rnnpsycholing Japanese NPI (sentences with embedding and embedded shika)

Takashi Morita

目次

Introduction

We are looking for a 2x2x2 interaction of:

- presence vs. absence of the Japanese NPI *shika* (しか) in the embedded clause.
- affirmativeness vs. negativeness of main verb
- affirmativeness vs. negativeness of embedded verb

for each of the three grammatical cases (NOM, ACC, DAT) of the *shika*-attached NP.

e.g.

- NOM
  - 佐藤は [社長-{しか, が} パーティ-に 友人-を {呼んだ, 呼ばなかった} と] {思った, 思わなかった}。
  - Sato-TOP [CEO-{*shika*, NOM} party-DAT friend-ACC {invited, didn't invite} that] {thought, didn't think}.
- ACC
  - 佐藤は [社長-が パーティ-に 友人-{しか, を} {呼んだ, 呼ばなかった} と] {思った, 思わなかった}。

  - Sato-TOP CEO-NOM party-DAT friend-{*shika*, ACC} {invited, didn't invite} that {thought, didn't think}.
- DAT
  - 佐藤は [社長-が パーティ-{に-しか, に} 友人-を {呼んだ, 呼ばなかった} と] {思った, 思わなかった}。
  - Sato-TOP [CEO-NOM party-{DAT-*shika*, DAT} friend-ACC {invited, didn't invite} that] {thought, didn't think}.

Why is this interesting?

1. A grammatical sentence with *shika* in the embedded clause must have a negative embedded verb.
   - A significant increase in surprisal of the affirmative embedded verbs must be predicted by the LSTM conditioned on the presence of *shika* if the learning is successful.
2. Negation of the embedded verb does not satisfy the *shika*'s grammatical condition.
   - No significant increase in surprisal of the affirmative main verbs given *shika* is expected for a successful learner.
   - Nor significant interaction between the main and embedded verbs given *shika* is expected for a successful learner.

Load data

```
rm(list = ls())
library(tidyverse)
library(brms)
library(lme4)
library(lmerTest)
library(plotrix)

REGIONS = c('main_prefix', 'embedded_prefix', 'embedded_V', 'complementizer', 'main_V', 'end')

token_based_data_path = 'jp_shika_test_sentences_embedded_shika-embedded_surprisal-per-token.tsv'
data_token_based = read_tsv(token_based_data_path)
```

```
## Parsed with column specification:
## cols(
##   sent_index = col_integer(),
##   token_index = col_integer(),
##   token = col_character(),
##   region = col_character(),
##   log_prob = col_double(),
##   shika_case = col_character(),
##   shika = col_character(),
##   embed_V = col_character(),
##   main_V = col_character(),
```

```
##   surprisal = col_double(),
##   LSTM = col_character()
## )
```

```
# Fill the initial surprisal by 0.
data_token_based[is.na(data_token_based$surprisal),]$surprisal = 0
data_token_based$region = factor(data_token_based$region, levels=REGIONS)

data_region_based = data_token_based %>%
    group_by(sent_index, region, shika, embed_V, main_V, shika_case) %>%
        summarise(surprisal=sum(surprisal)) %>%
        ungroup() %>%
    mutate(
        shika=factor(shika, levels=c("shika", "no-shika")),
        embed_V=factor(embed_V, levels=c("affirmative", "negative")),
        main_V=factor(main_V, levels=c("affirmative", "negative")),
        shika_case=factor(shika_case, levels=c("NOM", "ACC", "DAT"))
        )

# Sum coding of the variables.
contrasts(data_region_based$shika) = "contr.sum"
contrasts(data_region_based$embed_V) = "contr.sum"
contrasts(data_region_based$main_V) = "contr.sum"

# Make sure that the dataframe is sorted appropriately.
# First by embed_V (affirmative vs. negative)
data_region_based = data_region_based[order(data_region_based$embed_V),]
# Then by main_V
data_region_based = data_region_based[order(data_region_based$main_V),]
# finally by sent_index
data_region_based = data_region_based[order(data_region_based$sent_index),]
```

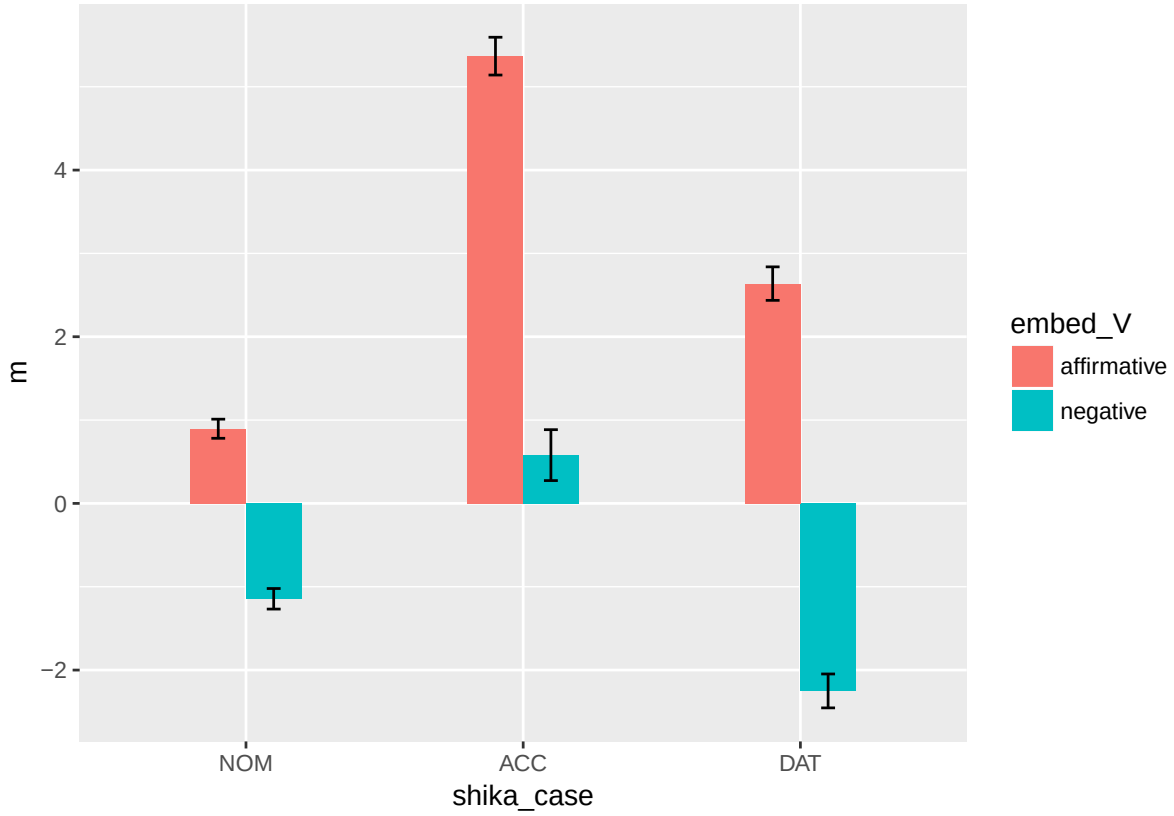
Embedded verb region

Visualization

```
# Focus on the V (verb) region.
data_V = subset(data_region_based, region == 'embedded_V')

# Get difference in surprisal between shika vs. no-shika.
data_V_shika = subset(data_V, shika == 'shika')
data_V_no_shika = subset(data_V, shika == 'no-shika')
data_V_shika$surprisal_diff = data_V_shika$surprisal - data_V_no_shika$surprisal
```

```r
# Visualize the difference in surprisal increase/dicrease between affirmative vs. negative verbs.
data_V_shika %>%
    group_by(embed_V, shika_case) %>%
    summarise(m=mean(surprisal_diff),
              s=std.error(surprisal_diff),
              upper=m + 1.96*s,
              lower=m - 1.96*s) %>%
    ungroup() %>%
    ggplot(aes(x=shika_case, y=m, ymin=lower, ymax=upper, width=0.4, fill=embed_V)) +
        geom_bar(stat = 'identity', position = "dodge") +
        geom_errorbar(position=position_dodge(0.4), width=.1)
```



- TOP
  - Small increase in surprisal of the affirmative verbs.
  - Visible decrease in surprisal of the negative verbs.
- ACC
  - Greatest increase in surprisal of the affirmative verbs.
  - Small increase in surprisal of the negative verbs.
- DAT
  - Visible increase in surprisal of the affirmative verbs.
  - Greatest decrease in surprisal of the negative verbs.

Regressions

NOM

```
sub_data = subset(data_V_shika, shika_case == 'NOM')

m = lmer(
        surprisal_diff
            ~ embed_V
                + (1 | sent_index)
        ,
        data=sub_data
        )
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: surprisal_diff ~ embed_V + (1 | sent_index)
##    Data: sub_data
##
## REML criterion at convergence: 10613.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.48522 -0.65671  0.05128  0.59891  2.40147
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  sent_index (Intercept) 3.165    1.779
##  Residual               1.799    1.341
## Number of obs: 2688, groups:  sent_index, 672
##
## Fixed effects:
##               Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)   -0.12440    0.07334  671.00000  -1.696   0.0903 .
## embed_V1       1.02042    0.02587 2015.00000  39.449   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr)
## embed_V1 0.000
```

- Statistically significant effect of the affirmativeness vs. negativeness of embedded verbs.

ACC

```
sub_data = subset(data_V_shika, shika_case == 'ACC')

m = lmer(
        surprisal_diff
            ~ embed_V
                + (1 | sent_index)
        ,
        data=sub_data
        )
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: surprisal_diff ~ embed_V + (1 | sent_index)
##    Data: sub_data
##
## REML criterion at convergence: 4315.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.18574 -0.39939 -0.05289  0.38984  1.79707
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  sent_index (Intercept) 8.500    2.915
##  Residual               1.534    1.239
## Number of obs: 1064, groups:  sent_index, 266
##
## Fixed effects:
##              Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)   2.97326    0.18274 264.99999   16.27   <2e-16 ***
## embed_V1      2.39404    0.03797 797.00000   63.04   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr)
## embed_V1 0.000
```

- Statistically significant effect of the affirmativeness vs. negativeness of embedded verbs.

- Greater effect than NOM.

DAT

```
sub_data = subset(data_V_shika, shika_case == 'DAT')

m = lmer(
        surprisal_diff
            ~ embed_V
                + (1 | sent_index)
        ,
        data=sub_data
        )
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: surprisal_diff ~ embed_V + (1 | sent_index)
##    Data: sub_data
##
## REML criterion at convergence: 3170.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.62747 -0.69928  0.00561  0.57277  1.70888
##
## Random effects:
##  Groups     Name        Variance Std.Dev.
##  sent_index (Intercept) 3.7801   1.9442
##  Residual               0.9997   0.9999
## Number of obs: 896, groups:  sent_index, 224
##
## Fixed effects:
##             Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)   0.1935     0.1341 223.0000   1.443    0.151
## embed_V1      2.4440     0.0334 671.0000  73.168   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr)
## embed_V1 0.000
```

- Statistically significant effect of the affirmativeness vs. negativeness of embedded verbs.

- Greater effect than NOM, similar to ACC.

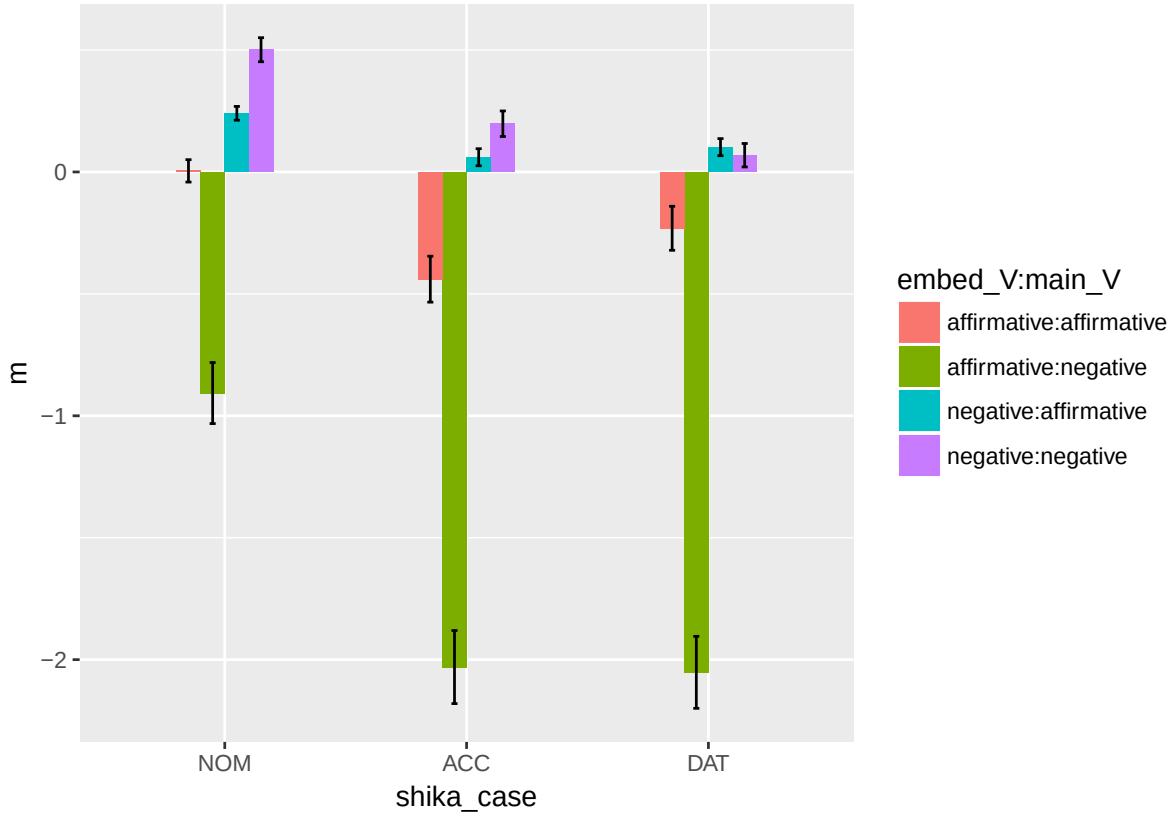
Main verb region

Visualization

```
# Focus on the V (verb) region.
data_V = subset(data_region_based, region == 'main_V')

# Get difference in surprisal between shika vs. no-shika.
data_V_shika = subset(data_V, shika == 'shika')
data_V_no_shika = subset(data_V, shika == 'no-shika')
data_V_shika$surprisal_diff = data_V_shika$surprisal - data_V_no_shika$surprisal


# Visualize the difference in surprisal increase/dicrease between affirmative vs. negative verbs.
data_V_shika %>%
    group_by(embed_V, main_V, shika_case) %>%
    summarise(m=mean(surprisal_diff),
            s=std.error(surprisal_diff),
            upper=m + 1.96*s,
            lower=m - 1.96*s) %>%
    ungroup() %>%
    ggplot(aes(x=shika_case, y=m, ymin=lower, ymax=upper, width=0.4, fill=embed_V:main_V)) +
        geom_bar(stat = 'identity', position = "dodge") +
        geom_errorbar(position=position_dodge(0.4), width=.1)
```



- Embedded verbs determine the increase vs. decrease in surprisal at the main verb region.
  - Affirmative embedded verbs decrease the surprisal.
  - Negative embedded verbs cause small increase in surprisal.

Regressions

NOM

```
sub_data = subset(data_V_shika, shika_case == 'NOM')

m = lmer(
        surprisal_diff
            ~ embed_V * main_V
                + (embed_V + main_V | sent_index)
        ,
        data=sub_data
        )
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## surprisal_diff ~ embed_V * main_V + (embed_V + main_V | sent_index)
```

```
##    Data: sub_data
##
## REML criterion at convergence: 6331.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.1700 -0.4939 -0.0063  0.5641  3.3761
##
## Random effects:
##  Groups     Name        Variance Std.Dev. Corr
##  sent_index (Intercept) 0.29976  0.5475
##             embed_V1    0.16708  0.4087    0.94
##             main_V1     0.08839  0.2973   -0.99 -0.97
##  Residual               0.36438  0.6036
## Number of obs: 2688, groups:  sent_index, 672
##
## Fixed effects:
##                    Estimate Std. Error         df t value Pr(>|t|)
## (Intercept)        -0.04018    0.02412  676.45694  -1.666   0.0962 .
## embed_V1           -0.41100    0.01960  676.30414 -20.969   <2e-16 ***
## main_V1             0.16259    0.01634  768.60442   9.949   <2e-16 ***
## embed_V1:main_V1    0.29295    0.01164 1342.00028  25.161   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) emb_V1 man_V1
## embed_V1     0.661
## main_V1     -0.611 -0.547
## embd_V1:_V1  0.000  0.000  0.000
```

ACC

```
sub_data = subset(data_V_shika, shika_case == 'ACC')

m = lmer(
        surprisal_diff
            ~ embed_V * main_V
                + (embed_V + main_V | sent_index)
        ,
        data=sub_data
        )
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## surprisal_diff ~ embed_V * main_V + (embed_V + main_V | sent_index)
##    Data: sub_data
##
## REML criterion at convergence: 2117.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.4306 -0.5660 -0.0202  0.5284  3.9856
##
## Random effects:
##  Groups     Name        Variance Std.Dev. Corr
##  sent_index (Intercept) 0.2081   0.4561
##             embed_V1    0.1107   0.3327    1.00
##             main_V1     0.0295   0.1718   -0.94 -0.97
##  Residual               0.2627   0.5125
## Number of obs: 1064, groups:  sent_index, 266
##
## Fixed effects:
##                   Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)       -0.55303    0.03208 267.69883  -17.24   <2e-16 ***
## embed_V1          -0.68207    0.02575 280.95105  -26.49   <2e-16 ***
## main_V1            0.36332    0.01892 304.15340   19.21   <2e-16 ***
## embed_V1:main_V1   0.43203    0.01571 529.99946   27.50   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) emb_V1 man_V1
## embed_V1     0.688
## main_V1     -0.457 -0.427
## embd_V1:_V1  0.000  0.000  0.000
```

DAT

```
sub_data = subset(data_V_shika, shika_case == 'DAT')

m = lmer(
        surprisal_diff
            ~ embed_V * main_V
                + (embed_V + main_V | sent_index)
```

```
        ,
        data=sub_data
        )
summary(m)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula:
## surprisal_diff ~ embed_V * main_V + (embed_V + main_V | sent_index)
##    Data: sub_data
##
## REML criterion at convergence: 1589.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.3921 -0.5444  0.0072  0.5199  4.2229
##
## Random effects:
##  Groups     Name        Variance Std.Dev. Corr
##  sent_index (Intercept) 0.16440  0.4055
##             embed_V1    0.08889  0.2982    0.99
##             main_V1     0.03171  0.1781   -0.86 -0.79
##  Residual               0.20295  0.4505
## Number of obs: 896, groups:  sent_index, 224
##
## Fixed effects:
##                   Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)       -0.52822    0.03099 226.47153  -17.04   <2e-16 ***
## embed_V1          -0.61333    0.02497 235.41760  -24.57   <2e-16 ***
## main_V1            0.46354    0.01919 233.98022   24.16   <2e-16 ***
## embed_V1:main_V1   0.44701    0.01505 446.00013   29.70   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) emb_V1 man_V1
## embed_V1     0.692
## main_V1     -0.467 -0.392
## embd_V1:_V1  0.000  0.000  0.000
```