M21 RT Orthographic Sensitivity

Joanna Morris

2025-11-06

Setup

Load libraries

1. Set `ggplot2` parameters

Load Files and Format Files

Load Files

```
#DIR <- "csv_files"
df_a <- read_csv("rt_data_hc_A.csv")
df_b <- read_csv( "rt_data_hc_B_fixed.csv")
frq_w <- read_csv("frq_cw.csv")
frq_nw <- read_csv("frq_nw.csv")
dmg <- read_csv("demo_lang_vsl_pca_hc.csv")
```

Format Files

```
# Concatenate datasets
rt <- bind_rows(AB = df_a,
                BA = df_b,
               .id = "List")
rt_dmg<- right_join(dmg, rt, join_by(SubjID == subject_nr)) |>  # Join Participant Demographic and Lang Data
                    mutate(target = tolower(target)) |>
                    filter(correct == 1)

# Divide into Experimental and Filler Items
rt_fill <- rt_dmg |> filter(str_detect(targ_type, "^FILL"))
rt_exp <- rt_dmg |> filter(!str_detect(targ_type, "^FILL"))

# Add logFS to frequency datasets
frq_w <- frq_w |> mutate(Log10FS = log10(FS))
frq_nw <- frq_nw |> mutate(Log10FS = log10(FS))

# Define Factors and Conditions
rt_exp_format <- rt_exp |>
  separate(targ_type, into = c("trial_type", "family_size", "complexity"), sep = "_",
           remove = TRUE, extra = "drop", fill = "right")

# Divide into Words and Nonwords
rt_words <- rt_exp_format |> filter(trial_type == "CW") |> select(- complexity)
rt_nwords <- rt_exp_format |> filter(trial_type == "NW")

# Join Stimulus Frequency Data
rt_words_frq <- left_join(rt_words, frq_w, join_by(target))|>
  select(-cond_trig.y, -word_trig.y) |>
  rename(cond_trig = cond_trig.x, word_trig = word_trig.x)  # remove duplicate columns
rt_nwords_frq <-  left_join(rt_nwords, frq_nw, join_by(target==word)) |>
  select(-cond_trig.y, -word_trig.y) |>
  rename(cond_trig = cond_trig.x, word_trig = word_trig.x)

# Rename BF_Split and FS_Split columns
rt_words_frq <- rt_words_frq |> rename(Base_Frequency = BF_Split, Family_Size = FS_Split) # Rename BF_Split and FS_Split columns
rt_nwords_frq <- rt_nwords_frq |> rename(Base_Frequency = BF_Split, Family_Size = FS_Split)

# Recode factor levels
# rt_words_frq <- rt_words_frq |>
#   mutate(Base_Frequency = case_match(Base_Frequency, "Low" ~ "Low BF", "High" ~ "High BF"),
```

```r
#         Family_Size = case_match(Family_Size, "Small" ~ "Small Family", "Large" ~ "Large Family"))
# rt_nwords_frq <- rt_nwords_frq |> mutate(Base_Frequency = case_match(Base_Frequency, "Low" ~ "Low BF", "High" ~ "High BF"),
#                                          Family_Size = case_match(Family_Size, "Small" ~ "Small Family", "Large" ~ "Large Family"))
#
# rt_words_frq$Orthographic_Sensitivity[rt_words_frq$Orthographic_Sensitivity == "Low"] <- "Low Sensitivity"
# rt_words_frq$Orthographic_Sensitivity[rt_words_frq$Orthographic_Sensitivity == "High"] <- "High Sensitivity"
```

Word Data

Use `complete.cases()` to find which rows have missing data in the model-relevant variables:

```r
# Specify only the variables used in the model
model_vars_w <- c("response_time",  "Log10BF","BF", "FS","Family_Size", "Base_Frequency", "Orthographic_Sensitivity","SubjID")

# Identify incomplete rows cohort 1
incomplete_cases_words <- rt_words_frq[!complete.cases(rt_words_frq[, model_vars_w]), ]
rt_words_cmpl <- rt_words_frq[complete.cases(rt_words_frq[, model_vars_w]), ]
# View them
# print(incomplete_cases_words)

# Standardize the predictors
rt_words_cmpl$Log10BF_std <- as.numeric(scale(rt_words_cmpl$Log10BF, center = TRUE, scale = TRUE))
rt_words_cmpl$FS_std <- as.numeric(scale(rt_words_cmpl$FS, center = TRUE, scale = TRUE))
rt_words_cmpl$Log10WF_std <- as.numeric(scale(rt_words_cmpl$Log10WF, center = TRUE, scale = TRUE))
rt_words_cmpl$Log10FS_std <- as.numeric(scale(rt_words_cmpl$Log10FS, center = TRUE, scale = TRUE))
rt_words_cmpl$Dim.2_std <- as.numeric(scale(rt_words_cmpl$Dim.2, center = TRUE, scale = TRUE))
```

Anova

```r
anova_model_words <- mixed(
  response_time ~ Base_Frequency * Family_Size * Orthographic_Sensitivity +
    (1 | SubjID) +
    (1 | STRING),
  data = rt_words_cmpl,
  method = "S")
anova_model_words
```

```
|| Mixed Model Anova Table (Type 3 tests, S-method)
||
|| Model: response_time ~ Base_Frequency * Family_Size * Orthographic_Sensitivity +
|| Model:     (1 | SubjID) + (1 | STRING)
|| Data: rt_words_cmpl
||                                                 Effect         df        F p.value
|| 1                                       Base_Frequency   1, 92.45 10.29 **    .002
|| 2                                          Family_Size   1, 92.44  9.41 **    .003
|| 3                             Orthographic_Sensitivity   1, 64.87   3.83 +    .055
|| 4                           Base_Frequency:Family_Size   1, 92.45     1.08    .300
|| 5              Base_Frequency:Orthographic_Sensitivity 1, 5682.28     0.05    .817
|| 6                 Family_Size:Orthographic_Sensitivity 1, 5682.26     0.06    .809
|| 7 Base_Frequency:Family_Size:Orthographic_Sensitivity 1, 5682.09     0.16    .691
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

```r
m1 <- anova_model_words$full_model    # Extract the lmer model
ranova(m1) # formally test whether adding each random effect improves fit
```

```
|| ANOVA-like table for random-effects: Single term deletions
||
|| Model:
|| response_time ~ Base_Frequency + Family_Size + Orthographic_Sensitivity + (1 | SubjID) + (1 | STRING) + Base_Frequency:Family_Size + Base_Frequ
||              npar logLik   AIC     LRT Df Pr(>Chisq)
|| <none>         11 -35809 71639
|| (1 | SubjID)   10 -36718 73455 1817.77  1  < 2.2e-16 ***
|| (1 | STRING)   10 -35899 71817  179.65  1  < 2.2e-16 ***
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Extract effect sizes from your ANOVA model
eta_squared(anova_model_words, partial = TRUE)
```

```
|| # Effect Size for ANOVA (Type III)
||
|| Parameter                                           | Eta2 (partial) |       95% CI
|| ------------------------------------------------------------------------------------
|| Base_Frequency                                      |           0.10 | [0.02, 1.00]
|| Family_Size                                         |           0.09 | [0.02, 1.00]
|| Orthographic_Sensitivity                            |           0.06 | [0.00, 1.00]
|| Base_Frequency:Family_Size                          |           0.01 | [0.00, 1.00]
|| Base_Frequency:Orthographic_Sensitivity             |       9.41e-06 | [0.00, 1.00]
```

```
|| Family_Size:Orthographic_Sensitivity                 |       1.03e-05 | [0.00, 1.00]
|| Base_Frequency:Family_Size:Orthographic_Sensitivity |       2.78e-05 | [0.00, 1.00]
||
|| - One-sided CIs: upper bound fixed at [1.00].
```

```r
# Compute Marginal(fixed effects only) and Conditional(fixed + random effects) R²
r2(anova_model_words)
```

```
|| # R2 for Mixed Models
||
||   Conditional R2: 0.360
||      Marginal R2: 0.028
```

Concise

Models including random slopes for Base Frequency and Family Size by subject failed to converge or produced singular fits, indicating that the data did not support estimation of these additional variance components. Consequently, we report results from a simpler model with random intercepts for subjects and items (STRING), which converged cleanly and provided stable estimates.

Fuller explanation

We initially attempted to fit a maximal random-effects structure following Barr et al. (2013), including random slopes for Base Frequency and Family Size by subject. However, these models yielded singular fits (zero variance estimates and perfect correlations among random effects). Because such structures can produce unreliable standard errors and inflated Type I error rates, we adopted the maximal non-singular model, containing random intercepts for both subjects and items (STRING). All reported statistics are based on this model.

Brief

(A more complex model including by-subject random slopes failed to converge; results from the non-singular intercept-only model are reported.)

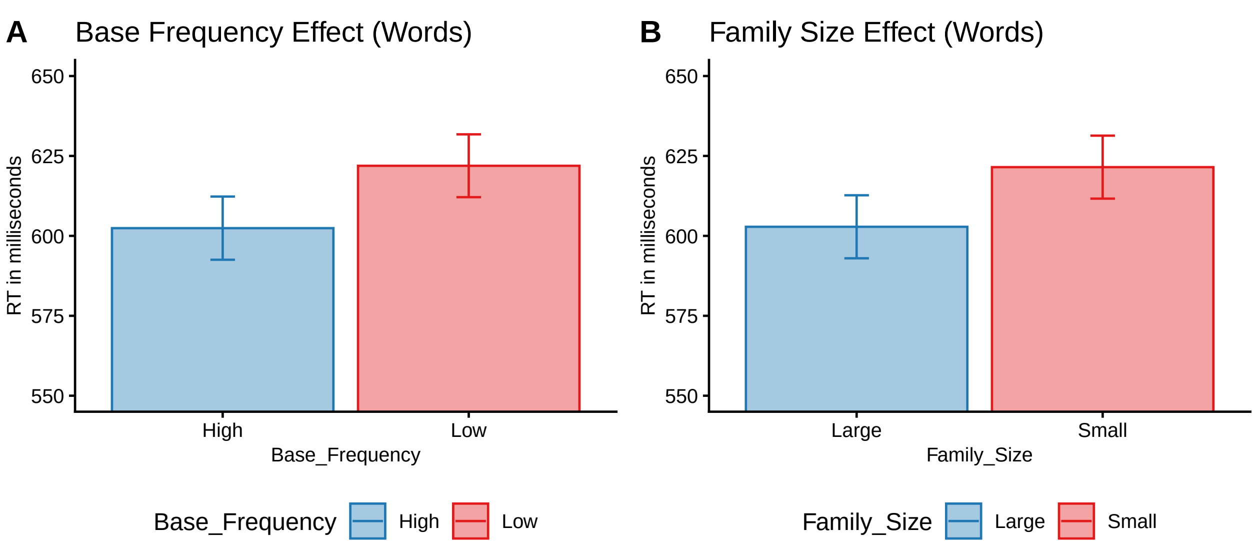
Main Findings

Effect	df	F	p.value
Base_Frequency	1, 92.45	10.29 **	.002
Family_Size	1, 92.44	9.41 **	.003

Plots

```
||  Base_Frequency   emmean       SE  df asymp.LCL asymp.UCL
||  High           602.4149 9.881996 Inf  583.0465  621.7832
||  Low            621.9280 9.826032 Inf  602.6694  641.1867
||
|| Results are averaged over the levels of: Family_Size, Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

```
||  Family_Size   emmean       SE  df asymp.LCL asymp.UCL
||  Large       602.8423 9.856568 Inf  583.5238  622.1608
||  Small       621.5006 9.851504 Inf  602.1920  640.8092
||
|| Results are averaged over the levels of: Base_Frequency, Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```



Non-word Data

Use `complete.cases()` to find which rows had missing data in the model-relevant variables:

Anova Family Size

```
anova_model_nwords_fs <- mixed(
  response_time ~ Complexity * Family_Size * Orthographic_Sensitivity +
    (1 | SubjID) +
    (1 | ItemID),
  data = rt_nwords_cmpl,
  method = "S")
anova_model_nwords_fs
```

```
|| Mixed Model Anova Table (Type 3 tests, S-method)
||
|| Model: response_time ~ Complexity * Family_Size * Orthographic_Sensitivity +
|| Model:     (1 | SubjID) + (1 | ItemID)
|| Data: rt_nwords_cmpl
||                                           Effect         df          F p.value
|| 1                                     Complexity 1, 4529.58 124.76 ***   <.001
|| 2                                    Family_Size   1, 95.23       1.10    .297
|| 3                       Orthographic_Sensitivity   1, 63.59     5.37 *    .024
|| 4                         Complexity:Family_Size 1, 4525.57       0.92    .338
|| 5            Complexity:Orthographic_Sensitivity 1, 4512.75       0.96    .327
|| 6           Family_Size:Orthographic_Sensitivity 1, 4448.54       0.09    .770
|| 7 Complexity:Family_Size:Orthographic_Sensitivity 1, 4508.97       0.06    .809
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

```
m2 <- anova_model_nwords_fs$full_model    # Extract the lmer model
ranova(m2) # Run random effects comparison
```

```
|| ANOVA-like table for random-effects: Single term deletions
||
|| Model:
|| response_time ~ Complexity + Family_Size + Orthographic_Sensitivity + (1 | SubjID) + (1 | ItemID) + Complexity:Family_Size + Complexity:Orthogr
||              npar logLik   AIC     LRT Df Pr(>Chisq)
|| <none>         11 -28033 56089
|| (1 | SubjID)   10 -28862 57743 1656.71  1  < 2.2e-16 ***
|| (1 | ItemID)   10 -28105 56230  142.95  1  < 2.2e-16 ***
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Main Findings

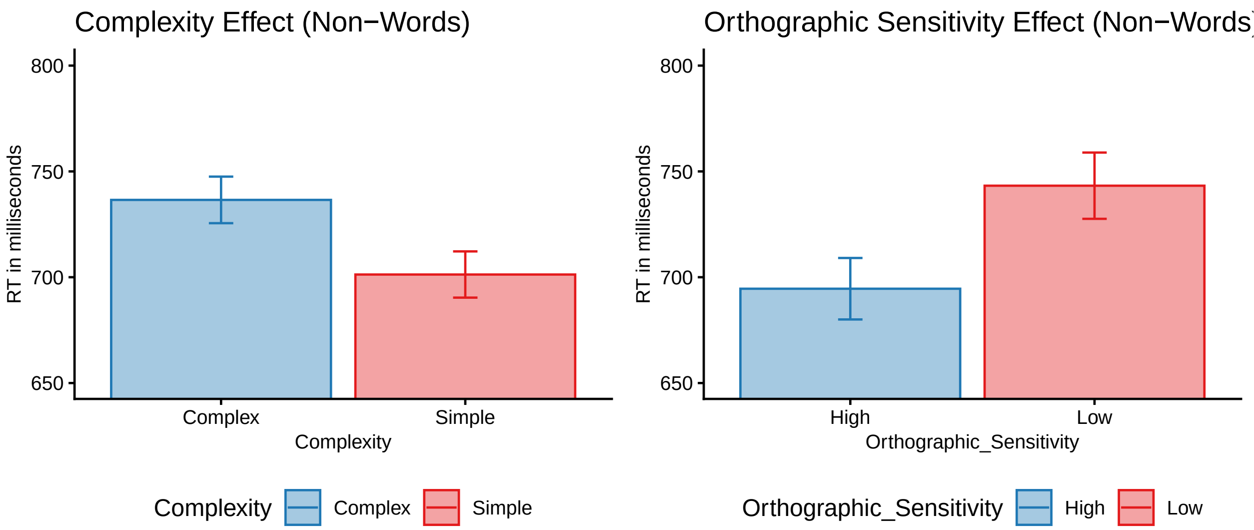
Effect	df	F	p.value
Complexity	1, 4529.58	124.76 ***	<.001
Orthographic_Sensitivity	1, 63.59	5.37 *	.024

Non-word complexity had a robust effect; complex non-words (e.g., pseudoderived forms) elicited longer response times than simple ones. Participants with higher **orthographic sensitivity** responded significantly faster overall, suggesting more efficient processing of letter patterns even in non-words. **Morphological family size** did not modulate non-word RTs, nor did it interact with complexity or orthographic sensitivity. Interpretation: In the absence of lexical representations, apparent "family size" (based on real-word analogues) does not measurably influence non-word recognition.

Plots

```
||  Complexity   emmean       SE  df asymp.LCL asymp.UCL
||  Complex    736.5355 11.00010 Inf  714.9757  758.0953
||  Simple     701.2803 10.92368 Inf  679.8703  722.6904
||
|| Results are averaged over the levels of: Family_Size, Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

```
||  Orthographic_Sensitivity   emmean       SE  df asymp.LCL asymp.UCL
||  High                     694.5669 14.52299 Inf  666.1023  723.0314
||  Low                      743.2490 15.65686 Inf  712.5621  773.9359
||
|| Results are averaged over the levels of: Complexity, Family_Size
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```


Complexity Effect (Non−Words)
800
750
700
650
RT in milliseconds
Complex
Simple
Complexity
Complexity
Complex
Simple
Orthographic Sensitivity Effect (Non−Words)
800
750
700
650
RT in milliseconds
High
Low
Orthographic_Sensitivity
Orthographic_Sensitivity
High
Low

Anova Base Frequency

```r
anova_model_nwords_bf <- mixed(
  response_time ~ Complexity * Base_Frequency * Orthographic_Sensitivity +
    (1 | SubjID) +
    (1 | ItemID),
  data = rt_nwords_cmpl,
  method = "S")
anova_model_nwords_bf
```

```
|| Mixed Model Anova Table (Type 3 tests, S-method)
||
|| Model: response_time ~ Complexity * Base_Frequency * Orthographic_Sensitivity +
|| Model:     (1 | SubjID) + (1 | ItemID)
|| Data: rt_nwords_cmpl
||                                              Effect         df          F p.value
|| 1                                        Complexity 1, 4534.35 127.54 ***   <.001
|| 2                                    Base_Frequency   1, 95.95  12.99 ***   <.001
|| 3                          Orthographic_Sensitivity   1, 63.60     5.32 *    .024
|| 4                         Complexity:Base_Frequency 1, 4535.47     4.26 *    .039
|| 5               Complexity:Orthographic_Sensitivity 1, 4517.84       0.86    .353
|| 6           Base_Frequency:Orthographic_Sensitivity 1, 4452.86       0.04    .838
|| 7 Complexity:Base_Frequency:Orthographic_Sensitivity 1, 4515.97       0.35    .555
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

```r
m3 <- anova_model_nwords_bf$full_model    # Extract the lmer model
ranova(m3) # Run random effects comparison
```

```
|| ANOVA-like table for random-effects: Single term deletions
||
|| Model:
|| response_time ~ Complexity + Base_Frequency + Orthographic_Sensitivity + (1 | SubjID) + (1 | ItemID) + Complexity:Base_Frequency + Complexity:O
||              npar logLik   AIC     LRT Df Pr(>Chisq)
|| <none>         11 -28026 56074
|| (1 | SubjID)   10 -28857 57734 1662.17  1  < 2.2e-16 ***
|| (1 | ItemID)   10 -28087 56195  122.89  1  < 2.2e-16 ***
|| ---
|| Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Main Findings

Effect	df	F	p.value
Complexity	1, 4534.35	127.54 ***	<.001
Base_Frequency	1, 177.77	12.99 **	<.001
Orthographic_Sensitivity	1, 63.60	5.32 *	.024
Complexity:Base_Frequency	1, 4535.47	4.26 *	.039

- Complexity ($F = 127.5$, $p < .001$): complex > simple non-words → slower responses.
- Base Frequency ($F = 13.0$, $p < .001$): non-words derived from high-frequency bases were processed faster than those from low-frequency bases — an echo of lexical familiarity effects even though the items are illegal.
- Orthographic Sensitivity ($F = 5.32$, $p = .024$): same direction as before.
- Complexity × Base Frequency ($F = 4.26$, $p = .039$): The effect of complexity was larger for high-frequency bases than for low-frequency ones.

Interaction Effects: `Complexity x Base_Frequency`

```r
# Estimated marginal means for the family_size × base frequency interaction
(emm1 <- emmeans(anova_model_nwords_bf, ~ Complexity * Base_Frequency))
```

Simple Contrasts

```
||  Complexity Base_Frequency emmean   SE  df asymp.LCL asymp.UCL
||  Complex    High              751 11.6 Inf       728       773
||  Simple     High              709 11.4 Inf       686       731
||  Complex    Low               723 11.5 Inf       700       745
||  Simple     Low               694 11.4 Inf       671       716
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

```r
# Get all pairswise contrasts
emm1_contrasts <- contrast(emm1,  method = "pairwise", by = NULL, adjust = "none")
emm1_contrasts
```

```
||  contrast                  estimate   SE  df z.ratio p.value
||  Complex High - Simple High     42.3 4.68 Inf   9.033  <.0001
||  Complex High - Complex Low     28.0 6.95 Inf   4.021  0.0001
||  Complex High - Simple Low      57.1 6.78 Inf   8.429  <.0001
||  Simple High - Complex Low     -14.3 6.69 Inf  -2.136  0.0326
||  Simple High - Simple Low       14.9 6.51 Inf   2.285  0.0223
||  Complex Low - Simple Low       29.2 4.27 Inf   6.834  <.0001
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
```

```
# Keep only the contrasts you want
# Simple effects of Complexity at each level of Base_Frequency
# Simple effects of Base_Frequency at each level of Complexity
keep <- c("Complex High - Simple High",
          "Complex Low - Simple Low",
          "Complex High - Complex Low",
          "Simple High - Simple Low")
(emm1_contrasts_filtered <- subset(emm1_contrasts, contrast %in% keep))
```

```
||  contrast                  estimate   SE  df z.ratio p.value
||  Complex High - Simple High     42.3 4.68 Inf   9.033  <.0001
||  Complex High - Complex Low     28.0 6.95 Inf   4.021  0.0001
||  Simple High - Simple Low       14.9 6.51 Inf   2.285  0.0223
||  Complex Low - Simple Low       29.2 4.27 Inf   6.834  <.0001
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
```

```
# Get Confidence Intervals
(emm1_contrasts_filtered_ci <- confint(emm1_contrasts_filtered))
```

```
||  contrast                  estimate   SE  df asymp.LCL asymp.UCL
||  Complex High - Simple High     42.3 4.68 Inf     33.09      51.4
||  Complex High - Complex Low     28.0 6.95 Inf     14.34      41.6
||  Simple High - Simple Low       14.9 6.51 Inf      2.12      27.6
||  Complex Low - Simple Low       29.2 4.27 Inf     20.81      37.5
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

```
# Get effect sizes
# Get all pairwise effect sizes
effs1 <- eff_size(emm1, sigma = sigma(m3), edf = df.residual(m3))
effs1
```

```
||  contrast                  effect.size     SE  df asymp.LCL asymp.UCL
||  Complex High - Simple High       0.414 0.0460 Inf    0.3235    0.5038
||  Complex High - Complex Low       0.274 0.0681 Inf    0.1402    0.4073
||  Complex High - Simple Low        0.559 0.0666 Inf    0.4288    0.6899
||  Simple High - Complex Low       -0.140 0.0655 Inf   -0.2682   -0.0115
||  Simple High - Simple Low         0.146 0.0638 Inf    0.0207    0.2707
||  Complex Low - Simple Low         0.286 0.0419 Inf    0.2035    0.3677
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| sigma used for effect sizes: 102.2
|| Degrees-of-freedom method: inherited from asymptotic when re-gridding
|| Confidence level used: 0.95
```

```
# Remove the two redundant rows (rows 3 and 4)
(effs1_filtered <- subset(effs1, !contrast %in% c("Complex High - Simple Low",
                                               "Simple High - Complex Low")))
```

```
||  contrast                  effect.size     SE  df asymp.LCL asymp.UCL
||  Complex High - Simple High       0.414 0.0460 Inf    0.3235     0.504
||  Complex High - Complex Low       0.274 0.0681 Inf    0.1402     0.407
||  Simple High - Simple Low         0.146 0.0638 Inf    0.0207     0.271
||  Complex Low - Simple Low         0.286 0.0419 Inf    0.2035     0.368
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| sigma used for effect sizes: 102.2
|| Degrees-of-freedom method: inherited from asymptotic when re-gridding
|| Confidence level used: 0.95
```

```

#  Interaction contrasts (difference-of-differences)
#    Compare base frequency effect in large vs small family)
contrast(emm1, interaction = "pairwise", by = NULL, adjust = "holm")
```

Interaction Contrasts

```
||  Complexity_pairwise Base_Frequency_pairwise estimate   SE  df z.ratio p.value
```

```
||  Complex - Simple    High - Low                  13.1 6.34 Inf   2.063  0.0391
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
```

```r
confint(contrast(emm1, interaction = c("pairwise", "pairwise")))
```

```
||  Complexity_pairwise Base_Frequency_pairwise estimate   SE  df asymp.LCL asymp.UCL
||  Complex - Simple    High - Low                  13.1 6.34 Inf     0.655      25.5
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

```r
# Get confidence intervals, for each base frequency effect for each family size and then for interaction effect
confint(contrast(emmeans(m3, ~ Complexity | Base_Frequency), "pairwise"))
```

```
|| Base_Frequency = High:
||  contrast         estimate   SE  df asymp.LCL asymp.UCL
||  Complex - Simple     42.3 4.68 Inf      33.1      51.4
||
|| Base_Frequency = Low:
||  contrast         estimate   SE  df asymp.LCL asymp.UCL
||  Complex - Simple     29.2 4.27 Inf      20.8      37.5
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

```r
confint(contrast(emmeans(m3, ~ Base_Frequency | Complexity), "pairwise"))
```

```
|| Complexity = Complex:
||  contrast   estimate   SE  df asymp.LCL asymp.UCL
||  High - Low     28.0 6.95 Inf     14.34      41.6
||
|| Complexity = Simple:
||  contrast   estimate   SE  df asymp.LCL asymp.UCL
||  High - Low     14.9 6.51 Inf      2.12      27.6
||
|| Results are averaged over the levels of: Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

Table 4. Estimated Marginal Means and Pairwise Contrasts for the Complexity × Base Frequency Interaction

Contrast	Delta RT (ms)	p	Effect size (d)	95% CI for Delta RT (ms)	Interpretation
Complex High (751) - Simple High (709)	+42.3	< .001	0.41	[33.1, 51.4]	Strong complexity cost at high base frequency
Complex Low (723) - Simple Low (694)	+29.2	< .001	0.29	[20.8, 37.5]	Moderate complexity cost at low base frequency
Complex High (751) - Complex Low (723)	+28.0	.0001	0.27	[14.3, 41.6]	Complex items slower when based on high-frequency stems
Simple High (709) - Simple Low (694)	+14.9	.022	0.15	[2.1, 27.6]	Small frequency benefit among simple items

Table 5. Summary of Main Effects and Interactions

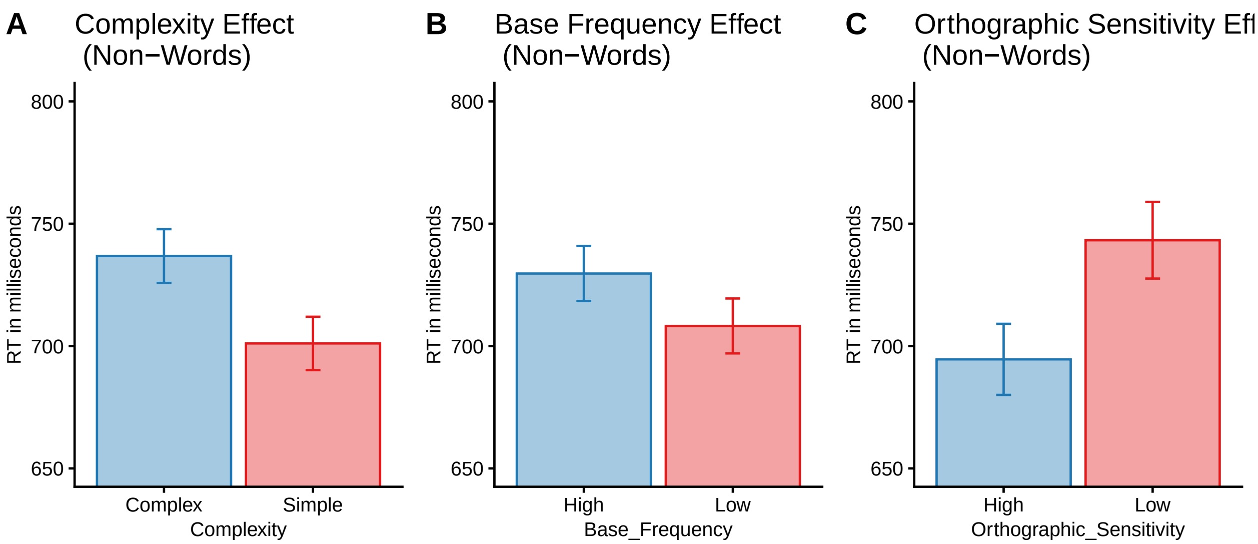
Effect	Direction / Pattern	Interpretation
Complexity	Complex > Simple across all conditions	Morphological structure slows non-word rejection (decomposition cost).
Base Frequency	High > Low (faster RTs)	Familiar letter sequences facilitate processing even for non-words.
Complexity × Base Frequency	Larger complexity cost for high-frequency bases (Delta ~ 13 ms)	Morphological activation stronger for familiar bases.
Family Size	n.s.	No measurable influence; non-words lack real morphological families.
Orthographic Sensitivity	High > Low (faster RTs overall)	Readers with higher orthographic sensitivity are generally more efficient.

Main Effects Plots

```
||  Complexity   emmean       SE  df asymp.LCL asymp.UCL
||  Complex    736.7913 10.98256 Inf  715.2659  758.3168
||  Simple     701.0784 10.90537 Inf  679.7042  722.4525
||
|| Results are averaged over the levels of: Base_Frequency, Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

```
||  Base_Frequency   emmean       SE  df asymp.LCL asymp.UCL
||  High           729.6473 11.23700 Inf  707.6232  751.6715
||  Low            708.2224 11.22244 Inf  686.2268  730.2180
||
|| Results are averaged over the levels of: Complexity, Orthographic_Sensitivity
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```

```
||  Orthographic_Sensitivity   emmean       SE  df asymp.LCL asymp.UCL
||  High                     694.6539 14.52304 Inf  666.1892  723.1185
||  Low                      743.2159 15.66167 Inf  712.5196  773.9122
||
|| Results are averaged over the levels of: Complexity, Base_Frequency
|| Degrees-of-freedom method: asymptotic
|| Confidence level used: 0.95
```



Interaction Plots

```r
p8 <- emmip(anova_model_nwords_bf, Complexity ~ Base_Frequency) + my_style
p9 <- emmip(anova_model_nwords_bf, Base_Frequency ~ Complexity) + my_style

plot_grid(p8, p9, ncol = 2)
```