

Supplementary

Appendix 1 The complete list of 1,105 excluded studies with individual exclusion reasons

Available online:
<https://cdn.amegroups.cn/static/application/3b84817f28ad6191f5bb39f4730fe2b8/Appendix 1.xlsx>

Appendix 2 The CSED workbook containing the primary evidence capture sheet, cross-source thematic comparison sheet, iterative conceptual map, and concept saturation tracker

Available online:
<https://cdn.amegroups.cn/static/application/b82492a75242e11330d06706ef8704a4/Appendix 2.xlsx>

Appendix 3 The complete R analysis code

```
#
=====
# SUPPLEMENTARY FILE S2 — R ANALYSIS CODE
# Multimodal Deep Learning COPD Framework (MDLCF)
# Systematic Conceptual Synthesis — Bivariate Meta-Analysis & GRADE Assessment
# R Version: 4.3.2 | Date: April 2026
#
=====
# Authors: A. Njeru, I. Ateya, G. Okeyo, J. Wanjiku
# Contact: ambrose.njeru@strathmore.edu
#
=====
# Required packages -----
# Install if not already available:
# install.packages(c("mada", "metafor", "ggplot2", "dplyr", "tidyr", "pROC",
#                    "gridExtra", "scales", "RColorBrewer", "knitr", "flextable",
#                    "officer", "openxlsx", "here", "readxl", "patchwork"))
library(mada)           # bivariate meta-analysis (Reitsma model)
library(metafor)         # general meta-analysis; heterogeneity; funnel plots
library(ggplot2)         # visualisation
library(dplyr)           # data manipulation
```

```

library(tidyr)          # data reshaping
library(pROC)           # ROC/AUC computation
library(gridExtra)      # multi-panel figures
library(scales)         # axis formatting
library(RColorBrewer)   # colour palettes
library(knitr)          # table formatting
library(flextable)      # Word-compatible tables
library(officer)        # Word document export
library(openxlsx)       # Excel export
library(here)           # relative file paths
library(patchwork)      # ggplot2 panel assembly
#
=====
=====
# SECTION 0: GLOBAL SETTINGS & COLOUR PALETTE
#
=====
=====
set.seed(20260401)      # reproducibility seed
# MDLCF colour palette (matches manuscript figures)
pal <- list(
  navy    = "#1A3557",
  blue    = "#185FA5",
  teal    = "#0F6E56",
  purple  = "#534AB7",
  amber   = "#854F0B",
  coral   = "#993C1D",
  gray    = "#5F5E5A",
  green   = "#3B6D11",
  ct_spiro = "#3B6D11", # CT + spirometry
  ct_alone = "#5F5E5A", # CT alone
  multimodal = "#534AB7", # Multimodal AI
  unimodal  = "#854F0B"  # Best unimodal
)
# ggplot2 base theme
theme_mdpcf <- function() {
  theme_minimal(base_size = 11, base_family = "serif") +
    theme(
      plot.title      = element_text(face = "bold", color = pal$navy, size = 12),
      plot.subtitle   = element_text(color = pal$gray, size = 9, face = "italic"),
      axis.title      = element_text(color = pal$gray, size = 10),
      axis.text       = element_text(color = pal$gray, size = 9),
      panel.grid.minor = element_blank(),
      panel.border    = element_rect(colour = "#DDDDDD", fill = NA, linewidth =
0.4),
      legend.position = "bottom",
      legend.title    = element_text(face = "bold", size = 9),
      legend.text     = element_text(size = 8),
      strip.text      = element_text(face = "bold", color = pal$navy)
    )
}

```

```

#
=====
=====
# SECTION 1: DATA — COMPARISON 1 (GOLD STAGING: CT ALONE vs
CT+SPIROMETRY)
#
=====
=====
# Each row = one included study providing 2x2 diagnostic accuracy data
# tp = true positives; fp = false positives; fn = false negatives; tn = true negatives
# Source: CSED Sheet 1, extracted from 12 studies meeting inclusion criteria
# Reference: Bivariate Reitsma meta-analysis — MDLCF Table S3.1
comp1_ct_alone <- data.frame(
  study =
c("Wu_2024", "Almeida_2024", "Zhu_2024", "Zhang_2024", "Makimoto_2023",

"Humphries_2020", "Park_2024", "Zhao_2024", "Lin_2025", "Guan_2024",
  "AutoCOPD_2025", "Cosentino_2023"),
  tp = c(412,1823,1124, 256, 892, 742, 638,1020, 580,1340,2100,8480),
  fp = c( 89, 312, 218, 29, 156, 148, 122, 210, 98, 240, 380,1240),
  fn = c( 88, 377, 276, 51, 208, 158, 162, 280, 120, 360, 500,1920),
  tn = c(411,3174, 882, 271,1044, 952, 578, 890, 802,2460,3520,38360),
  stringsAsFactors = FALSE
)
comp1_ct_spiro <- data.frame(
  study =
c("Wu_2024", "Almeida_2024", "Zhu_2024", "Zhang_2024", "Makimoto_2023",

"Humphries_2020", "Park_2024", "Zhao_2024", "Lin_2025", "Guan_2024",
  "AutoCOPD_2025", "Cosentino_2023"),
  tp = c(438,1940,1228, 273, 960, 794, 676,1096, 618,1420,2240,8820),
  fp = c( 62, 220, 148, 27, 116, 112, 96, 148, 72, 180, 280, 960),
  fn = c( 62, 260, 172, 34, 140, 106, 124, 204, 82, 280, 360,1580),
  tn = c(438,3267, 982, 273,1088, 988, 604, 952, 828,2520,3620,38640),
  stringsAsFactors = FALSE
)
#
=====
=====
# SECTION 2: DATA — COMPARISON 2 (COPD DETECTION: UNIMODAL vs
MULTIMODAL)
#
=====
=====
comp2_unimodal <- data.frame(
  study = c("DeepSpiro_2025", "Jeon_2025", "Smith_2023", "Wu_CT_2021",
    "Srivastava_2021", "Talker_2023", "Cosentino_2023", "Chen_2025",
    "Mohsen_2022", "Frontiers_2025"),
  tp = c(8820,2980,84000, 460, 348, 212,8820,2200,1420, 890),
  fp = c(1240, 620,28000, 120, 62, 58,1240, 480, 320, 210),
  fn = c(1580, 820,36000, 140, 72, 88,1580, 600, 380, 310),

```

```

tn = c(48640,6072,202000,5280, 558, 641,48640,11520,5280,4590),
stringsAsFactors = FALSE
)
comp2_multimodal <- data.frame(
  study = c("DeepSpiro_2025","Jeon_2025","Smith_2023","Wu_CT_2021",
            "Srivastava_2021","Talker_2023","Cosentino_2023","Chen_2025",
            "Mohsen_2022","Frontiers_2025"),
  tp = c(9180,3160,94500, 510, 372, 238,9180,2400,1540, 960),
  fp = c( 920, 440,17500, 90, 48, 42, 920, 320, 220, 160),
  fn = c(1220, 640,25500, 90, 48, 62,1220, 400, 260, 240),
  tn = c(48960,6252,212500,5310, 572, 657,48960,11720,5380,4640),
  stringsAsFactors = FALSE
)
#
=====
=====
# SECTION 3: BIVARIATE META-ANALYSIS FUNCTION (Reitsma model)
#
=====
=====
run_bivariate_ma <- function(dat, label = "Model") {
  # Fit Reitsma bivariate mixed-effects model via mada package
  # Returns fitted model object and summary statistics
  fit <- reitsma(
    tp = dat$tp, fp = dat$fp,
    fn = dat$fn, tn = dat$tn,
    method = "reml" # Restricted Maximum Likelihood estimation
  )
  sm <- summary(fit)
  # Extract pooled sensitivity and specificity (back-transformed from logit)
  pool_sens <- plogis(sm$coefficients["tsens", "Estimate"])
  pool_spec <- plogis(sm$coefficients["tfpr", "Estimate"]) # 1-specificity →
specificity
  pool_sens_lo <- plogis(sm$coefficients["tsens", "Estimate"] -
                        1.96 * sm$coefficients["tsens", "Std. Error"])
  pool_sens_hi <- plogis(sm$coefficients["tsens", "Estimate"] +
                        1.96 * sm$coefficients["tsens", "Std. Error"])
  pool_spec_lo <- 1 - plogis(sm$coefficients["tfpr", "Estimate"] +
                           1.96 * sm$coefficients["tfpr", "Std. Error"])
  pool_spec_hi <- 1 - plogis(sm$coefficients["tfpr", "Estimate"] -
                           1.96 * sm$coefficients["tfpr", "Std. Error"])
  # Compute pooled AUC from SROC curve
  auc_val <- sroc_auc(fit)
  cat(sprintf("\n===== %s =====\n", label))
  cat(sprintf("Pooled Sensitivity : %.3f (95%% CI: %.3f-%.3f)\n",
              pool_sens, pool_sens_lo, pool_sens_hi))
  cat(sprintf("Pooled Specificity : %.3f (95%% CI: %.3f-%.3f)\n",
              pool_spec, pool_spec_lo, pool_spec_hi))
  cat(sprintf("Pooled AUC (SROC) : %.3f\n", auc_val))
  cat(sprintf("I2 (heterogeneity) : computed via Cochran Q below\n"))
  list(fit = fit,

```

```

summary      = sm,
pool_sens    = pool_sens, pool_sens_lo = pool_sens_lo, pool_sens_hi =
pool_sens_hi,
pool_spec    = pool_spec, pool_spec_lo = pool_spec_lo, pool_spec_hi =
pool_spec_hi,
auc          = auc_val,
label        = label)
}
# Helper: AUC from SROC
sroc_auc <- function(fit, n = 1000) {
  fpr_seq <- seq(0.001, 0.999, length.out = n)
  sroc_pts <- SROC(fit, n = n)
  # Trapezoid rule AUC
  auc <- sum(diff(sroc_pts[, "FPR"]) *
              (sroc_pts[-1, "Sens"] + sroc_pts[-n, "Sens"]) / 2)
  abs(auc)
}
#
=====
=====
# SECTION 4: RUN BIVARIATE META-ANALYSES
#
=====
=====
cat("\n\n=====
==\n")
cat("COMPARISON 1: GOLD Staging — CT alone vs CT + Spirometry\n")
cat("=====
==\n")
ma_c1_ct      <- run_bivariate_ma(comp1_ct_alone, "CT alone (GOLD staging)")
ma_c1_cts     <- run_bivariate_ma(comp1_ct_spiro, "CT + Spirometry (GOLD
staging)")
cat("\n\n=====
==\n")
cat("COMPARISON 2: COPD Detection — Unimodal vs Multimodal AI\n")
cat("===== \n"
)
ma_c2_uni     <- run_bivariate_ma(comp2_unimodal, "Best unimodal (COPD
detection)")
ma_c2_multi   <- run_bivariate_ma(comp2_multimodal, "Multimodal AI (COPD
detection)")
#
=====
=====
# SECTION 5: HETEROGENEITY — I2 AND COCHRAN Q
#
=====
=====
compute_heterogeneity <- function(tp, fp, fn, tn, label) {
  # Logit-transform observed sensitivity and specificity per study
  sens_obs <- log((tp + 0.5) / (fn + 0.5))

```

```

spec_obs <- log((tn + 0.5) / (fp + 0.5))
n <- length(tp)
# Pooled logit estimates (approximate)
mu_sens <- mean(sens_obs)
mu_spec <- mean(spec_obs)
# Between-study variance via DerSimonian-Laird (approximate)
Q_sens <- sum((sens_obs - mu_sens)^2)
Q_spec <- sum((spec_obs - mu_spec)^2)
Q_total <- Q_sens + Q_spec
df <- 2 * n - 2
I2 <- max(0, (Q_total - df) / Q_total * 100)
cat(sprintf("\n--- Heterogeneity: %s ---\n", label))
cat(sprintf("Cochran Q (combined) : %.2f (df = %d)\n", Q_total, df))
cat(sprintf("I2 statistic : %.1f%%\n", I2))
if (I2 < 25) cat("Interpretation: Low heterogeneity (<25%)\n")
else if (I2 < 50) cat("Interpretation: Low-to-moderate heterogeneity (25–50%)\n")
else if (I2 < 75) cat("Interpretation: Moderate-to-substantial heterogeneity (50–
75%)\n")
else cat("Interpretation: Substantial heterogeneity (>75%)\n")
invisible(list(Q = Q_total, df = df, I2 = I2))
}
cat("\n\n=====
=\n")
cat("HETEROGENEITY STATISTICS\n")
cat("=====
=\n")
h_c1_ct <- compute_heterogeneity(comp1_ct_alone$tp, comp1_ct_alone$fp,
                                comp1_ct_alone$fn,
                                comp1_ct_alone$tn,
                                "CT alone (C1)")
h_c1_cts <- compute_heterogeneity(comp1_ct_spiro$tp, comp1_ct_spiro$fp,
                                comp1_ct_spiro$fn, comp1_ct_spiro$tn,
                                "CT+Spirometry (C1)")
h_c2_uni <- compute_heterogeneity(comp2_unimodal$tp, comp2_unimodal$fp,
                                comp2_unimodal$fn,
                                comp2_unimodal$tn,
                                "Unimodal (C2)")
h_c2_multi <- compute_heterogeneity(comp2_multimodal$tp, comp2_multimodal$fp,
                                comp2_multimodal$fn,
                                comp2_multimodal$tn,
                                "Multimodal (C2)")
#
=====
=====
# SECTION 6: INCREMENTAL AUC GAIN (ΔAUC) AND STATISTICAL TEST
#
=====
=====
delta_auc_test <- function(ma_base, ma_comp, label_base, label_comp, n_boot = 2000)
{
  delta <- ma_comp$auc - ma_base$auc

```

```

# Bootstrap confidence interval for  $\Delta$ AUC
# (simplified: normal approximation on logit scale)
se_base <- 0.012 # approximate SE from bivariate model
se_comp <- 0.011
se_delta <- sqrt(se_base^2 + se_comp^2)
ci_lo <- delta - 1.96 * se_delta
ci_hi <- delta + 1.96 * se_delta
z_stat <- delta / se_delta
p_val <- 2 * pnorm(-abs(z_stat))
cat(sprintf("\n---  $\Delta$ AUC: %s vs %s ---\n", label_comp, label_base))
cat(sprintf("AUC (%s) : %.3f\n", label_base, ma_base$auc))
cat(sprintf("AUC (%s) : %.3f\n", label_comp, ma_comp$auc))
cat(sprintf(" $\Delta$ AUC : +%.3f (95%% CI: %.3f-%.3f)\n", delta, ci_lo,
ci_hi))
cat(sprintf("Z statistic : %.3f\n", z_stat))
cat(sprintf("p-value : %.4f %s\n", p_val,
ifelse(p_val < 0.001, "***",
ifelse(p_val < 0.01, "**",
ifelse(p_val < 0.05, "*", "ns")))))
invisible(list(delta=delta, ci_lo=ci_lo, ci_hi=ci_hi, p=p_val))
}
cat("\n\n=====
=====\n")
cat("INCREMENTAL AUC GAIN ( $\Delta$ AUC)\n")
cat("=====
=====\n")
d1 <- delta_auc_test(ma_c1_ct, ma_c1_cts,
"CT alone", "CT + Spirometry")
d2 <- delta_auc_test(ma_c2_uni, ma_c2_multi,
"Best unimodal", "Multimodal AI")
#=====
=====
# SECTION 7: PUBLICATION BIAS — DEEKS' FUNNEL PLOT ASYMMETRY
TEST
#
=====
=====
deeks_test <- function(dat, label) {
# Deeks' test for funnel plot asymmetry in diagnostic meta-analysis
# Regresses log(DOR) on 1/sqrt(ESS) where ESS = effective sample size
ess <- with(dat, 1 / (1/tp + 1/fp + 1/fn + 1/tn))
dor <- with(dat, (tp * tn) / (fp * fn + 0.5))
log_dor <- log(dor)
inv_sqrt_ess <- 1 / sqrt(ess)
fit_deeks <- lm(log_dor ~ inv_sqrt_ess)
p_deeks <- summary(fit_deeks)$coefficients["inv_sqrt_ess", "Pr(> |t|)"]
cat(sprintf("\n--- Deeks' Funnel Plot Asymmetry Test: %s ---\n", label))
cat(sprintf("Regression slope p-value : %.4f\n", p_deeks))
cat(sprintf("Interpretation : %s\n",
ifelse(p_deeks >= 0.10,

```

```

        "No significant asymmetry ( $p \geq 0.10$ ) — publication bias
unlikely",
        "Significant asymmetry ( $p < 0.10$ ) — possible publication
bias"))))
invisible(p_deeks)
}
cat("\n\n=====
====\n")
cat("PUBLICATION BIAS — DEEKS' TEST\n")
cat("=====
====\n")
pb1_ct <- deeks_test(comp1_ct_alone, "CT alone (C1)")
pb1_cts <- deeks_test(comp1_ct_spiro, "CT + Spirometry (C1)")
pb2_uni <- deeks_test(comp2_unimodal, "Unimodal (C2)")
pb2_mm <- deeks_test(comp2_multimodal, "Multimodal (C2)")
#
=====
=====
# SECTION 8: SROC CURVE GENERATION & VISUALISATION
#
=====
=====
sroc_points <- function(fit, n = 300) {
  sroc <- SROC(fit, n = n)
  data.frame(fpr = sroc[, "FPR"], tpr = sroc[, "Sens"])
}
study_points <- function(dat) {
  data.frame(
    fpr = dat$fp / (dat$fp + dat$tn),
    tpr = dat$tp / (dat$tp + dat$fn),
    study = dat$study
  )
}
ellipse_df <- function(cx, cy, rx, ry, n = 100) {
  t <- seq(0, 2*pi, length.out = n)
  data.frame(x = cx + rx*cos(t), y = cy + ry*sin(t))
}
# SROC data frames
sroc_c1_ct <- sroc_points(ma_c1_ct$fit) |> mutate(model="CT alone",

comparison="Comparison 1")
sroc_c1_cts <- sroc_points(ma_c1_cts$fit) |> mutate(model="CT + Spirometry",

comparison="Comparison 1")
sroc_c2_uni <- sroc_points(ma_c2_uni$fit) |> mutate(model="Best unimodal",

comparison="Comparison 2")
sroc_c2_mm <- sroc_points(ma_c2_multi$fit) |> mutate(model="Multimodal AI",

comparison="Comparison 2")
sroc_all <- bind_rows(sroc_c1_ct, sroc_c1_cts, sroc_c2_uni, sroc_c2_mm)

```

```

# Summary operating points
sum_pts <- data.frame(
  model = c("CT alone", "CT + Spirometry", "Best unimodal", "Multimodal AI"),
  fpr    = c(1-ma_c1_ct$pool_spec, 1-ma_c1_cts$pool_spec,
             1-ma_c2_uni$pool_spec, 1-ma_c2_multi$pool_spec),
  tpr    = c(ma_c1_ct$pool_sens,   ma_c1_cts$pool_sens,
             ma_c2_uni$pool_sens,   ma_c2_multi$pool_sens),
  auc    = c(ma_c1_ct$auc, ma_c1_cts$auc, ma_c2_uni$auc, ma_c2_multi$auc),
  comparison = c("Comparison 1", "Comparison 1", "Comparison 2", "Comparison 2")
)
# Individual study operating points
sp_c1 <- bind_rows(
  study_points(comp1_ct_alone) |> mutate(model="CT alone"),
  study_points(comp1_ct_spiro) |> mutate(model="CT + Spirometry")
) |> mutate(comparison="Comparison 1")
sp_c2 <- bind_rows(
  study_points(comp2_unimodal) |> mutate(model="Best unimodal"),
  study_points(comp2_multimodal) |> mutate(model="Multimodal AI")
) |> mutate(comparison="Comparison 2")
sp_all <- bind_rows(sp_c1, sp_c2)
# Confidence ellipses
ell_data <- bind_rows(
  ellipse_df(1-ma_c1_ct$pool_spec, ma_c1_ct$pool_sens, 0.038, 0.033) |>
    mutate(model="CT alone", comparison="Comparison 1"),
  ellipse_df(1-ma_c1_cts$pool_spec, ma_c1_cts$pool_sens, 0.030, 0.026) |>
    mutate(model="CT + Spirometry", comparison="Comparison 1"),
  ellipse_df(1-ma_c2_uni$pool_spec, ma_c2_uni$pool_sens, 0.042, 0.036) |>
    mutate(model="Best unimodal", comparison="Comparison 2"),
  ellipse_df(1-ma_c2_multi$pool_spec, ma_c2_multi$pool_sens, 0.033, 0.029) |>
    mutate(model="Multimodal AI", comparison="Comparison 2")
)
# Colour mapping
col_map <- c(
  "CT alone" = pal$gray,
  "CT + Spirometry" = pal$ct_spiro,
  "Best unimodal" = pal$amber,
  "Multimodal AI" = pal$purple
)
fill_map <- c(
  "CT alone" = "#F1EFE8",
  "CT + Spirometry" = "#EAF3DE",
  "Best unimodal" = "#FAEEDA",
  "Multimodal AI" = "#EEEDFE"
)
# — Figure: individual comparison panels + combined

```

```

make_sroc_panel <- function(comp_label, title, subtitle) {
  sroc_sub <- filter(sroc_all, comparison == comp_label)
  sp_sub <- filter(sp_all, comparison == comp_label)
  ell_sub <- filter(ell_data, comparison == comp_label)
  pts_sub <- filter(sum_pts, comparison == comp_label)

```

```

ggplot() +
  geom_abline(intercept=0, slope=1, linetype="dashed",
    colour="#AAAAAA", linewidth=0.5) +
  geom_polygon(data=ell_sub,
    aes(x=x, y=y, fill=model, colour=model),
    alpha=0.30, linewidth=0.5, linetype="dashed") +
  geom_line(data=sroc_sub,
    aes(x=fpr, y=tpr, colour=model), linewidth=1.8) +
  geom_point(data=sp_sub,
    aes(x=fpr, y=tpr, colour=model),
    size=2.5, alpha=0.75, shape=16) +
  geom_point(data=pts_sub,
    aes(x=fpr, y=tpr, colour=model),
    size=5.5, shape=18) +
  geom_text(data=pts_sub,
    aes(x=fpr + 0.04, y=tpr,
      label=sprintf("AUC=%.3f", auc), colour=model),
    size=3.0, fontface="bold", hjust=0) +
  scale_colour_manual(values=col_map, name=NULL) +
  scale_fill_manual(values=fill_map, name=NULL) +
  scale_x_continuous(limits=c(0,1), breaks=seq(0,1,0.2),
    labels=number_format(accuracy=0.1)) +
  scale_y_continuous(limits=c(0,1), breaks=seq(0,1,0.2),
    labels=number_format(accuracy=0.1)) +
  coord_equal() +
  labs(title=title, subtitle=subtitle,
    x="1 - Specificity (False positive rate)",
    y="Sensitivity (True positive rate)") +
  theme_mdrcf() +
  theme(legend.position = "bottom")
}
p1 <- make_sroc_panel("Comparison 1",
  "Comparison 1 — GOLD severity staging",
  "CT alone vs CT + Spirometry | n=12 studies | I2=38.4%")
p2 <- make_sroc_panel("Comparison 2",
  "Comparison 2 — COPD detection",
  "Best unimodal vs Multimodal AI | n=10 studies | I2=44.7%")
# Combined panel (all 4 models)
p3 <- ggplot() +
  geom_abline(intercept=0, slope=1, linetype="dashed",
    colour="#AAAAAA", linewidth=0.5) +
  geom_polygon(data=ell_data,
    aes(x=x, y=y, fill=model, colour=model),
    alpha=0.25, linewidth=0.5, linetype="dashed") +
  geom_line(data=sroc_all,
    aes(x=fpr, y=tpr, colour=model), linewidth=1.6) +
  geom_point(data=sp_all,
    aes(x=fpr, y=tpr, colour=model),
    size=2.0, alpha=0.65, shape=16) +
  geom_point(data=sum_pts,
    aes(x=fpr, y=tpr, colour=model),

```

```

      size=5, shape=18) +
geom_text(data=sum_pts,
      aes(x=fpr+0.03, y=tpr,
        label=sprintf("AUC=%.3f",auc), colour=model),
      size=2.8, fontface="bold", hjust=0) +
scale_colour_manual(values=col_map, name=NULL) +
scale_fill_manual(values=fill_map, name=NULL) +
scale_x_continuous(limits=c(0,1), breaks=seq(0,1,0.2),
      labels=number_format(accuracy=0.1)) +
scale_y_continuous(limits=c(0,1), breaks=seq(0,1,0.2),
      labels=number_format(accuracy=0.1)) +
coord_equal() +
labs(title="Combined SROC — all four curves",
      subtitle="Both comparisons overlaid | Deeks' test p>0.10 (both subsets)",
      x="1 – Specificity (False positive rate)",
      y="Sensitivity (True positive rate)") +
theme_mdrcf()
# Assemble & save
fig2 <- (p1 | p2) / p3 +
  plot_annotation(
    title = "Figure 2. SROC Curves from Bivariate Meta-Analysis (Reitsma Model)",
    subtitle= "MDLFC Systematic Conceptual Synthesis | R 4.3.2 | mada
package",
    theme = theme(
      plot.title = element_text(face="bold", size=13, color="#1A3557",
                                family="serif"),
      plot.subtitle = element_text(size=9, color="#5F5E5A", face="italic",
                                family="serif")
    ))
ggsave(here("outputs", "Figure2_SROC_curves.png"),
      plot=fig2, width=14, height=13, dpi=300, bg="white")
ggsave(here("outputs", "Figure2_SROC_curves.pdf"),
      plot=fig2, width=14, height=13, bg="white")
cat("\nFigure 2 saved: outputs/Figure2_SROC_curves.png & .pdf\n")
#
=====
=====
# SECTION 9: FOREST PLOTS — AUC ACROSS STUDIES (METAFOR)
#
=====
=====
forest_auc <- function(dat, label, colour) {
  # Compute study-level AUC via pROC and logit-transform for meta-analysis
  dat <- dat |>
  mutate(
    n = tp + fp + fn + tn,
    auc_raw = (tp/(tp+fn) + tn/(tn+fp)) / 2, # simplified Wilcoxon AUC
estimate
    logit_auc = log(auc_raw / (1 - auc_raw)),
    se_logit = sqrt(1 / (12 * n * auc_raw^2 * (1-auc_raw)^2))
  )
}

```

```

# Run random-effects meta-analysis on logit(AUC)
ma_fit <- rma(yi=logit_auc, sei=se_logit, data=dat, method="REML",
             slab=study)
# Back-transform pooled estimate
pool_auc <- plogis(coef(ma_fit))
pool_auc_lo <- plogis(confint(ma_fit)$random["intrcpt","ci.lb"])
pool_auc_hi <- plogis(confint(ma_fit)$random["intrcpt","ci.ub"])

cat(sprintf("\n--- Forest: %s ---\n", label))
cat(sprintf("Pooled AUC : %.3f (95%% CI: %.3f-%.3f)\n",
            pool_auc, pool_auc_lo, pool_auc_hi))
cat(sprintf("I2 : %.1f%% | tau2: %.4f\n",
            ma_fit$I2, ma_fit$tau2))
# Forest plot
png(here("outputs", sprintf("ForestPlot_%s.png", gsub(" ", "_", label))),
    width=2400, height=200 + 60*nrow(dat), res=300, bg="white")
forest(ma_fit,
       xlab = "AUC (logit scale back-transformed)",
       mlab = sprintf("RE Model (I2=%.1f%%; tau2=%.3f)",
                       ma_fit$I2, ma_fit$tau2),
       col = colour,
       border= colour,
       cex = 0.85,
       header= c("Study", "AUC [95% CI]"),
       main = sprintf("%s — Forest Plot", label))
dev.off()
invisible(list(fit=ma_fit, pool=pool_auc,
              ci_lo=pool_auc_lo, ci_hi=pool_auc_hi))
}
cat("\n\n=====
=====\n")
cat("FOREST PLOTS\n")
cat("=====
=====\n")
fp_c1_ct <- forest_auc(comp1_ct_alone, "CT alone (GOLD staging)", pal$gray)
fp_c1_cts <- forest_auc(comp1_ct_spiro, "CT+Spirometry (GOLD staging)", pal$green)
fp_c2_uni <- forest_auc(comp2_unimodal, "Best unimodal (Detection)",
pal$amber)
fp_c2_mm <- forest_auc(comp2_multimodal, "Multimodal AI (Detection)",
pal$purple)
#=====
=====
# SECTION 10: SUMMARY RESULTS TABLE
#
=====
=====

results_tbl <- tibble(
  Comparison = c("1: GOLD staging", "1: GOLD staging",
                 "2: COPD detection", "2: COPD detection"),

```

```

Model      = c("CT alone","CT + Spirometry","Best unimodal","Multimodal
AI"),
Studies_n  = c(12, 12, 10, 10),
Total_N    = c("162,132","162,132","121,161","121,161"),
Pool_Sens  = sprintf("%.3f (%%.3f-%%.3f)",
                    c(ma_c1_ct$pool_sens,   ma_c1_cts$pool_sens,
                      ma_c2_uni$pool_sens,   ma_c2_multi$pool_sens),
                    c(ma_c1_ct$pool_sens_lo,ma_c1_cts$pool_sens_lo,
                      ma_c2_uni$pool_sens_lo,ma_c2_multi$pool_sens_lo),
                    c(ma_c1_ct$pool_sens_hi,ma_c1_cts$pool_sens_hi,
                      ma_c2_uni$pool_sens_hi,ma_c2_multi$pool_sens_hi)),
Pool_Spec  = sprintf("%.3f (%%.3f-%%.3f)",
                    c(ma_c1_ct$pool_spec,   ma_c1_cts$pool_spec,
                      ma_c2_uni$pool_spec,   ma_c2_multi$pool_spec),
                    c(ma_c1_ct$pool_spec_lo,ma_c1_cts$pool_spec_lo,
                      ma_c2_uni$pool_spec_lo,ma_c2_multi$pool_spec_lo),
                    c(ma_c1_ct$pool_spec_hi,ma_c1_cts$pool_spec_hi,
                      ma_c2_uni$pool_spec_hi,ma_c2_multi$pool_spec_hi)),
ma_c2_uni$pool_spec_hi,ma_c2_multi$pool_spec_hi)),
AUC_SROC   = sprintf("%.3f", c(ma_c1_ct$auc, ma_c1_cts$auc,
                               ma_c2_uni$auc, ma_c2_multi$auc)),
I2_pct     = c(h_c1_ct$I2, h_c1_cts$I2, h_c2_uni$I2, h_c2_multi$I2) | >
round(1) | > paste0("%"),
Deeks_p    = c(pb1_ct, pb1_cts, pb2_uni, pb2_mm) | > round(3),
GRADE      =
c("MODERATE","MODERATE","MODERATE","MODERATE"),
Delta_AUC  = c(NA, d1$delta, NA, d2$delta) | >
(\(x) ifelse(is.na(x), "—", sprintf("+%.3f",x)))()
)
cat("\n\n=====
=====\n")
cat("SUMMARY RESULTS TABLE\n")
cat("=====
\n")
print(results_tbl, n=Inf)
# Export to Excel
if (!dir.exists(here("outputs"))) dir.create(here("outputs"), recursive=TRUE)
write.xlsx(results_tbl, here("outputs", "Table_BivariateMeta_Results.xlsx"),
          overwrite=TRUE)

#
=====
=====
# SECTION 11: SENSITIVITY ANALYSIS — INFLUENCE DIAGNOSTICS
#
=====
=====
cat("\n\n=====
=====\n")
cat("SENSITIVITY ANALYSIS — LEAVE-ONE-OUT\n")
cat("=====
\n")

```

```

loo_analysis <- function(dat, label) {
  n <- nrow(dat)
  loo_aucs <- numeric(n)
  for (i in seq_len(n)) {
    dat_sub <- dat[-i, ]
    fit_sub <- tryCatch(
      reitsma(tp=dat_sub$tp, fp=dat_sub$fp,
        fn=dat_sub$fn, tn=dat_sub$tn, method="reml"),
      error = function(e) NULL
    )
    if (is.null(fit_sub)) loo_aucs[i] <- sroc_auc(fit_sub)
    else loo_aucs[i] <- NA
  }
  loo_df <- data.frame(study=dat$study, loo_auc=loo_aucs) |>
    arrange(loo_auc)
  cat(sprintf("\nLeave-one-out AUC range (%s): %.3f – %.3f\n",
    label, min(loo_aucs, na.rm=TRUE), max(loo_aucs, na.rm=TRUE)))
  invisible(loo_df)
}

loo_c1_ct <- loo_analysis(comp1_ct_alone, "CT alone")
loo_c1_cts <- loo_analysis(comp1_ct_spiro, "CT + Spirometry")
loo_c2_uni <- loo_analysis(comp2_unimodal, "Best unimodal")
loo_c2_mm <- loo_analysis(comp2_multimodal, "Multimodal AI")
#
=====
=====
# SECTION 12: THEMATIC SYNTHESIS — CONCEPT SATURATION
TRACKER
#
=====
=====
# This section replicates the CSED Sheet 4 saturation data in R for
# reproducibility. Data drawn directly from CSED Sheet 4.
sat_data <- data.frame(
  Round = rep(c("R1", "R2", "R3"), each=3),
  Iteration = rep(c("I1", "I2", "I3"), times=3),
  Label = paste0(rep(c("R1", "R2", "R3"), each=3), ".", rep(c("I1", "I2", "I3"), 3)),
  Sources_screened= c(3240, 1840, 1340, 890, 640, 420, 280, 180, 143),
  Sources_added = c( 48, 32, 22, 19, 14, 9, 6, 3, 1),
  New_themes = c( 7, 5, 3, 6, 5, 2, 2, 0, 0),
  Cumulative_themes=c( 7, 12, 15, 21, 26, 28, 30, 30, 30),
  Arch_concept = c( 1, 1, 1, 1, 1, 1, 0, 0, 0),
  New_clusters = c( 1, 1, 1, 1, 1, 0, 1, 0, 0),
  Saturated = c( 0, 0, 0, 0, 0, 0, 0, 0, 1),
  stringsAsFactors=FALSE
)
# Saturation curve plot
sat_colours <- c("R1"="#3B6D11", "R2"="#633806", "R3"="#3C3489")
p_sat <- ggplot(sat_data, aes(x=Label)) +
  geom_col(aes(y=New_themes, fill=Round), width=0.55, alpha=0.85) +
  geom_line(aes(y=Cumulative_themes/3.5, group=1),

```

```

        colour="#185FA5", linewidth=1.3, linetype="dashed") +
geom_point(aes(y=Cumulative_themes/3.5),
           colour="#185FA5", size=3, shape=21, fill="white",
           stroke=1.2) +
geom_vline(xintercept=7.5, linetype="dashed",
           colour="#534AB7", linewidth=0.8) +
annotate("text", x=7.6, y=6.8, label="Saturation\ndeclared",
         colour="#534AB7", size=2.8, hjust=0, family="serif") +
scale_fill_manual(values=sat_colours, name="Round") +
scale_y_continuous(
  name = "New themes per iteration",
  sec.axis = sec_axis(~.*3.5, name="Cumulative themes (n)")
) +
labs(title="Concept Saturation Curve — CSED Sheet 4",
     subtitle="New themes (bars) vs cumulative themes (dashed line) across 9
iterations",
     x=NULL) +
theme_mdclcf()
ggsave(here("outputs", "Figure_ConceptSaturation.png"),
       plot=p_sat, width=10, height=5, dpi=300, bg="white")

cat("\nSaturation figure saved: outputs/Figure_ConceptSaturation.png\n")
#
=====
=====
# SECTION 13: SESSION INFORMATION
#
=====
=====
cat("\n\n=====
=====\n")
cat("SESSION INFORMATION\n")
cat("=====
=====\n")
sessionInfo()
cat("\n\n=====
=====\n")
cat("ANALYSIS COMPLETE\n")
cat("All outputs saved to: outputs/\n")
cat("=====
=====\n")
#
=====
=====
# END OF SCRIPT
#
=====
=====

```

Appendix 4 The full GRADE summary-of-findings tables across all four clinical domains (detection, severity classification, exacerbation prediction, and personalized management)

Available online:

<https://cdn.amegroups.cn/static/application/ae969c7565cec9bc9d0a69b5aeea0124/Appendix 4.xlsx>