Lakes
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Great Lakes")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Guinea-Bissau
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Guinea-Bissau")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
```

```
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Haiti
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Haiti")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Iraq
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Iraq")) %>%
```

```
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Iraq/Kuwait
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Iraq/Kuwait")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
```

```
   axis.text.x = element_text(size = 11),
   axis.text.y = element_text(size = 11),
   axis.title.x = element_text(size = 13),
   axis.title.y = element_text(size = 13),
   strip.text = element_text(size = 13),
   legend.position = "none")

# Israel/Lebanon
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Israel/Lebanon")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
            ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
            fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
   axis.text.x = element_text(size = 11),
   axis.text.y = element_text(size = 11),
   axis.title.x = element_text(size = 13),
   axis.title.y = element_text(size = 13),
   strip.text = element_text(size = 13),
   legend.position = "none")

# Israel/Palestine
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Israel/Palestine")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
```

```
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Israel/Syria
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Israel/Syria")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                x = "Year",
                y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")
```

```
# Kosovo
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Kosovo")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Lebanon
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Lebanon")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
```

```
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Liberia
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Liberia")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
           x = "Year",
           y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
           ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
           fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Libya
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Libya")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
```

```
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Mali
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Mali")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
```

```
   strip.text = element_text(size = 13),
   legend.position = "none")

# Middle East
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Middle East")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Mozambique
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Mozambique")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
```

```
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Myanmar
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Myanmar")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                x = "Year",
                y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Nagorny Karabakh
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Nagorny Karabakh")) %>%
```

```
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Nepal
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Nepal")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
```

```
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# North Korea
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Democratic People's Republic Of Korea")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
            ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
            fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Papua New Guinea
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Papua New Guinea")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
```

```
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Rwanda
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Rwanda")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                x = "Year",
                y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")
```

```
# Sierra Leone
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Sierra Leone")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
            ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
            fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Somalia
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Somalia")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
            ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
            fill = country_org, alpha = 0.2), color = NA) +
```

```
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# South Africa
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("South Africa")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
            ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
            fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Sudan/South Sudan
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Sudan", "Sudan/South Sudan")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
```

```
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Syria
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Syria")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                     x = "Year",
                     y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
                  ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
                  fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
```

```
   strip.text = element_text(size = 13),
   legend.position = "none")

# Tajikistan
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Tajikistan")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# Timor-Leste
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Timor-Leste")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
```

```
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Ukraine
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Ukraine")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                x = "Year",
                y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Ukraine
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Venezuela")) %>%
```

```
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
  axis.text.x = element_text(size = 11),
  axis.text.y = element_text(size = 11),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13),
  strip.text = element_text(size = 13),
  legend.position = "none")

# West Africa
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("West Africa")) %>%
 group_by(year)
ggplot(summaryplot_data,
    aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
             ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
             fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
```

```
   axis.text.x = element_text(size = 11),
   axis.text.y = element_text(size = 11),
   axis.title.x = element_text(size = 13),
   axis.title.y = element_text(size = 13),
   strip.text = element_text(size = 13),
   legend.position = "none")

# Western Sahara
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Western Sahara")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
 scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
 scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
 geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
            ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
            fill = country_org, alpha = 0.2), color = NA) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
 facet_wrap(~country_org, scales = "free_x", nrow = 3) +
 theme(
   axis.text.x = element_text(size = 11),
   axis.text.y = element_text(size = 11),
   axis.title.x = element_text(size = 13),
   axis.title.y = element_text(size = 13),
   strip.text = element_text(size = 13),
   legend.position = "none")

# Yemen
summaryplot_data <- dfm_theme_intervention %>%
 filter(agenda_item_manual %in% c("Yemen")) %>%
 group_by(year)
ggplot(summaryplot_data,
     aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
 geom_line() + labs(title = "",
              x = "Year",
              y = "Polarization (negative values = CM, positive values = CR)") +
```

```
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")

# Yugoslavia
summaryplot_data <- dfm_theme_intervention %>%
  filter(agenda_item_manual %in% c("Former Yugoslavia")) %>%
  group_by(year)
ggplot(summaryplot_data,
       aes(x = year, y = mean_fit, group = interaction(agenda_item_manual, country_org), color
= country_org)) +
  geom_line() + labs(title = "",
                x = "Year",
                y = "Polarization (negative values = CM, positive values = CR)") +
  scale_y_continuous(breaks = c(-5, -2.5, 0, 2.5, 5), limits = c(-5, 5)) +
  scale_x_continuous(breaks = c(1995, 2000, 2005, 2010, 2015, 2020), limits = c(1992,
2023)) + theme_minimal() +
  geom_ribbon(aes(ymin = mean_fit - 1.96 * sd(mean_fit, na.rm = TRUE) /
sqrt(total_documents),
              ymax = mean_fit + 1.96 * sd(mean_fit, na.rm = TRUE) / sqrt(total_documents),
              fill = country_org, alpha = 0.2), color = NA) +
  geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  facet_wrap(~country_org, scales = "free_x", nrow = 3) +
  theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_text(size = 11),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    legend.position = "none")
```

```
--------------------------------------------------------------------------------

# 10) Analysis: Position of P5 across three periods

# visualization per period
dfm_intervention_1992_2001 <- dfm_intervention %>% filter(year %in% 1992:2001)
summary_by_country <- dfm_intervention_1992_2001 %>% group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE), total_documents = n())
summary_by_country <- summary_by_country %>% mutate(
    sd_fit = sd(dfm_intervention_1992_2001$fit, na.rm = TRUE),
    lower_ci = mean_fit - 1.96 * sd_fit / sqrt(total_documents),
    upper_ci = mean_fit + 1.96 * sd_fit / sqrt(total_documents))
summary_by_country$country_org <- factor(summary_by_country$country_org,
                            levels = c("United States", "United Kingdom", "Russia", "France",
"China"))
plot1 <- ggplot(summary_by_country, aes(x = mean_fit, y = country_org, label =
country_org)) +
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) + labs(title = "1992-2001", x = "",
    y = "Country names (alphabetical)") + theme_minimal() +
  theme(axis.text.x = element_text(size = 11),
        axis.text.y = element_blank(),
        axis.ticks.y = element_blank(),
        axis.title.x = element_text(size = 13),
        axis.title.y = element_text(size = 13),
        strip.text = element_text(size = 13),
        plot.title = element_text(hjust = 0.5),
        legend.position = "none") + geom_vline(xintercept = 0, linetype = "dashed", color =
"red") +
  scale_x_continuous(breaks = seq(-1, 1, 0.5), limits = c(-1, 1))
dfm_intervention_2002_2011 <- dfm_intervention %>% filter(year %in% 2002:2011)
summary_by_country <- dfm_intervention_2002_2011 %>% group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE), total_documents = n())
summary_by_country <- summary_by_country %>%
  mutate(sd_fit = sd(dfm_intervention_2002_2011$fit, na.rm = TRUE),
    lower_ci = mean_fit - 1.96 * sd_fit / sqrt(total_documents),
    upper_ci = mean_fit + 1.96 * sd_fit / sqrt(total_documents))
summary_by_country$country_org <- factor(summary_by_country$country_org,
                            levels = c("United States", "United Kingdom", "Russia", "France",
"China"))
plot2 <- ggplot(summary_by_country, aes(x = mean_fit, y = country_org, label =
country_org)) +
```

```
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) +
  labs(title = "2002-2011",x = "Polarization (negative values = CM, positive values = CR)",
       y = "") + theme_minimal() + theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    plot.title = element_text(hjust = 0.5),
    legend.position = "none") +
  geom_vline(xintercept = 0, linetype = "dashed", color = "red") +
  scale_x_continuous(breaks = seq(-1, 1, 0.5), limits = c(-1, 1))
dfm_intervention_2012_2023 <- dfm_intervention %>% filter(year %in% 2012:2023)
summary_by_country <- dfm_intervention_2012_2023 %>% group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE), total_documents = n())
summary_by_country <- summary_by_country %>%
  mutate(sd_fit = sd(dfm_intervention_2012_2023$fit, na.rm = TRUE),
    lower_ci = mean_fit - 1.96 * sd_fit / sqrt(total_documents),
    upper_ci = mean_fit + 1.96 * sd_fit / sqrt(total_documents))
summary_by_country$country_org <- factor(summary_by_country$country_org,
                              levels = c("United States", "United Kingdom", "Russia", "France",
"China"))
plot3 <- ggplot(summary_by_country, aes(x = mean_fit, y = country_org, label =
country_org)) +
  geom_point(color = "black", size = 3.5) +
  geom_errorbarh(aes(xmin = lower_ci, xmax = upper_ci), height = 0.2, color = "black") +
  geom_text(vjust = -1.4, size = 5, angle = 0) + labs(title = "2012-2023", x = "",
                                        y = "") +
  theme_minimal() + theme(
    axis.text.x = element_text(size = 11),
    axis.text.y = element_blank(),
    axis.ticks.y = element_blank(),
    axis.title.x = element_text(size = 13),
    axis.title.y = element_text(size = 13),
    strip.text = element_text(size = 13),
    plot.title = element_text(hjust = 0.5),
    legend.position = "none") + geom_vline(xintercept = 0, linetype = "dashed", color = "red")
+
  scale_x_continuous(breaks = seq(-1, 1, 0.5), limits = c(-1, 1))
ggarrange(plot1, plot2, plot3, nrow = 1)
```

```
# visualize by state
summary_data <- bind_rows(dfm_intervention %>%filter(year %in% 1992:2001) %>%
  group_by(country_org) %>% summarize(mean_fit = median(fit, na.rm = TRUE),
                          sd_fit = sd(fit, na.rm = TRUE),
                          total_documents = n(),
                          period = "1992-2001") %>% mutate(
                           lower_ci = mean_fit - 1.96 * sd_fit / sqrt(total_documents),
                           upper_ci = mean_fit + 1.96 * sd_fit / sqrt(total_documents)),
  dfm_intervention %>% filter(year %in% 2002:2011) %>% group_by(country_org) %>%
  summarize(mean_fit = median(fit, na.rm = TRUE), sd_fit = sd(fit, na.rm = TRUE),
        total_documents = n(), period = "2002-2011") %>% mutate(
         lower_ci = mean_fit - 1.96 * sd_fit / sqrt(total_documents),
         upper_ci = mean_fit + 1.96 * sd_fit / sqrt(total_documents)),
  dfm_intervention %>% filter(year %in% 2012:2023) %>% group_by(country_org) %>%
   summarize(mean_fit = median(fit, na.rm = TRUE), sd_fit = sd(fit, na.rm = TRUE),
         total_documents = n(), period = "2012-2023") %>%
   mutate(lower_ci = mean_fit - 1.96 * sd_fit / sqrt(total_documents),
       upper_ci = mean_fit + 1.96 * sd_fit / sqrt(total_documents)))
summary_data$country_org <- factor(summary_data$country_org,
                     levels = c("China", "France", "Russia", "United States", "United Kingdom"))
summary_data$period <- factor(summary_data$period,
                    levels = c("1992-2001", "2002-2011", "2012-2023"))
ggplot(summary_data, aes(x = period, y = mean_fit)) +
 geom_line(group = 1, color = "black", linetype = "dashed") +
 geom_point(color = "black", size = 3) +
 geom_errorbar(aes(ymin = lower_ci, ymax = upper_ci), width = 0.1, color = "black") +
 facet_wrap(~ country_org, scales = "free_x", ncol = 2) +
 labs(title = "", x = "Time periods", y = "Polarization (negative values = CM, positive values =
CR)") +
 theme_minimal() + theme(
  axis.text.x = element_text(angle = 0, size =11),
  axis.text.y = element_text(size =11),
  plot.title = element_text(hjust = 0.5),
  strip.text = element_text(size = 13),
  axis.title.x = element_text(size = 13),
  axis.title.y = element_text(size = 13)) +
 geom_hline(yintercept = 0, linetype = "dashed", color = "red") + ylim(-1, 1)


#-------------------------------------------------------------------------------

# 11) Analysis: Heatmaps across conflicts

# define order
```

```
agenda_item_order <- c("Yemen", "Western Sahara", "West Africa", "Timor-Leste",
                       "Tajikistan", "Syria", "Sudan/South Sudan", "Sudan",
                       "Somalia", "Sierra Leone", "Rwanda", "Papua New Guinea",
                       "Former Yugoslav Republic Of Macedonia", "Nepal",
                       "Mozambique", "Mali", "Libya", "Liberia", "Lebanon",
                       "Kosovo", "Israel/Palestine", "Iraq", "Haiti", "Guinea-Bissau",
                       "Guinea", "Guatemala", "Georgia", "Former Yugoslavia",
                       "Ethiopia/Eritrea", "El Salvador", "Democratic Republic Of The Congo",
                       "Cyprus", "Croatia", "Côte D'ivoire", "Colombia", "Central African Republic",
                       "Cambodia", "Burundi", "Bosnia And Herzegovina", "Angola", "Afghanistan")

custom_country_labels <- c("Yemen"  = "Yemen", "Western Sahara" = "Western Sahara",
                           "West Africa" = "West Africa", "Timor-Leste" = "Timor-Leste",
                           "Tajikistan" = "Tajikistan", "Syria" = "Syria",
                           "Sudan/South Sudan" = "Sudan/South Sudan", "Sudan" = "Sudan",
                           "Somalia" = "Somalia", "Sierra Leone" = "Sierra Leona",
                           "Rwanda" = "Rwanda", "Papua New Guinea" = "Papua New Guinea",
                           "Former Yugoslav Republic Of Macedonia" = "North Macedonia",
                           "Nepal" = "Nepal", "Mozambique" = "Mozambique",
                           "Mali" = "Mali", "Libya" = "Libya", "Liberia" = "Liberia",
                           "Lebanon" = "Lebanon", "Kosovo" = "Kosovo",
                           "Israel/Palestine" = "Israel", "Iraq" = "Iraq",
                           "Haiti" = "Haiti", "Guinea-Bissau" = "Guinea-Bissau",
                           "Guinea" = "Guinea", "Guatemala" = "Guatemala", "Georgia" = "Georgia",
                           "Former Yugoslavia" = "Former Yugoslavia", "Ethiopia/Eritrea" =
"Ethiopia/Eritrea",
                           "El Salvador" = "El Salvador", "Democratic Republic Of The Congo" = "DR
Congo",
                           "Cyprus" = "Cyprus", "Croatia" = "Croatia", "Côte D'ivoire" = "Côte d'Ivoire",
                           "Colombia = Colombia", "Central African Republic" = "CAR",
                           "Cambodia" = "Cambodia", "Burundi" = "Burundi", "Bosnia And Herzegovina" =
"Bosnia-Herzegovina",
                           "Angola" = "Angola", "Afghanistan" = "Afghanistan")

# custom color breaks, labels and colors
color_breaks <- c(-Inf, 0, Inf)
color_labels <- c("CM", "CR")
colors <- c("orangered2", "mediumblue")

# filter data (visualization by P5)
filtered_data <- dfm_intervention %>%
  # Recode "Sudan/South Sudan" and "Sudan" to a single label
  mutate(agenda_item_manual = case_when(
    agenda_item_manual %in% c("Sudan/South Sudan", "Sudan") ~ "Sudan",
```

```
  TRUE ~ agenda_item_manual
 )) %>%
 # Rest of your existing filtering and preprocessing
 filter(agenda_item_manual %in% agenda_item_order) %>%
 mutate(period = case_when(
  year >= 1992 & year <= 2001 ~ "1992-2001",
  year > 2002 & year <= 2011 ~ "2002-2011",
  year > 2012 & year <= 2023 ~ "2012-2023",
  TRUE ~ NA_character_
 )) %>%
 filter(!is.na(period)) %>%
 group_by(agenda_item_manual, country_org, period) %>%
 summarise(mean_fit = median(fit, na.rm = TRUE), .groups = 'drop') %>%
 mutate(agenda_item_manual = factor(agenda_item_manual, levels =
unique(agenda_item_order))) %>%
 mutate(mean_fit_category = cut(mean_fit, breaks = color_breaks,
                    labels = color_labels, include.lowest = TRUE))

# visualize
ggplot(filtered_data, aes(x = period, y = agenda_item_manual, fill = mean_fit_category)) +
 geom_tile(color = "black") +
 scale_fill_manual(values = colors, name = "Mean outcome", labels = c("CM" = "CM", "CR"
= "CR")) +
 labs(title = "", x = "", y = "") +
 theme_minimal() +
 geom_text(aes(label = round(mean_fit, 2)), color = "white", size = 3.2) +
 scale_y_discrete(labels = custom_country_labels) + # This line adds custom labels
 theme(axis.text.x = element_text(angle = 0, hjust = 0.5, size = 9),
     axis.text.y = element_text(size = 10),
     axis.title.x = element_text(size = 9),
     strip.text = element_text(size = 11),
     legend.position = c(0.8, 0.1)) +
 facet_wrap(~ country_org, scales = "free_x")

# filter data (general visualization)
filtered_data <- dfm_intervention %>% filter(agenda_item_manual %in%
agenda_item_order) %>%
 mutate(agenda_item_manual = case_when(
 agenda_item_manual %in% c("Sudan/South Sudan", "Sudan") ~ "Sudan",
 TRUE ~ agenda_item_manual)) %>%
 mutate(period = case_when(year >= 1992 & year <= 2001 ~ "1992-2001",
                year > 2002 & year <= 2011 ~ "2002-2011",
                year > 2012 & year <= 2023 ~ "2012-2023",
                TRUE ~ NA_character_  )) %>%
```

```
 filter(!is.na(period)) %>% group_by(agenda_item_manual, period) %>%
 summarise(mean_fit = median(fit, na.rm = TRUE), .groups = 'drop') %>%
 mutate(agenda_item_manual = factor(agenda_item_manual, levels =
agenda_item_order)) %>%
 mutate(mean_fit_category = factor(ifelse(mean_fit < 0, "CM", ifelse(mean_fit > 0, "CR",
NA)),
                                   levels = c("CM", "CR")))

# visualize
ggplot(filtered_data, aes(x = period, y = agenda_item_manual, fill = mean_fit_category)) +
 geom_tile(color = "black") +
 scale_fill_manual(values = colors, name = "Mean outcome", labels = c("CM" = "CM", "CR"
= "CR")) +
 labs(title = "", x = "", y = "") +
 theme_minimal() +
 geom_text(aes(label = round(mean_fit, 2)), color = "white", size = 4) +
 scale_y_discrete(labels = custom_country_labels) + # This line adds custom labels
 theme(axis.text.x = element_text(angle = 0, hjust = 0.5, size = 11),
       axis.text.y = element_text(size = 11),
       axis.title.x = element_text(size = 9),
       strip.text = element_text(size = 11),
       legend.position = "bottom")


#------------------------------------------------------------------------------

# 12) Analysis: Most polarized cases

# calculate polarization of countries
median_by_country <- dfm_theme_intervention %>%
 group_by(country_org, agenda_item_manual, year) %>%
 summarise(median_fit = median(mean_fit, na.rm = TRUE))
agenda_item3_range <- median_by_country %>%
 group_by(agenda_item_manual) %>%
 summarise(
   n = n(),
   polarization_range = max(median_fit) - min(median_fit),
   sd_polarization_range = if(n > 2) sd(median_fit) else NA_real_,
   se_polarization_range = if(n > 2) sd_polarization_range / sqrt(n) else NA_real_,
   ci_lower = if(n > 2) {
     polarization_range - (qt(0.975, df = n-1) * se_polarization_range)
   } else NA_real_,
   ci_upper = if(n > 2) {
     polarization_range + (qt(0.975, df = n-1) * se_polarization_range)
```

```
  } else NA_real_)

# visualization
most_polarized_countries <- c(
 "Sierra Leone", "Burundi", "Bosnia And Herzegovina", "Iraq", "Liberia",
 "Israel/Syria", "Yemen", "Libya", "Syria", "Haiti")
least_polarized_countries <- c(
 "Guinea", "Guatemala", "Croatia", "Central African Republic/Chad", "Nepal",
 "Guinea-Bissau", "Cambodia", "Côte D'ivoire", "Western Sahara", "Papua New Guinea")

country_labels <- c("Sierra Leone" = "Sierra Leone", "Burundi" = "Burundi",
                    "Bosnia And Herzegovina" = "Bosnia-Herzegovina", "Iraq" = "Iraq",
                    "Liberia" = "Liberia","Israel/Syria" = "Israel/Syria", "Yemen" = "Yemen",
                    "Libya" = "Libya", "Syria" = "Syria", "Haiti" = "Haiti",
                    "Guinea" = "Guinea", "Papua New Guinea" = "Papua New Guinea",
                    "Guatemala" = "Guatemala", "Croatia" = "Croatia",
                    "Central African Republic/Chad" = "CAR/Chad",
                    "Nepal" = "Nepal", "Guinea-Bissau" = "Guinea-Bissau",
                    "Western Sahara" = "Western Sahara", "Cambodia" = "Cambodia",
                    "Côte D'ivoire" = "Côte d'Ivoire")
country_order <- c("Yemen", "Western Sahara", "Syria", "Sierra Leone", "Papua New Guinea",
                   "Nepal", "Libya", "Liberia", "Israel/Syria", "Iraq", "Haiti",
                   "Guinea-Bissau", "Guinea", "Guatemala", "Croatia",
                   "Côte D'ivoire", "Central African Republic/Chad",
                   "Cambodia","Burundi", "Bosnia And Herzegovina")
# Process the timeline data
timeline_data <- dfm_intervention %>%
 filter(agenda_item_manual %in% country_order) %>%
 group_by(agenda_item_manual) %>%
 summarize(first_year = min(year),
           last_year = max(year),
           years_discussed = n_distinct(year),
           mean_polarization = mean(abs(fit), na.rm = TRUE)) %>%
 mutate(polarization_category = case_when(
    agenda_item_manual %in% most_polarized_countries ~ "Most polarized",
    agenda_item_manual %in% least_polarized_countries ~ "Least polarized"),
  display_label = country_labels[agenda_item_manual],
  last_year = if_else(agenda_item_manual == c("Guatemala", "Guinea"),
                      first_year + 0.5,
                      last_year)) %>%
 mutate(display_label = factor(display_label, levels = country_labels[country_order]))

ggplot(timeline_data, aes(x = first_year, xend = last_year, y = display_label, yend =
display_label, color = polarization_category)) +
```

```
 geom_segment(size = 1, alpha = 0.7) +
 labs(title = "",
     subtitle = "",
     x = "Year",
     y = "",
     color = "Polarization category") +
 theme_minimal() +
 theme(axis.text.x = element_text(angle = 0, hjust = 0.5, size = 11),
      axis.text.y = element_text(size = 11),
      axis.title.x = element_text(size = 13),
      legend.text = element_text(size = 11),
      strip.text = element_text(size = 11),
      legend.position = "bottom") +
 scale_x_continuous(breaks = seq(1995, 2020, by = 5)) +
 scale_color_manual(values = c("Most polarized" = "#404040", "Least polarized" =
"#A9A9A9"))

additional_lines <- data.frame(
 x_start = c(2004, 2001),
 x_end = c(2015, 2011),
 y_pos = c("Burundi", "Iraq"),
 color = rep("red", 2)
)

ggplot(timeline_data, aes(x = first_year, xend = last_year, y = display_label, yend =
display_label, color = polarization_category)) +
 geom_segment(size = 1, alpha = 0.7) +
 # Add the two specific dotted red lines
 geom_segment(data = additional_lines,
           aes(x = x_start, xend = x_end,
              y = paste0(y_pos, " "),
              yend = paste0(y_pos, " ")),
           color = "red",
           linetype = "dotted",
           size = 1) +
 labs(title = "",
     subtitle = "",
     x = "Year",
     y = "",
     color = "Polarization category") +
 theme_minimal() +
 theme(axis.text.x = element_text(angle = 0, hjust = 0.5, size = 11),
      axis.text.y = element_text(size = 11),
      axis.title.x = element_text(size = 13),
```

```
      legend.text = element_text(size = 11),
      strip.text = element_text(size = 11),
      legend.position = "bottom") +
  scale_x_continuous(breaks = seq(1995, 2020, by = 5)) +
  scale_color_manual(values = c("Most polarized" = "#404040", "Least polarized" =
"#A9A9A9"))
```