

بخش اول

مقدمه

نمایش انتزاعی معنای فارسی یا نامفا بر پایه نمایش انتزاعی معنای انگلیسی شکل گرفته و برای زبان فارسی تعمیم و تغییر داده شده است.

نامفا روابط معنایی رویدادهای جمله را استخراج می‌کند. به عبارت دیگر، نمایش انتزاعی معنا در جمله تعیین کننده عاملی است که عملی را انجام می‌دهد و آنچه که از انجام این عمل تاثیر می‌پذیرد.

نامفا، هر جمله را در قالب یک گراف ریشه‌دار، جهت‌دار و بدون دور نمایش می‌دهد که در این گراف، یال‌ها دربردارنده روابط و برگ‌ها نشان‌دهنده مفاهیم هستند.

مانند درخت‌های تجزیه نحوی، نامفا ساختاری قابل پیمایش است که در آن تمامی کلمات جمله در نظر گرفته شده‌اند. از طرفی، نامفا مجموعه‌ای منقطع از لایه‌های برچسب‌خورده نیست. بر خلاف درخت تجزیه نحوی، این نوع از نمایش کاملاً انتزاعی است که می‌تواند هر تعداد از جملات زبان طبیعی را نمایش دهد. همچنین برخلاف تجزیه وابستگی، تک تک کلمات جمله در نامفا برچسب‌گذاری نمی‌شوند. نمایش انتزاعی معنا هیچ برنامه‌ای را در مورد چگونگی پردازش و استفاده از آن ارائه نمی‌دهد.

یک مثال



این نامفا نشان می‌دهد که دو رویداد با مفاهیم مخواستن و خندیدن وجود دارد. در رویداد خواستن arg0: (کسی که می‌خواهد) دلقک است و arg1: (چیزی که خواسته شده) رویداد خندیدن است. در رویداد خندیدن arg0: (کسی که می‌خندد) ماری است و arg1: (کسی که به او می‌خندند) دلقک است. در واقع، در این جمله دلقک با دو رویداد در ارتباط است: یکی arg0: رویداد خواستن و دیگری arg1: رویداد خندیدن. نامفا با وصل کردن دو یال جهت‌دار به یک گره این موضوع را نمایش می‌دهد.

گراف بالا را می‌توان با شیوه نمایش PENMAN به صورت زیر نمایش داد. این شیوه نمایش از طرفی خوانش ساده‌ای برای انسان دارد و از طرف دیگر برای ماشین قابل پردازش است.

خواستن / x

(دلک / x2) :ARG0

خندیدن / x3) :ARG1

(ماری / x4) :ARG0

((x2) :ARG1

متغیرهای x, x2, x3 و x4 گره‌های داخلی گراف بالا را نشان می‌دهند. توجه داشته باشید که x2 دو بار در این نمایش حضور دارد.

انتزاع: دوری از دستور فارسی

از نامفای بالا می‌توان جملات فارسی زیر را تولید کرد:

دلک می‌خواهد که ماری به او بخندد.

خواسته دلک خنده ماری به اوست.

خنده ماری به دلک خواسته دلک است.

خواسته دلک خنده ماری به او می‌باشد.

دلک این را می‌خواهد، خنده ماری به او.

رویدادهای خواستن و خندیدن می‌توانند به صورت فعلی، اسمی و یا صفتی ایجاد شوند. برچسب برگ‌ها در نامفا، دربردارنده مفاهیم هستند و کلمات با آن‌ها نشان داده نمی‌شوند. به جای اشاره به عنصری در نامفا به عنوان اسم یا فعل و سایر نقش‌های دستوری، از آنها به عنوان رویداد یا شیء یاد می‌کنیم.

موجودیتی که «دلک» نام دارد همزمان دو نقش را ایفا می‌کند. در نامفا ضمائر، ضمائر تهی، شناسه‌ها و شناسه‌های تهی تعریف خاصی ندارند و تنها به عنوان ابزارهای عادی زبان برای نمایش چندین نقش یک موجودیت خاص شناخته می‌شوند.

در بسیاری از موارد کلمات دستوری در نامفا نمایش داده نمی‌شوند:

تعمیر کردن / x

:ARG0 (x2 / ماری)

:ARG1 (x3 / ساعت)

ماری ساعت را تعمیر کرد.
ساعت به دست ماری تعمیر شد.
ساعت توسط ماری تعمیر شد.

تغییر دادن / x

:ARG0 (x2 / دلچک)

:ARG1 (x3 / تنظیمات)

دلچک تنظیمات را تغییر داد.

رخ دادن / x

:ARG1 (x2 / قتل عام)

:time (x2 / دیروز)

قتل عام دیروز رخ داد.
قتل عام دیروز به وقوع پیوست.

و / x

:op1 (x2 / دلچک)

:op2 (x3 / ماری)

ماری و دلچک
هردوی ماری و دلچک
ماری و دلچک، هردو

کلاه / x

((دلک / x2):poss

کلاه دلک

کلاه خود دلک

منطق محور تر از دستور زبان

در زبان انگلیسی دو جمله the boy may not go و the boy must not go از نظر نحوی ساختار مشابهی دارند، اما جایگاه قرار گیری رابطه polarity: در گراف هر جمله متفاوت است. با این حال، بر خلاف زبان انگلیسی، در زبان فارسی نفی بر روی مفهومی اعمال می شود که در نامفا نیز polarity: باید اعمال گردد. برای مثال «دلک نباید برود» همان معنای «دلک باید نرود» را می دهد. در حقیقت جمله دوم بسیار کم بسامد و نشاندار است. برای مشاهده مثال های انگلیسی دو جمله بالا به شیوه نامه انگلیسی رجوع کنید.

کانون

در یک جمله نامفانویسی شده، ریشه نامفا علاوه بر اینکه نقطه شروعی برای پیمایش گراف است، کانون کلی جمله را نیز دربرمی گیرد:

سفید / x

((مروارید / x2):domain

مروارید سفید است.

سفیدی مروارید.

مروارید / x

((سفید / x2):mod

مروارید سفید

مرواریدی که سفید است.

دیدن / x)

:ARG0 (x2 / دلک)

:ARG1 (x3 / سفید

:domain (x4 / مروارید)))

دلک می بیند که مروارید سفید است.

دلک سفیدی مروارید را می بیند.

دیدن / x)

:ARG0 (x2 / دلک)

:ARG1 (x3 / مروارید

:mod (x4 / سفید)))

دلک مروارید سفید را می بیند.

دلک مرواریدی که سفید است را می بیند.

می توان برای رابطه domain: از رابطه معکوس domain-of: استفاده کرد. اما برای اختصار از mod: استفاده می کنیم. روابط معکوس برای حفظ ساختار تک ریشه کاربرد فراوان دارند. برای مثال:

دیدن / x)

:ARG0 (x2 / دلک)

:ARG1 (x3 / دختر

:ARG0-of (x4 / خندیدن

:ARG1 x2)))

دلک دختری که به او می خندید را دید.

دختری که به دلک می خندید توسط دلک دیده شد.

در این نامفا رابطه arg0-of: مفهوم «دختر» را به طور طبیعی به رویداد «خندیدن» متصل می کند. برای تغییر کانون در نامفا، گره مورد نظر را در ریشه قرار می دهیم و باقی گره ها به دنبال آن می آیند.

خندیدن (x /
 :ARG0 (x2 / دختر
 :ARG1-of (x3 / دیدن
 :ARG0 (x4 / دلچسپ)))
 :ARG1 x4)

دختری که دلچسپ دید به او می‌خندید.

این مقوله را کانوی‌سازی می‌نامیم. مفهوم اصلی در مثال اول دیدن است، در حالی که در مثال دوم خندیدن مفهوم اصلی جمله می‌باشد. نامفای زیر نمونه‌ای دیگر از روابط معکوس را نشان می‌دهد.

تغییرکردن (x /
 :ARG1 (x2 / اسناد
 :instrument-of (x3 / بازداشتن)))

اسناد بازدارنده تغییر کردند.

شعارهای نامفا

محدودیت‌های نامفا

مفاهيم و روابط

پدیده‌ها

وجهیت

بر خلاف انگلیسی افعال وجهی کمکی در فارسی افعال واژگانی هستند. در نتیجه فعلی که مفهوم وجهی دارد به صورت مستقیم در گراف نامفا قرار می‌گیرد. ای رویه در انگلیسی برای افعال واژگانی که هدارا معنا وجهی می‌باشند قابل مشاهده است. علاوه بر این، «شاید» در نامفا به با رابطه قید به فعل اصلی متصل می‌شود. مثال‌های زیر چگونگی نمایش جملات وجهی را نشان می‌دهند. برای اطلاع از چگونگی نمایش وجهیت در انگلیسی به شیوه‌نامه انگلیسی رجوع کنید.

توانستن / x

:ARG0 (x2 / دلچک)

:ARG1 (x3 / نوشتن

:ARG0 x2))

دلچک می‌تواند بنویسد.

توانستن / x

:ARG0 (x2 / دلچک)

:ARG1 (x3 / رفتن

:ARG0 x2))

دلچک می‌تواند برود.

بایستن / x

:ARG0 (x2 / دلچک)

:ARG1 (x3 / رفتن

:ARG0 x2))

دلچک باید برود.

باریدن / x)
:ARG0 (x2 / باران
:mod (x3 / شاید))

شاید باران ببارد.

توجه داشته باشید که در نمایش افعال کمکی وجهی آرگوان‌ها تنها ذیل فعل اصلی می‌آیند.

نفی

در نمایش انتزاعی معنا، منفی بودن گزاره را به طور منطقی با polarity: نمایش می‌دهیم.

(x / رفتن :polarity -
:ARG0 (x2 / دلچک))

دلچک نمی‌رود.

(x / توانستن :polarity -
:ARG0 (x2 / دلچک)
:ARG1 (x3 / رفتن
:ARG0 x2))

دلچک نمی‌تواند برود.

فکرکردن / x

:ARG0 (x2 / دلچک)

:ARG1 (x3 / برگشتن :polarity -

:ARG0 (x4 / ماری)))

دلچک فکر می‌کند ماری برنمی‌گردد.

در برخورد با صفات منفی نیز از polarity: استفاده می‌کنیم:

به‌نظررسیدن / x

:ARG0 (x2 / اتاق)

:ARG1 (x3 / مرتب :polarity -))

اتاق نامرتب به‌نظر می‌رسید.

جملات پرسشی

برای نشان‌دادن پرسش با واژه‌های پرسشی در نمایش انتزاعی معنا از مفهوم amr-unknown استفاده می‌کنیم. در واقع جمله پرسشی به صورت خبری نامفاهم می‌شود و مفهومی که مورد پرسش است با مفهوم amr-unknown و با رابطه مناسب وارد گراف می‌شود:

پیدا کردن (x /

:ARG0 (x2 / ماری)

:ARG1 (a / amr-unknown))

ماری چه چیزی را پیدا کرد؟

پیدا کردن (x /

:ARG0 (x2 / ماری)

:ARG1 (x3 / دلک)

:location (a / amr-unknown))

ماری دلک را کجا پیدا کرد؟

پیدا کردن (x /

:ARG0 (x2 / ماری)

:ARG1 (x3 / دلک)

:manner (a / amr-unknown))

ماری چگونه دلک را پیدا کرد؟

پیدا کردن (x /

:ARG0 (x2 / ماری)

:ARG1 (x3 / کلاه)

:poss (a / amr-unknown)))

ماری کلاه چه کسی را پیدا کرد؟

دویدن / x
:ARG0 (x2 / ماری)
:manner (x3 / تند
:degree (a / amr-unknown)))

ماری چقدر تند می دوید؟

دیدن / x
:ARG0 (x2 / ماری)
:ARG1 (a / amr-unknown
:mod(x3 / بنفشی)))

ماری چه شیء بنفشی را دید؟

بررسی کردن / x
:ARG0 (x2 / ماری)
:ARG1 (a / amr-unknown
:domain (x3 / پرونده)))

ماری کدام پرونده را بررسی کرد؟

در نوع دوم جملات پرسشی در زبان فارسی که با بله یا خیر پاسخ داده می شوند نیز از مفهوم amr-unknown استفاده می کنیم. در چنین جملاتی از رابطه polarity: استفاده می شود.

پیدا کردن / x
:ARG0 (x2 / ماری)
:ARG1 (x3 / دلچک)
:polarity (a / amr-unknown))

آیا ماری دلچک را پیدا کرد؟

```

پیداشدن / x
  :ARG1 (x2 / دلک)
  :polarity (a / amr-unknown))

```

آیا دلک پیدا شد؟

در صورت استفاده از ادوات استفهام در جملاتی غیر از جملات پرسشی، به جای amr-unknown از نقش‌های معکوس استفاده می‌کنیم:

```

حدس زدن / x
  :ARG0 (x2 / من)
  :ARG1 (x3 / کس
    :ARG1-of (x4 / دیدن
      :ARG0 (x5 / تو))))

```

حدس می‌زنم چه کسی را دیدی.
کسی که دیدی را حدس می‌زنم.

از طرف دیگر، نقش‌های معکوس برای نمایش جملات پرسشی که به صورت مفهومی در جمله جاسازی شده‌اند نیز استفاده می‌شود. در این صورت، از مفهوم truth-value برای نمایش دادن این که آیا اتفاقی افتاده یا خیر استفاده می‌کنیم:

```

- polarity : دانستن / x
  :ARG0 (x2 / دلک)
  :ARG1 (t / truth-value
    :polarity-of (x3 / برگشتن
      :ARG0 (x4 / ماری))))

```

دلک نمی‌داند که ماری برگشته یا نه.
دلک نمی‌داند آیا ماری برگشته.
دلک نمی‌داند ماری برگشته یا نه.

نمایش انتزاعی معنا در جمله زیر متفاوت از مثال قبل است:

- polarity: دانستن / x
:ARG0 (x2 / دلچک)
:ARG1 (x3 / برگشتن
:ARG1 (x4 / ماری)))

دلچک نمی‌داند که ماری برگشته.
دلچک از برگشتن ماری مطلع نیست.

در پرسش‌های چندگزینه‌ای که انتخابی را پیش روی مخاطب قرار می‌دهند از مفهوم amr-choice استفاده می‌کنیم. این مفهوم در جایگاه «یا» قرار می‌گیرد و می‌تواند به تعداد نامحدودی آرگومان op را در ذیل خود جای دهد:

خواستن / x
:ARG0 (x2 / شما)
:ARG1 (a / amr-choice
:op1 (x3 / چای)
:op2 (x4 / قهوه)))

چای می‌خواهید یا قهوه؟

گفتن / x
:ARG0 (x2 / تو)
:ARG1 (a / amr-choice
:op1 (x3 / رفتن
:ARG0 (x4 / من))
:op2 (x5 / ماندن
:ARG0 (x6 / من))))

می‌گویی بروم یا بمانم؟

حالت امری و بیانی و گفتار رسمی

برای جملات امری از mode imperative استفاده می‌کنیم. جملات امری تعجبی نیز در این گروه قرار می‌گیرند.

```
(x / رفتن :mode imperative  
  :ARG0 (x2 / تو))
```

برو.
برو!

```
(x / رفتن :mode imperative  
  :ARG0 (x2 / ما))
```

برویم.
برویم!

کلماتی که بیانگر احساسات هستند و به‌طور مشخص به رویداد، شخص و یا چیز خاصی اشاره نمی‌کنند مانند: آه، آها، وای با mode expressive نمایش داده می‌شوند. اما دربرخورد با کلماتی که بیانگر تاکید هستند، علامت تعجب، و یا آواهایی همچون «اوووو» و «آآآآ» از این نقش استفاده نمی‌شود. بلکه چنین علائم و کلماتی را حذف می‌کنیم.

```
(x / آها :mode expressive)
```

آها!

در مواردی که جمله حاوی مفاهمی است که نشانگر حالت رسمی و احترام می‌باشد از رابطه polite استفاده می‌کنیم.

```
(x / فوت کردن :polite +)
```

فوت نمودن.

(x / بستن :mode imperative :polite +
:ARG0 (x2 / تو)
:ARG1 (x2 / پنجره))

لطفاً پنجره را ببندید.

حروف تعریف، شخص، شمار، ...

در نسخه کنونی نمایش انتزاعی معنا زمان (دستوری) (به غیر از رابطه مستقیم :time)، شناس و یا ناشناس بودن، نشانه‌های جمع، و علامت نقل قول نمایش داده نمی‌شود.

(x / رفتن
:ARG0 (x2 / دلچک))

دلچک رفت.
دلچک‌ها رفتند.
دلچکی رفت.
دلچک می‌رود.
دلچک‌ها می‌روند.

در نمایش انتزاعی معنا ضمائر اشاره نشان داده می‌شوند:

(x / دلچک
:mod (x2 / آن))

آن دلچک
آن دلچک‌ها

دلفک / x

$((\text{این} / x2) : \text{mod})$

این دلفک

این دلفک‌ها

چنانچه ضمائر اشاره در جمله مرجعی نداشته باشند، همچنان نمایش داده می‌شوند:

فاجعه / x

$((\text{آن} / x2) : \text{domain})$

آن فاجعه‌ای بود.

برخی از کلمات با خط تیره به دو جزء تقسیم می‌شوند. چنین کلماتی را در نمایش انتزاعی معنا به صورت زیر نشان می‌دهیم:

ساخت / x

$((\text{ویژه} / x2) : \text{mod})$

ساخت-ویژه

نقش‌های تلویحی

در نمایش انتزاعی معنا، نقش‌هایی وجود دارند که در جمله به صورت ضمنی به آن‌ها اشاره شده است. به عنوان مثال:

```

شدن / x
  ARG1 (x2 / او)
  ARG2 (x3 / متهم
    ARG2 (x4 / و
      op1 (x5 / دزدی کردن
        ARG1 x2)
      op2 (x6 / مقاومت کردن
        ARG0 x2
        ARG1 (x7 / بازداشتن
          ARG1 x2))))))

```

او به دزدی و مقاومت در برابر بازداشت متهم شد.

در این جمله، متغیر x_2 چهار بار تکرار شده است چرا که روشن است x_2 در برابر بازداشت شدن خودش مقاومت می‌کند و نه شخص دیگری. اگرچه، باید توجه داشت نقش «متهم کننده» و «بازداشت کننده» در نمایش انتزاعی معنا نشان داده نشده است، چرا که ممکن است این رویدادها از سوی دو شخص متفاوت رخ داده باشند.

افعال اسنادی

برای ساخت نامفا از جملات حاوی ساخت‌های اسنادی سه حالت زیر را در نظر می‌گیریم.

ساخت اسنادی ساده

در این ساخت‌ها فعل اسنادی «است» به کار رفته. در ساختارهایی که صفتی به اسمی نسبت داده شده است، اسم با رابطه domain: به صفت متصل می‌شود:

```

سرد / x
  domain (x2 / غذا)

```

غذا سرد است.

در ساختاری هایی که اسمی به اسم دیگر نسبت داده شده نیز دو اسم با رابطه domain: نمایش داده می‌شوند. این

نکته را مدنظر قرار دهید که اسمی که نقش توصیفی دارد در ریشه نامفا قرار می‌گیرد:

وکیل / x
:domain (x2 / مرد)

مرد یک وکیل بود.

با این حال ممکن است در جمله‌ای اسمی که توصیف می‌شود کانون جمله باشد:

مرد / x
:mod (x2 / وکیل)

مردی که یک وکیل بود.

نامفا نمایش ساده‌ای را برای «است»، «هست» و «وجود داشتن» در نظر گرفته‌است:

کتاب / x

کتاب
یک کتاب هست
کتابی وجود دارد.

کتاب / x
:location (x2 / اتاق)

یک کتاب داخل اتاق است.
کتابی که داخل اتاق وجود دارد.

ساخت اسنادی «شدن»

در ساخت‌های اسنادی که با «شدن» ساخته می‌شوند، «شدن» را رویداد در نظر می‌گیریم. ساخت موضوعی «شدن» به صورت زیر در نظر گرفته می‌شود:

شدن

Arg1: موجودی که دستخوش تغییر قرار می‌گیرد
Arg2: وضعیت جدید

شدن / x

:ARG1 (x2 / غذا)

:ARG2 (x3 / سرد)

غذا سرد شد.

شدن / x

:ARG1 (x2 / مرد)

:ARG2 (x3 / وکیل)

مرد وکیل شد

ساخت اسنادی سببی

در این نوع ساخت‌ها «کردن» رویداد قرار می‌گیرد. ساختار موضوعی «کردن» به صورت زیر است:

کردن

Arg0: عامل تغییر
Arg1: موجودی که دستخوش تغییر قرار می‌گیرد
Arg2: وضعیت جدید

کردن / x

:ARG0 (x4 / او)

:ARG1 (x2 / غذا)

:ARG2 (x3 / سرد))

او غذا را سرد کرد.

افعال مرکب

با توجه به ماهیت زبان فارسی، برخلاف زبان انگلیسی بخش سبک افعال مرکب حذف نمی‌شوند. همچنین نامفا توجهی به فاصله بین بخش اسمی و فعلی افعال مرکب ندارد. به مثال‌های زیر توجه کنید.

لطمه‌زدن / x

:ARG0 (x2 / تگرگ)

:ARG1 (x3 / باغ

:poss (x4 / من))

:mod (x5 / بد)

:time (x6 / دیروز))

تگرگ لطمه بدی دیروز به باغ من زد.

تگرگ دیروز به باغ من بد لطمه زد.

به‌منظور یک‌دست‌سازی جملات نامفانویسی شده در پیکره، افعال مرکبی که می‌توانند بدون تغییر معنی با افعال سبک متفاوتی استفاده شوند را یکسان‌سازی می‌کنیم.

بیرون‌کردن / x

:ARG0 (x2 / علی)

:ARG1 (x3 / من))

علی من را بیرون‌کرد.

علی من را بیرون‌انداخت.

همچنین افعالی مثل «نقاشی‌کشیدن» به صورت «نقاشی‌کردن» در نامفا نمایش داده می‌شوند.

افعال تناوبی

افعال تناوبی می‌توانند در سه دسته جای‌گیرند. در دسته اول ریشه دو فعل متفاوت است. برای مثلاً «کشتن» و «مردن» در این دسته قرار می‌گیرند. نامفا چنین افعالی را بدون تغییر وارد گراف می‌کند. در دسته دوم فعل سببی و خودانگیخته با یک ریشه واحد در جملات ظاهر می‌شوند. «ریختن» مثالی از این دسته است. در دو جمله «آب ریخت.» و «علی آب را ریخت.» رویداد «ریختن» را داریم که در جمله اول رویداد فاقد $arg0$: است و جمله دوم $arg0$: دارد. در چنین مواردی نامفا تمایزی بین دو حالت رویداد نمی‌دهد. در دسته سوم تناوب در فعل سبک دیده می‌شود. از آنجا که ریشه بخش واژگانی یکی است و فعل سبک بار معنایی بیار کمی دارد، رویداد سببی که ساختار موضوعی کاملیدارد رانشان می‌دهد و در صورتی که فعل خودانگیخته است، مانند دسته دوم $arg0$: را نمایش نمی‌دهد.

آب‌کردن / x
 ((ریخ / $x2$) :ARG1)

NOT:

آب‌شدن / x
 ((ریخ / $x2$) :ARG1)

ریخ آب شد.

آب‌کردن / x
 (دلک / $x3$) :ARG0
 ((ریخ / $x2$) :ARG1)

دلک ریخ را آب کرد.

اسم‌هایی که رویداد می‌شوند

تمایل شدید نمایش انتزاعی معنای انگلیسی به رویداد سازی اسامی از یکسو و قابلیت زبان فارسی در تولید فعل مرکب با استفاده از اسامی موجود در زبان از سوی دیگر ممکن است این تصور را ایجاد کند که شاید بهتر باشد بسیاری از اسامی در زبان فارسی را نیز به رویداد تبدیل کنیم. اما در نامفا تنها اسم‌هایی گزاره‌ای به رویداد تبدیل می‌شوند:

01-لطمه زدن (x /

:ARG0 (x2 / ماری)

:ARG1 (x3 / دلچک))

ماری به دلچک لطمه زد.

لطمه ماری به دلچک.

دلچک از ماری لطمه خورد.

01-انجام دادن (x /

:ARG0 (x2 / دلچک)

:ARG1 (x3 / کار))

دلچک کار را انجام داد.

کار به دست دلچک انجام شد.

انجام کار توسط دلچک.

انجام کار به دست دلچک.

اسامی مانند «کتاب» و «گل» به رویدادهای «کتابت کردن» و «گل کردن» تبدیل نمی‌شوند.

صفت‌هایی که رویداد می‌شوند

صفت‌ها نیز مانند اسامی قابلیت رویداد شدن را دارند:

داستان (x /

:ARG0-of (x2 / جذب کردن))

داستان جذاب

جذب کردن / x

((داستان / x2) :ARGO

داستان جذاب است.

داستان جذب می‌کند.

چطوری بشه؟

جذب کردن / x

((داستان / x2) :ARGO

((دلچسب / x2) :ARGO

داستان برای دلچسب جذاب است.

داستان دلچسب را جذب می‌کند.

ARGO معمولاً به چیزی یا کسی اشاره می‌کند که توسط صفت توصیف می‌شود. درحالی‌که، ARG1 به طبیعی‌ترین آرگومان بعدی اشاره دارد. چنانچه صفت کنادی (حالت کنشگر) نباشد، از ARG1/ARG2 به جای ARG0 استفاده می‌کنیم. حال در برخورد با صفت‌هایی همچون “ناراحت” چه باید کرد؟ آیا می‌بایست از حالت سببی رویداد استفاده کرد و یا کنادی؟ در چنین شرایطی از قاب‌های آماده‌ای که نامفا تهیه و گردآوری کرده است استفاده می‌کنیم:

ناراحت / x

((ماری / x2) :ARG1

ماری ناراحت است.

صفت‌های وجهی که با قابل، پذیر، و —ی نسبت می‌آیند نیز مانند مثال‌های زیر نامفانویسی می‌شوند:

شیشه / x

:ARG1-of (x2 / شکستن

:ARG1-of (x3 / امکان داشتن))

شیشه شکستنی است.

تلاش کردن / x

:ARG0 (x2 / او)

:ARG1-of (x3 / درک کردن

:ARG1-of (x4 / امکان داشتن))

تلاش او قابل درک است.

قید

در نامفا، گروه قیدی را به صفت تبدیل می‌کنیم که این صفت‌ها اغلب دارای قاب مخصوص به خود هستند:

ترک کردن / x

:ARG0 (x2 / دلچک)

:ARG1 (x3 / خانه)

:manner (x4 / سریع)

دلچک به سرعت خانه را ترک کرد.

نقش‌های غیراصلی

تا اینجا با نقش‌هایی مانند time و location: آشنا شدیم. اما در نمایش انتزاعی معنا، نقش‌های غیراصلی دیگری نیز وجود دارند:

دقت داشته‌باشید که برخی نقش‌های غیراصلی در واقع بخشی از ساختار موضوعی رویداد می‌باشند که با نقش‌های اصلی argx: وارد نامفا می‌شوند.

:source

:destination

رانندگی کردن (x /

:ARG0 (x2 / او)

:direction (x3 / شمال)

:source (x4 / تهران)

:destination (x5 / مازندران))

او از تهران تا مازندران، به سمت شمال رانندگی کرد .

:path

رفتن (x /

:ARG0 (x2 / او)

:ARG4 (x3 / تهرانپارس)

:path (x4 / اتوبان))

او از اتوبان به تهرانپارس رفت .

:beneficiary
:accompanier

(x / آوازخواندن
:ARG0 (x2 / دلچک)
:beneficiary (x4 / ماری)
:time (x5 / راهرفتن
:ARG0 x3
:accompanier x2
:destination (x7 / شهر)))

دلچک در حالی با ماری تا شهر راه می‌رفت، برایش آوازی خواند.

:topic

(x / -polarity: اطلاعات
:topic (x2 / پرونده))

اطلاعاتی درمورد پرونده وجود ندارد.

(p / person :wiki -
:name (n / name :op1 "جی" :op2 "بارتروف")
:ARG0-of (h / have-org-role-91
:ARG1 (u / university
:wiki "دانشگاه کالیفرنیا جنوبی"
:name (n2 / name :op1 "اس.سی.یو")
:ARG2 (p2 / استادیار
:mod (m / ریاضی))))

استادیار ریاضی دانشگاه یو.اس.سی، جی بارتروف

:duration

کارکردن (x /
:ARG0 (x2 / او)
:duration (t / temporal-quantity :quant 2
:unit (x3 / ساعت)))

او دو ساعت کار کرد .

:instrument

این نقش توصیف‌کننده موجودیتی فیزیکی است . برای مثال، ابزار، وسیله، سلاح، اعضای بدن از جمله انگشت یا پا که در عملی مورد استفاده قرار گرفته‌است. ازطرفی، چنانچه بخواهیم عملی را از نظر چگونگی رخ دادن توصیف کنیم از نقش کلی‌تر *manner* : استفاده می‌کنیم.

خوردن (x /
:ARG0 (x2 / او)
:ARG1 (x3 / غذا)
:instrument (x4 / دست))

او غذا را با دست خورد .

حمله‌کردن (x /
:ARG0 (x2 / عراق)
:instrument (x3 / موشک))

حمله موشکی عراق

:medium

نقش *medium* : برای اشاره به راه‌های ارتباطی بشر مانند روزنامه، تلویزیون، اینترنت، یوتیوب، فیس بوک، سخنرانی، و زبان‌ها استفاده می‌شود.

صحبت کردن / (x

:ARG0 (x2 / ماری)

:ARG2 (x3 / دلک)

:medium (x4 / فرانسوی))

ماری به فرانسوی با دلک صحبت کرد .

اعلام کردن / (x

:ARG0 (x2 / علی)

:ARG1 (x3 / به دنیا آمدن

:ARG1 (x4 / شخص

:ARG0-of (h / have-rel-role-91

:ARG1 x2

:ARG2 (x5 / پسر))))

:medium (x6 / اینستاگرام))

علی به دنیا آمدن پسرش را در اینستاگرام اعلام کرد .

:manner

این نقش در نامفانووسی چگونگی انجام شدن عملی استفاده می شود. باید توجه داشت که شرح چگونگی وقوع عمل به وسیله نقش های جزئی تر medium: و instrument: قابل توصیف نباشد.

خواندن / (x

:ARG0 (x2 / دلک)

:ARG1 (x3 / آواز)

:manner (x4 / زیبا))

دلک به زیبایی آواز می خواند .

تزئین کردن (x /

:ARG0 (x2 / او)

:ARG1 (x3 / اتاق)

:manner (x4 / خلافتانه))

او به طرز خلافتانه‌ای اتاق را تزئین کرد .

به علاوه، از نقش manner: می‌توان برای توصیف روش انجام عملی (با استفاده از، به کمک) نیز استفاده کرد.

پیشنهاد کردن (x /

:ARG0 (x2 / شخص

:ARG0-of (h / have-org-role-91

:ARG2 (x3 / شهردار)))

:ARG1 (x4 / کاهش دادن

:ARG1 (x5 / جرم)

:manner (x6 / استخدام کردن

:ARG2 (x7 / شخص

:ARG0-of (h2 / have-org-role-91

:ARG1 (x8 / پلیس)

:ARG2 (x9 / افسر))

:mod (x10 / بیشتر))))))

شهردار پیشنهاد کرد که با استخدام بیشتر افسران پلیس جرائم را کاهش دهند.

برای بیان وسیله نقلیه‌ای که شخص یا چیزی با آن به مقصد رسیده است نیز از manner: استفاده می‌کنیم.

رفتن (x /

:ARG0 (x2 / ماری)

:ARG4 (x3 / فرانسه)

:manner (x4 / قطار))

ماری با قطار به فرانسه رفت .

:purpose

رفتن / x)
:ARG0 (x2 / او)
:ARG4 (x3 / انگلستان)
:purpose (x4 / درمان کردن
:ARG1 x2
:ARG2 (x5 / بیماری
:poss x2)))

او برای درمان بیماری‌اش به انگلستان رفت .

:cause

صحبت کردن / x)
:ARG0 (x2 / دلچک)
:manner (x3 / نرم)
:purpose (x4 / تسلی دادن
:ARG0 x2
:ARG1 (x5 / ماری))
:cause (x6 / دوست داشتن
:ARG0 x2
:topic x5))

دلچک به نرمی صحبت می‌کرد تا ماری را تسلی دهد، چرا که دوستش داشت .

توجه داشته باشید که ادیتور نامفا به صورت خودکار cause: و cause-of: را در جهت رویدادسازی به cause-01 تبدیل می‌کند، بنابراین cause: را یک میانبر می‌نامیم.

:concession

برگزارکردن (x /
:ARG0 (x5 / آنها
:ARG1 (x2 / بازی)
:concession (x3 / باریدن
:ARG0 (x4 / باران))

علی رغم بارندگی بازی را برگزار کردند .

برگزارکردن (x /
:ARG0 (x2 / ما)
:ARG1 (x3 / بازی)
:concession (x4 / اگر-حتی
:op1 (x5 / باریدن
:ARG0 (x6 / باران))))

بازی را برگزار می‌کنیم، حتی اگر باران بیارد .

:condition

خواندن (x /
:ARG0 (x2 / دلچک)
:ARG1 (x3 / آواز)
:condition (x4 / دادن
:ARG1 (x5 / پول)
:ARG2 (x2))

اگر به دلچک پول بدهند آواز می‌خواند .

- polarity: خواندن / x)
 :ARG0 (x2 / دلک)
 :ARG1 (x3 / آواز)
 :condition (x4 / دادن
 :ARG1 (x5 / پول)
 :ARG2 x2))

دلک آواز خواهد خواند مگر آنکه به او پول بدهند .
 اگر به دلک پول ندهند، آواز خواهد خواند.

در نمایش انتزاعی معنا

X : cause Y

به این معناست که Y علت X است. همچنین،

X : cause-of Y

یعنی Y علت X است.

برای نحوه استفاده از مفهوم cause-01 به جای cause : و cause-of : به مبحث «قاب‌گری» مراجعه کنید.

حمله کردن / x)
 :ARG0 (x2 / کوسه)
 :cause-of (x3 / آسیب‌پیدا کردن
 :ARG1 (x4 / کشتی))

حمله کوسه‌ها باعث شد کشتی آسیب ببیند .
 حمله کوسه‌ها باعث آسیب به کشتی شد.
 در اثر حمله کوسه‌ها کشتی آسیب دید.

زمانی که نقش‌هایی مثل location : ، beneficiary : و تعداد دیگری از نقش‌های غیراصلی بخشی از ساختار موضوعی اصلی رویداد باشند، با مفاهیم مربوطه ب.

انتخاب کردن (x /

:ARG0 (x2 / دلک)

:ARG1 (x3 / گل)

:ARG3 (x4 / ماری))

NOT

انتخاب کردن (x /

:ARG0 (x2 / دلک)

:ARG1 (x3 / گل)

:beneficiary (x4 / ماری))

دلک برای ماری گلی را انتخاب کرد .

گاهی در رویدادهایی که معنای «ساختن»، «خلق کردن» و یا «بناکردن» را می‌رسانند به روشنی مشخص نیست که روابط location: و time: ذیل چه مفهومی بیایند. برای مثال در جمله زیر «مریلند» مکان پل است یا مکان رویداد ساختن. از این رو در چنین شرایطی این روابط را ذیل رویداد می‌آوریم.

(x / ساختن

:ARG0 (x2 / آنها)

:ARG1 (x3 / پل)

:location (s / state :wiki "مریلند"
:name (n / name :op2 "مریلند"))
:time (d / date-entity :month 12))

NOT

(x / ساختن

:ARG0 (x2 / آنها)

:ARG1 (x3 / پل

:location (x4 / ...)))

آنها پل را در مریلند، در ماه دسامبر ساختند .
پل توسط آنها در مریلند، در ماه دسامبر ساخته شد.

:part

(x / موتور

:part-of (x2 / ماشین))

موتور ماشین

واحد / x
:part-of (x2 / شرکت))

یکی از واحدهای شرکت
واحد شرکت

جنوب / x
:part-of (x2 / فرانسه))

جنوب فرانسه

توجه داشته باشید که از part: برای بیان عضویت در گروه استفاده نمی‌شود، مانند اعضای هیئت مدیره.

:subevent

بردن / x
:ARG0 (x2 / او)
:ARG1 (x3 / مسابقه
:subevent-of (x4 / المپیک)))

او در المپیک مسابقه را برد .

:consist-of

(x / حلقه
:consist-of (x2 / طلا))

یک حلقه از طلا

(x / تیم
:consist-of (x2 / خلبان))

تیمی از خلبانان

:example

(x / شرکت
:example (x2 / و
:op1 (x3 / گوگل)
:op2 (x4 / آی بی ام)))

شرکت‌هایی مانند گوگل و آی بی ام

:direction

حرکت کردن (x /
:ARG0 (x2 / او)
:direction (x3 / غرب))

او به سمت غرب حرکت کرد .

:frequency

frequency: تعدد وقوع یک عمل را توصیف می کند.

3 frequency: ملاقات کردن (x /
:ARG0 (x2 / من)
:ARG1 (x3 / او))

من سه بار او را ملاقات کردم .

قاب ویژه rate-entity-91 برای توصیف رویدادهای تکرارشونده‌ای مانند “هر سه هزار مایل” و یا “گالنی سه دلار” استفاده می شود.

(r / rate-entity-91
:ARG1 2
:ARG2 (t / temporal-quantity :quant 1
:unit (x / سال)))

دو بار در سال
سالی دو بار

بازی کردن (x /
 :ARG0 (x2 / ما)
 :ARG1 (x3 / گلف)
 :frequency (r / rate-entity-91
 :ARG4 (d / date-entity
 :weekday (x4 / پنجشنبه)
 :dayperiod (x5 / بعدازظهر))))

هر پنجشنبه بعدازظهر گلف بازی می‌کنیم .

نقش‌های اصلی: rate-entity-91 در جدول زیر آمده‌اند:

rate-entity-91:

Arg1: مقدار
 Arg2: مقدار مرجع («در هر» مقدار)
 Arg3: فواصل منظم بین وقوع رویدادها («هر دو ماه» – تفاوت آن با ARG2: در دقیق‌تر بودن است)
 Arg4: موجودیتی که در آن رویداد تکرارشونده اتفاق می‌افتد

:extent

رفتن (x /
 :ARG1 (x2 / جاده
 :mod (x3 / این))
 :extent (x4 / بینهایت))

این جاده تا بینهایت می‌رود .

کانون

برای کانون قرار دادن مفاهیم از روابط معکوس استفاده می‌کنیم (برای اطلاعات بیشتر به مقدمه رجوع شود).

```
رقصیدن / x  
  دلک (x1 / ARG0  
    :source (x2 / سیرک)))
```

دلک سیرک رقصید.

```
دلک / x  
  رقصیدن / x1 / ARG0-of  
  :source (x2 / سیرک)))
```

دلکی از سیرک رقصید.
دلکی که در سیرک است رقصید.

```
سیرک / x  
  :source-of (x1 / دلک  
    :ARG0-of (x2 / رقصیدن)))
```

سیرکی که دلک رقصان از آنجا بود.

مفهوم کانون تنها هنگام انتخاب ریشه از اهمیت برخوردار است. پس از انتخاب ریشه، کانون موضوعیتی ندارد و ادامه کار تنها با توجه به روابط معنایی خواهد بود. برای مثال زمانی که (سیرک \ x) سیرک را به عنوان ریشه انتخاب کردیم، نمی‌توانیم source-of را با «رقصیدن» مقداردهی کنیم و باید مفهوم «دلک» را برای آن انتخاب کنیم.

قاب‌گری

تبدیل نقش به یک مفهوم را قاب‌گری می‌نامیم به این معنا که گاهی نیاز است رابطه‌ای را با رویدادی جایگزین کنیم. برای مثال به جای رابطه cause: از رویداد cause-01 استفاده می‌کنیم. در این مورد به جای

x :cause y

از

x :ARG1-of (c/ cause-01 :ARG0 y)

استفاده می‌کنیم.

از آنجا که روابط در نامفا با برجسب‌های انگلیسی نمایش داده می‌شوند. قاب‌هایی که از آن برای قاب‌گری روابط استفاده می‌شوند مستقیم از نمایش انتزاعی معنا انگلیسی گرفته شده‌اند.

بدون قاب‌گری

رفتن / (x
:ARG0 (x2 / ماری)
:cause (x3 / آمدن
:ARG1 (x4 / دلقک))))

ماری رفت، چون دلقک آمد.

با قاب‌گری

رفتن / (x
:ARG0 (x2 / ماری)
:ARG1-of (c / cause-01
:ARG0 (x3 / آمدن
:ARG1 (x4 / دلقک))))

ماری رفت، چون دلقک آمد.

بدون قاب‌گری نامفا ساده‌تر است. پس چرا قاب‌گری می‌کنیم؟

یکی از دلایل این کار کانونی کردن یک رابطه است. برای مثال می‌دانیم که چاقویی در کشور است. ممکن است بخواهیم چاقو را کانون قرار دهیم:

دانستن / (x
:ARG0 (x2 / ما)
:ARG1 (x3 / چاقو
:location (x4 / کشور))

می‌دانیم چاقویی که در کشور است.

و یا کشور را کانون کنیم:

دانستن / x)

:ARG0 (x2 / ما)

:ARG1 (x3 / کشو

:location-of (x4 / چاغو)))

کشویی که چاقو در آن است را می‌دانیم.

اما در واقع می‌خواهیم مکان چاقو را کانون قرار دهیم. در این راستا، نامفا بسیاری از روابط را قاب‌گیری کرده است. برای نمونه location: به be-located-at-91 قاب‌گیری شده و به ما اجازه می‌دهد تا نامفای زیر را بسازیم:

دانستن / x)

:ARG0 (x2 / ما)

:ARG1 (b / be-located-at-91

:ARG1 (x4 / چاغو)

:ARG2 (x3 / کشو)))

می‌دانیم چاقو در کشو است.

زمانی که می‌خواهیم توصیفی را به رابطه‌ای نسبت دهیم نیز از قاب‌گیری استفاده می‌کنیم:

دانستن / x)

:ARG0 (x2 / ما)

:ARG1 (b / be-located-at-91 :polarity -

:ARG1 (x4 / چاغو)

:ARG2 (x3 / کشو)

:time (x5 / دیروز)))

می‌دانیم چاقو دیروز در کشو نبود.

روابط زیر قاب‌گیری نشده‌اند:

```
:ARG0, :ARG1, :ARG2, ... :op1, :op2, :op3, :op4, ...  
:calendar, :century, :day, :dayperiod, :decade, :era, :month,  
:quarter, :season, :timezone, :weekday, :year, :year2  
:unit, :direction, :scale
```

در جدول صفحه بعد لیست روابط قاب‌گیری شده آمده است.

Relation	Reification	Domain	Range	Example
:accompanier	accompany-01	:ARG0	:ARG1	“she’s with him”
:age	age-01	:ARG1	:ARG2	“she’s 41 years old”
:beneficiary	benefit-01	:ARG0	:ARG1	“the 5k run is for kids”
:cause	cause-01	:ARG1	:ARG0	“he came ’cause of her”
:concession	have-concession-91	:ARG1	:ARG2	“he came despite of her”
:condition	have-condition-91	:ARG1	:ARG2	“he comes if she comes”
:degree	have-degree-91	:ARG1	:ARG2	“very tall” (intensifier or downtoner)
:destination	be-destined-for-91	:ARG1	:ARG2	“i’m off to Atlanta”
:duration	last-01	:ARG1	:ARG2	“it’s 15 minutes long”
:example	exemplify-01	:ARG0	:ARG1	“cities such as Atlanta”
:extent	have-extent-91	:ARG1	:ARG2	“trip was 2500 miles”
:frequency	have-frequency-91	:ARG1	:ARG2	“he came three times”
:instrument	have-instrument-91	:ARG1	:ARG2	“forks are for eating”
:location	be-located-at-91	:ARG1	:ARG2	“she’s not here”
:manner	have-manner-91	:ARG1	:ARG2	“it was done quickly”
:mod	have-mod-91	:ARG1	:ARG2	“he is half Chinese”
:name	have-name-91	:ARG1	:ARG2	“the city formerly named Constantinople”
:part	have-part-91	:ARG1	:ARG2	“the roof of the house”
:polarity	have-polarity-91	:ARG1	:ARG2	“I don’t know.”
:poss	own-01, have-03	:ARG0	:ARG1	“that dog’s not mine”
:purpose	have-purpose-91	:ARG1	:ARG2	“it’s to eliminate bugs”
:quant	have-quant-91	:ARG1	:ARG2	“there are 4 rabbits”
:source	be-from-91	:ARG1	:ARG2	“she’s from Ipanema”
:subevent	have-subevent-91	:ARG1	:ARG2	“presentation at a conference”
:subset	include-91	:ARG2	:ARG1	“10% of the workers”
:time	be-temporally-at-91	:ARG1	:ARG2	“the party is on friday”
:topic	concern-02	:ARG0	:ARG1	“the show’s about me”
:value	have-value-91	:ARG1	:ARG2	“The phone number is 1-800-555-1223.”

اما چه زمانی از قاب‌گری استفاده کنیم؟ یک پاسخ می‌تواند این باشد که «هر زمان احساس نیاز کردید». در اینصورت، متأسفانه یک جمله ممکن است با دو نامفا نمایش داده شود بدون اینکه هیچ‌یک نسبت به دیگری برتری داشته باشد. هر دو نامفای زیر می‌توانند نمایشی منطقی برای جمله «ماری رفت چون دل‌ک‌آمد» باشند:

بدون قاب‌گری

```
رفتن (x /
      :ARG0 (x2 / ماری)
      :cause (x3 / آمدن
              :ARG1 (x4 / دل‌ک‌)))
```

با قاب‌گری

```
رفتن (x /
      :ARG0 (x2 / ماری)
      :ARG1-of (c / cause-01
                :ARG0 (x3 / آمدن
                      :ARG1 (x4 / دل‌ک‌)))
```

پاسخ دیگر این است که «همواره از قاب‌گری استفاده کنید». در اینصورت روابطی چون `cause`، `location` و `subset`: به کلی از نامفا حذف می‌شوند و به جای آنها مفاهیم `cause-01`، `be-located-at-91` و `include-91` مورد استفاده قرار می‌گیرند. اما این کار را دشوار می‌کند و نوشتن `location`: به مراتب ساده‌تر و مرسوم‌تر است. بطور کلی، ساختار نمایش انتزاعی معنایی که در آن از قاب‌گری استفاده شده است به ساختار مدنظر پایه‌گذاران آن نزدیکتر است. هدف تیم اصلی نمایش انتزاعی معنای انگلیسی در نهایت تبدیل تمامی روابط به روابطی قاب‌گیری شده است. اما در حال حاضر نیازی به حساسیت روی این موضوع نیست و نامفانویسان می‌توانند هر زمان که خواستند از قاب‌گری استفاده نمایند.

حروف اضافه

بیشتر حرف‌های اضافه‌ای که نشان‌گر روابط معنایی هستند حذف می‌شوند.

عاشق شدن (x /
:ARG0 (x2 / دلکک)
:time (d / date-entity :month 8
:calendar (s / solar-hijri)))

دلکک در بهمن عاشق شد.

مردن (x /
:ARG1 (x2 / ماری)
:location (x3 / خانه
:poss x2))

ماری در خانه‌اش مرد.

بالین‌حال، حروف اضافه‌ای که بیانگر زمان و مکان هستند اگر اطلاعات بیشتری در خود داشته باشند حفظ می‌شوند و درحروف اضافه مرکب، بخشی که حاوی اطلاعات بیشتر نیست حذف می‌شود. برای این کار از opN: استفاده می‌کنیم. op1: مورد استفاده در این حالت با op1: حروف ربط متفاوت است.

عاشق شدن (x /
:ARG0 (x2 / دلکک)
:time (x3 / بعد
:op1 (x4 / جنگ)))

دلکک بعد از جنگ عاشق شد.

مردن / x)
 :ARG1 (x2 / ماری)
 :location (x3 / نزدیک
 :op1 (x4 / خانه
 :poss x2)))

ماری نزدیک خانه‌اش مرد.
 ماری در نزدیکی خانه‌اش مرد.

کشتن / x)
 :ARG1 (x2 / ماری)
 :location (x3 / بین
 :op1 (x4 / رودخانه
 :op2 (x5 / جنگل)))

ماری بین رودخانه و جنگل کشته شد.
 ماری در میان رودخانه و جنگل کشته شد.

در برخی موارد عبارت حرف اضافه‌ای در قالب محمول و آرگومان نمی‌گنجد. همچنین امکان استفاده از روابطی مانند location یا time نیز وجود ندارد. در چنین شرایطی نامفا در کمال شرمندگی، از رابطه prep-x: استفاده می‌کند.

آسیب‌زدن / x)
 :ARG1 (x2 / دل‌تک)
 :prep-dar (x3 / حادثه))

دل‌تک در حادثه آسیب‌دید.

در چنین مواردی کل حرف اضافه مرکب حذف می‌شود

امضا کردن (x /

:ARG0 (x4 / او)

:prep-be-jay (x2 / مادر

:poss x4))

او به جای مادرش امضاء کرد.

بند موصولی

همانطور که در مقدمه تشریح شده است، در نامفا معمولاً بندهای موصولی را با روابط معکوس نشان می‌دهیم.

ترساندن (x /

:ARG1 (x2 / پسر))

پسر می‌ترسد.

پسر (x /

:ARG1-of (x2 / ترساندن))

پسری که می‌ترسد.

در زبان فارسی، زمانی که منفی کردن صفت‌ها مشکل‌زا است، از بندهای موصولی استفاده می‌شود.

جوشیدن (x /

:ARG1 (x2 / آب))

آب جوش.

آب / x
(ARG1-of (جوشیدن / x2) polarity -))

آبی که جوش نیست.

در ژاپنی از پسوند منفی ساز استفاده می شود (آب نجوش!).

چند رابطه معنایی هم نام

هر مفهوم می تواند چندین رابطه هم نام داشته باشد. در مثال های زیر صفحه دو mod: دارد، رسیدن دو time: دارد و دلک arg0-of: دو رویداد خواستن و دیدن است.

صفحه / x
:mod (کیفیت / x2)
:mod (شفاف / x3)

صفحه با کیفیت و شفاف.

رسیدن / x
:ARG1 (او / x4)
:time (امروز / x2)
:time (صبح / x3)

او امروز، صبح رسید.

دلقک / x
 :ARG0-of (x2 / خواستن
 :ARG1 (x3 / دیدن
 :ARG1 (x4 / ماری)))
 :ARG0-of x2)

دلقکی که می‌خواست ماری را ببیند.

حروف ربط

در نامفا برای نمایش روابط بین عبارات از مفاهیم «و» «یا» و contrast-01 به همراه opx: استفاده می‌کنیم.

و / x
 :op1 (x2 / دلقک)
 :op2 (x3 / ماری)

دلقک و ماری

یا / x
 :op1 (x2 / خودگار)
 :op2 (x3 / مدار)
 :op3 (x3 / روان‌نویس)

یا خودکار یا مداد یا روان‌نویس

صفاتی که با حرف ربط آمده‌اند بدون «و» نامفا می‌شوند.

توپ / x
:mod (x4 / بزرگ)
:mod (x5 / سنگین))

توپ بزرگ و سنگین

برای بندها نیز از opx: استفاده می‌شود.

و / x
:op1 (x2 / دعواکردن)
:op2 (x3 / رفتن
:ARG0 (x4 / دلچک)))

دعوا شده بود و دلچک رفت.

c / contrast01-
:ARG1 (x2 / دعواکردن)
:ARG2 (x / ماندن
:ARG0 (x4 / دلچک)))

دعوا شده بود، اما دلچک ماند.

در برخی موارد op1 و arg1: وجود ندارند.

c / contrast01-
:ARG2 (x / ماندن
:ARG0 (x4 / دلچک)))

اما دلچک ماند.

نامفا استفاده از منطق را ترجیح می‌دهد، حتی زمانی که فاعل حذف به قرینه لفظی شده‌باشد.

(x / و
:op1 (x2 / فریادزدن
:ARG0 (x4 / ماری))
:op2 (x3 / رفتن
:ARG1 x4))

ماری فریاد زد و رفت.

نیاز به انجام این کار زمانی مشهود می‌شود که موجودیتی دو نقش متفاوت در دو رویداد متفاوت را دارد.

(x / و
:op1 (x2 / رسیدن
:ARG1 (x4 / دلچک))
:op2 (x3 / کشتن
:ARG1 x4
:manner (x5 / درجا))

دلچک رسید و درجا کشته‌شد.

با این حال، در نامفا نقش‌های فرعی مثل time و location: را «بیرون می‌کشد». در مثال زیر time: کل عبارت عاطفی که ریشه آن «و» است را شامل می‌شود.

```

و / x
  :op1 (x2 / رسیدن
        :ARG1 (x4 / دلچک))
  :op2 (x3 / رفتن
        :ARG1 x4)
  :time (d / date-entity
        :weekday (x5 / شنبه))

```

دلچک شنبه رسید و رفت.

کمیت‌ها و قلمرو

نامفا نمایش عمیقی را برای مقادیر کمیتی ارائه نمی‌دهد و تنها جایگاه آن‌ها را یکسان‌سازی کرده است.

```

رفتن / x
  :ARG0 (x2 / پسر
        :mod (x3 / همه))

```

همه پسرها رفتند.
پسرها همگی رفتند.
تک تک پسرها رفتند.

در فارسی «هیچ» فعل منفی می‌طلبد، دو مثال زیر درواقع یک معنا دارند.

```

رفتن / x
  :ARG0 (x2 / پسر
        :mod (x3 / هیچ))

```

هیچ‌یک از پسرها نرفتند.
هیچ پسری نرفت.

```
(x / رفتن :polarity -  
  :ARG0 (x2 / پسر  
    :mod (x3 / همه)))
```

پسرها، همگی نرفتند.
هیچ پسری نرفت.

با اینکه هر دو نمایش صحیح هستند، برای تمامی جملات دو مثال بالا گرایش نامفا به نمایش منطق محور، یعنی نمایش دوم است. در مثال زیر به جایگاه polarity: و تفاوت معنایی آن توجه داشته باشید.

```
(x / رفتن :polarity -  
  :ARG0 (x2 / پسر  
    :mod (x3 / همه :polarity -)))
```

همه پسرها نرفتند.
پسرها همه‌شان نرفتند.

انتخاب جایگاه polarity: در برخی موارد دشوار است. مثال‌های زیر را در نظر بگیرید:

```
(x / فکرکردن  
  :ARG0 (x2 / ماری)  
  :ARG1 (x3 / کارکردن  
    :ARG0 (x4 / دلچک)  
    :manner (x5 / خوب)))
```

ماری فکر می‌کند دلچک خوب کار می‌کند.
ماری فکر می‌کند کارکردن دلچک خوب است.

حال اگر ماری فکر کند نحوه کار کردن دلچک خوب نیست، باید polarity: را روی مفهوم «خوب» اعمال کنیم، و نه فکرکردن و یا کارکردن.

فکرکردن / x)

:ARG0 (x2 / ماری)

:ARG1 (x3 / کارکردن

:ARG0 (x4 / دلچک)

:manner (x5 / خوب :polarity -)))

ماری فکر می کند دلچک خوب کار نمی کند.

ماری فکر می کند کارکردن دلچک خوب نیست.

اگر polarity: جای دیگری بیاید، معنی جمله تغییر می کند:

فکرکردن / x)

:ARG0 (x2 / ماری)

:ARG1 (x3 / کارکردن :polarity -

:ARG0 (x4 / دلچک)

:manner (x5 / خوب)))

ماری فکر می کند نحوه کار نکردن دلچک خوب است.

(x / فکرکردن :polarity -

:ARG0 (x2 / ماری)

:ARG1 (x3 / کارکردن

:ARG0 (x4 / دلچک)

:manner (x5 / خوب)))

ماری این فکر را نمی کند که دلچک خوب کار می کند.

فکرکردن / x

:ARG0 (x2 / ماری: polarity -)

:ARG1 (x3 / کارکردن

:ARG0 (x4 / دلک: polarity -)

:manner (x5 / خوب))

آنکه ماری نیست فکر می‌کند آنکه دلک نیست خوب کار می‌کند.

از اینکه نامفا در مورد جایگاه polarity:

در شیوه نامه نمایش انتزاعی معنای انگلیسی نویسندگان از این که برای مفاهیم کمی پیشنهادی ارائه نکرده‌اند پوزش می‌طلبند.

درجه

قیده‌های شدت (خیلی، بسیار) و تضعیفی (برخی، به نسبت) را با degree: نشان می‌دهیم:

سرد \ x

:domain (x2 \ هوا)

:degree (x3 \ خیلی))

هوا خیلی سرد است.

برای نمایش صفات برتر و برترین (تفضیلی و عالی) از قاب انگلیسی have-degree-91 استفاده می‌کنیم.

have-degree-91

Arg1: موصوف، آنچه ویژگی یا صفت به آن نسبت داده شده. (مثل «هوا»)

Arg2: صفت (مثل «سرد»)

Arg3: میزان، درجه (مثل «بیش‌تر» «کافی» «خیلی» «تر» «سترین»)

Arg4: آنچه معیار قیاس واقع شده است (مثل (...تر از) هوای دیروز)

Arg5: برترین: مجموعه مرجع قیاس

Arg6: مرجع یا مرز کفایت (مثل، (کافی نیست) تا مدرسه‌ها تعطیل شوند)

به نامفا نویسان پیشنهاد می‌شود که زمانی have-degree-91 را در ریشه قرار دهند که مقایسه کانون مرکزی و اصلی جمله است (در قیاس با صفت یا موجودیتی که با قید یا صفتی توصیف شده است) . این مورد ساخت‌های اسنادی را شامل می‌شود. مثال‌های زیر تفاوت میان این دو را نشان می‌دهند.

دلقک / x

```
:ARG1-of (h / have-degree-91
  :ARG2 (x2 / هوش)
  :ARG3 (x3 / برتر))
```

دلقک باهوش‌تر

دلقک / x

```
:ARG1-of (h / have-degree-91
  :ARG2 (x2 / هوش)
  :ARG3 (x3 / برترین))
```

باهوش‌ترین دلقک

برنامه / x

```
:ARG1-of (h / have-degree-91
  :ARG2 (x2 / خوب)
  :ARG3 (x3 / برتر))
```

برنامه بهتر

برنامه / x

```
:ARG1-of (h / have-degree-91
  :ARG2 (x2 / بد)
  :ARG3 (x3 / برترین))
```

بهترین برنامه

برنامه (x /

:ARG1-of (h / have-degree-91

:ARG2 (x2 / افراطی)

:ARG3 (x3 / خیلی)))

برنامه‌ای که خیلی افراطی است.

(h / have-degree-91

:ARG1 (x / دلچسپ)

:ARG2 (x2 / بلند)

:ARG3 (x3 / برتر)

:ARG4 (x4 / ماری)))

دلچسپ از ماری بلندتر است.

در مواجهه با ساخت‌های دربردارنده صفت عالی که رابطه زیرمجموعه و فوق مجموعه را نیز دربرمی‌گیرند از have-degree-91 استفاده می‌شود و استفاده از آن بر include-91 تقدم دارد.

(h / have-degree-91

:ARG1 (x / او)

:ARG2 (x2 / بلند)

:ARG3 (x3 / برترین)

:ARG5 (x4 / دلچسپ

:ARG0-of (h2 / have-org-role-91

:ARG1 (x5 / سیرک)))

او بلندترین دلچسپ سیرک است.

در برخی ساخت‌ها درجه، میزان فاصله تا صدق حالتی را توصیف می‌کند که آن حالت موجب فراهم شدن شرایطی خواهد شد. برای چنین مواردی از آرگومان ششم have-degree-91 استفاده می‌شود.

(h / have-degree-91

:ARG2 (x / زود)

:ARG3 (x2 / خیلی)

:ARG6 (x3 / نتیجه‌گیری))

برای نتیجه‌گیری خیلی زود است.

توجه داشته باشید که همچون مثال بالا آرگومان اصلی نتیجه، می‌تواند به قید تبدیل نشود. در چنین حالتی نامفا نویسان باید منطقی‌ترین رابطه‌ای که با بافت همخوانی دارد را به کار گیرند. ممکن است اضافه کردن وجهیت، امکان و یا قطبیت وسوسه برانگیز باشد (به عبارت دیگر نشان دهیم که نتیجه حاصل نشده، یا نمی‌توان نتیجه‌گیری کرد). اما بررسی تعداد زیادی از چنین ساخت‌هایی نشان داده که امکان ورود چنین روابطی به طور یکنواخت امکان‌پذیر نیست. پس از نامفا نویسان درخواست می‌شود که arg6 را با رابطه‌ای ساده با آنچه مرجع «درجه» است نمایش دهند. قطبیت منفی تنها زمانی که به طور مستقیم ذکر شده است (have-degree-91 را منفی می‌کند) استفاده شود.

(h / have-degree-91 :polarity -

:ARG1 (x / او)

:ARG2 (x2 / بلند)

:ARG3 (x3 / کافی)

:ARG6 (x4 / سوارشدن

:ARG0 x

:ARG1 (x5 / ترن‌هوایی)))

برای سوارشدن به ترن‌هوایی به مقدار کافی بلند نیست.

متغیرها و هم‌مرجع بودن

اگر نام دو متغیر یکسان است، به مرجعی یکسان اشاره می‌کنند.

خواستن / x

:ARG0 (x2 / دلک)

:ARG1 (x3 / رفتن

:ARG0 x2))

دلک می‌خواهد برود.

در زبان فارسی، ضمائر و شناسه‌ها نمایش‌گر هم‌مرجع بودن هستند، اما در نامفا از متغیرها استفاده می‌شود.

خواستن / x

:ARG0 (x2 / دلک)

:ARG1 (x3 / دیدن

:ARG0 x2

:ARG1 x2))

دلک می‌خواهد خود را ببیند.

اگر مرجع ضمیر در جمله حاضر نباشد، نامفا خود ضمیر را به‌کار می‌برد.

دیدن / x

:ARG0 (x2 / او)

:ARG1 (x3 / ایشان))

او دیدشان.

مالکیت

نامفا از رابطه poss: برای نمایش مالکیت استفاده می‌کند.

کفش / x

:poss (x2 / دلک))

کفش دلک کفش مال دلک است.

قهرمان / x
((کشور / x2) :poss)

قهرمانان کشور

در نظر داشته باشید که مالکیت همواره با poss: نمایش داده نمی‌شود. نامفا در بسیاری از موارد از روابطی چون part-of: و consist-of: و غیره استفاده می‌کند.

صفات نسبی

صفات نسبی، مثل پسته‌ای به معنی «آنچه از ویژگی‌های پسته بهره می‌برد»، در نامفا وجود ندارند و ریشه اسم آن به همراه mod: می‌آید.

سبز / x
((پسته / x2) :mod)

سبز پسته‌ای

رفتار / x
((محترم / x2) :mod)

رفتار محترمانه

در صورتی که صورت اسمی صفات نسبی گزاره‌ای باشد به صورت رویداد تبدیل خواهند شد.

ادعا / x
((دروغ‌گفتن / x2) :mod)

ادعای دروغین
ادعای دروغ

اعداد ترتیبی

برای نمایش اعداد ترتیبی از نقش ord: و مفهوم ordinal-entity استفاده می‌کنیم.

```
(x / فصل  
  :ord (o / ordinal-entity :value 2))
```

فصل دوم.
فصل شماره دو.

```
(x / پروازکردن  
  :ARG0 (x2 / هواپیما)  
  :ord (o / ordinal-entity :value 1  
        :range (t / temporal-quantity :quant 10  
                :unit (x3 / سال))))
```

اولین پرواز هواپیما در ده سال گذشته.

زیرمجموعه

جملاتی که در مورد یک یا بخشی از یک مجموعه هستند در زبان فراوان یافت می‌شوند. در این موارد نامفا از نقش‌های subset و subsetof: استفاده می‌کند.

```
(x / شهیدکردن  
  :ARG1 (x2 / سرباز :quant 9  
        :subset-of (x3 / سرباز :quant 20)))
```

از بیست سرباز، نه سرباز به شهادت رسیدند.


```

(x / قبول کردن / polarity -
  :ARG1 (x2 / گل :quant 3
    :subset-of (x3 / گل :quant 5
      :mod (x6 / زیبا)))
    :subset (x4 / گل :quant 2
      :mod (x5 / آفساید))))

```

با احتساب دو گل آفساید، سه گل از پنج گل زیبا قبول نشد.

ویژگی‌های مشترک بین مجموعه و زیرمجموعه اش تنها در مجموعه دربرگیرنده وارد می‌شوند. مثل زیبا در مثال بالا. رویداد سازی subset: با استفاده از include-91 صورت می‌گیرد. پس نامفای پایین با مثال اول برابر است.

```

(x / شهید کردن
  :ARG1 (x2 / سرباز :quant 9
    :ARG1-of (i / include-91
      :ARG2 (x3 / سرباز :quant 20))))

```

از بیست سرباز، نه سرباز به شهادت رسیدند.

نامفا در مورد subset: قوانین سختی را اعمال نمی‌کند. چرا که در غیر این صورت کنترل کار از دست خارج می‌شد. برای مثال در جمله «سه تن از کارگرهای معدن» از این رابطه استفاده نمی‌شود و به سادگی به عنوان «سه کارگر» آن را وارد می‌کنیم.

موجودیت‌های نامدار و ویکی‌سازی

در نامفا هر مفهومی می‌تواند دارای نقش name: باشد. نامفا خود را به مجموعه محدودی از مقوله‌هایی چون «کشورها» و «اشخاص» محدود نمی‌کند. «کشتی‌ها»، «حیوانات»، و «رایانه‌ها» نیز می‌توانند نام داشته باشند. مقوله‌های موجودیت‌های نامدار از دسته مفاهیم انتزاعی هستند که وارد نمایش نامفای یک جمله می‌شوند. از این رو برای موازی سازی این مقوله‌ها را به زبان انگلیسی وارد نامفا می‌کنیم. در بسیاری موارد اسامی متنوعی به یک موجودیت نامدار اشاره می‌کنند. برای مثلا «مجلس»، «مجلس شورای اسلامی»، «بهارستان» و ... همگی به یک نهاد اشاره می‌کنند. چنین موجودیت‌های نامداری را با استفاده از ویکی‌پدیای فارسی و نقش wiki: یکسان‌سازی می‌کنیم:

(مجلس_شورای_شورای wiki):

این کار در مواردی که نام می‌تواند ابهام نیز داشته باشد کمک می‌کند. در همین مثال بهارستان می‌تواند به میدان بهارستان نیز اشاره داشته باشد.

(میدان_بهارستان wiki):

در صورتی که نامی صفحه ویکی‌پدیا نداشته باشد، آن را به صورت wiki: - نمایش می‌دهیم.

```
(p / person
  :wiki "براون_مارگارت"
  :name (n / name
    :op1 "مالی"
    :op2 "براون"))
```

مالی براون

```
(p / person
  :wiki "براون_مارگارت"
  :name (n / name
    :op1 "مالی"
    :op2 "براون")
  :ARG0-of (x / قتل‌کردن
    :ARG1 (x2 / رکا)))
```

مالی براون، قاتل اُرک‌ها

```
(s / ship
  :wiki "تایتانیک_آرام‌اس"
  :name (n / name
    :op1 (x / تایتانیک)))
```

تایتانیک
کشتی تایتانیک
کشتی‌ای به نام تایتانیک

```
(c / city
  :wiki "کنگ_بندر"
  :name (n / name
    :op1 (x / بندر)
    :op2 (x2 / کنگ)))
```

صالح‌آباد
شهر صالح‌آباد

نام‌ها روابط معنایی داخل موجودیت‌های نام‌دار را استخراج نمی‌کند. برای مثال در حمید ساکت رویداد «ساکت‌کردن» که «حمید» arg1: آن باشد ساخته نمی‌شود. نام‌ها اسامی خاصی که به صورت اختصاری به کار رفته‌اند را کامل نمی‌کند. اما اختصار اسامی عام را به صورت کامل نمایش می‌دهد.

```
(p / person :wiki "اصفهانی_محمد"
  :name (n / name
    :op1 (x / ممد)
    :op2 (x2 / اصفهانی)))
```

ممد اصفهانی

```

سیگار / x
    :quant 1
    :unit (x2 / نخ))

```

یه نخ سیگ
یه نخ سیگار

برای نمایش نامفای موجودیت های نامدار باید ریشه را پر کنیم. برای این کار با یکی از حالت های زیر مواجه می شویم:

الف: به طور کلی، به جز حالاتی که متن فارسی مقوله خاصی را مشخص کرده باشد، ریشه را با یکی از مقوله های استاندارد موجودیت های نامدار نامفا پر می کنیم. برای مثال person یا complany در چنین شرایطی راهی به جز تصور یک مقوله وجود ندارد:

```

p / person
    :wiki -
    :name (n / name
        :op1 "رضا"))

```

رضا

```

c / company
    :ARGO-of (x / ساختن
        :ARG1 (x2 / قطعه))

```

قطعه سازی

با این حال برای حفظ یکدستی در نامفا و جلوگیری از این امر که در برخی نامفاها person و در برخی دیگری women را داشته باشیم، در مواردی که باید مقوله موجودیت نامدار را متصور شویم از لیست یکسان سازی شده زیر استفاده می کنیم.

person, family, animal, language, nationality, ethnic-group, regional-group, religious-group, political-movement

organization, company, government-organization, military, criminal-organization, political-party, market-sector, school, university, research-institute, team, league

location, city, city-district, county, state, province, territory, country, local-region, country-region, world-region, continent; ocean, sea, lake, river, gulf, bay, strait, canal; peninsula, mountain, volcano, valley, canyon, island, desert, forest moon, planet, star, constellation

facility, airport, station, port, tunnel, bridge, road, railway-line, canal, building, theater, museum, palace, hotel, worship-place, market, sports-facility, park, zoo, amusement-park

event, incident, natural-disaster, earthquake, war, conference, game, festival

product, vehicle, ship, aircraft, aircraft-type, spaceship, car-make, work-of-art, picture, music, show, broadcast-program

publication, book, newspaper, magazine, journal

award, law, court-decision, treaty, music-key, musical-note, food-dish, writing-script, variable, program

همواره خاص ترین مقوله را انتخاب می‌کنیم.

در صورت عدم تعلق به هیچ‌یک از مقوله های بالا از thing استفاده می‌کنیم.

به مثال‌های زیر توجه کنید:

```
(a / award
  :wiki "نوبل_جایزه"
  :name (n / name
    :op1 "جایزه"
    :op2 "نوبل"))
```

جایزه نوبل

```
(g / government-organization
  :wiki "ناجا_اطلاعات_تبادل_و_تولید_فضای_پلیس"
  :name (n / name
    :op1 "پلیس"
    :op2 "سایبری")
  :mod (c / country
    :wiki "ایران"
    :name (n2 / name
      :op1 (x / ایران))))
```

پلیس سایبری ایران

```
(n / natural-object
  :wiki "سورت_باداب"
  :name (n2 / name
    :op1 "باداب"
    :op2 "سورت"))
```

باداب سورت

در صورت عدم تعلق به هیچ‌یک از مقوله‌های بالا از thing استفاده می‌کنیم.
ب: اگر متن فارسی حاوی عبارت خاص‌تری باشد از آن استفاده می‌کنیم. برای مثال:

شاعر / x

```
:wiki "فردوسی"  
:name (n / name  
      :op1 "ابولقاسم"  
      :op2 "فردوسی")  
:mod (x2 / نام‌دار))
```

ابولقاسم فردوسی شاعر نامدار
شاعر نامدار ابولقاسم فردوسی

در این مثال poet از person خاص‌تر است.

روستا / x

```
:wiki "کندوان"  
:name (n / name  
      :op1 "کندوان"))
```

روستای کندوان

در مثال بالا village از city خاص‌تر است.

دکتر / x

```
:wiki -  
:name (n / name  
      :op1 "انتزاعی"))
```

دکتر انتزاعی

عناوین تجلیلی در نامفا بخشی از نام محسوب می‌شوند:

```
(p / person
  :wiki -
  :name (n / name
    :op1 "جناب"
    :op2 "آقای"
    :op3 "انتزاعی"))
```

جناب آقای انتزاعی

در بخش بعدی «قاب‌های خاص برای نقش‌ها» به چگونگی نمایش القابی چون «رئیس جمهور» خواهیم پرداخت. نامفا در برخورد با عبارات توصیفی بسیار خونسرد عمل کرده و اطلاعات موجود را به هر گره اضافه می‌کند:

```
(g / group
  :wiki "ققنوس_انتشارات"
  :name (e / name
    :op1 "ققنوس")
  :mod (c / country
    :wiki "ایران"
    :name (h / name
      :op1 "ایران"))
  :ARGO-of (x / نشرکردن))
```

ققنوس، یک گروه انتشاراتی ایرانی است.

این عبارت را از نظر معنایی می‌توان یک group با یک نام مشخص و چند ویژگی دیگر در نظر گرفت. ج: در برخی موارد چند کلمه برای یک جایگاه در رقابت هستند. از آنجا که دو مفهوم نمی‌توانند یک جایگاه را پر کنند، از رابطه mod: استفاده می‌کنیم.

دکتر (x /

```
:wiki "پاریزی_باستانی_محمدابراهیم"  
:name (n / name  
      :op1 "باستانی"  
      :op2 "پاریزی")  
:mod (x2 / نویسنده)  
:mod (x3 / بنام))
```

نویسنده بنام، دکتر باستانی پاریزی

قاب‌های خاص برای نقش‌ها

برای نقش‌های سازمانی از قاب انگلیسی have-org-role-91 استفاده می‌کنیم:

(x / شخص :wiki "حسین_روحانی"

```
:name (n / name :op1 "حسن" :op2 "روحانی")  
:ARG0-of (h / have-org-role-91  
          :ARG1 (x2 / کشور :wiki "ایران"  
                  :name (n2 / name :op1 "ایران"))  
          :ARG2 (x3 / رئیس‌جمهور)))
```

حسن روحانی، رئیس‌جمهور ایران.

نقش‌های اصلی have-org-role-91 در جدول زیر آمده‌اند:

have-org-role-91

Arg0: شخصی که مسئولیت سازمانی را دارد.

Arg1: سازمان مورد نظر

Arg2: عنوان مسئولیت (مثل رئیس‌جمهور)

Arg3: شرح مسئولیت (به ندرت استفاده می‌شود)

برخی نقش‌های have-org-role-91 عبارتند از: سفیر، مدیر، مدیرگروه، رهبر، پادشاه، سخن‌گو، شهردار، سناتور، امپراتور، رئیس‌جمهور، امام‌جمعه، بخش‌دار، نماینده مجلس، مدیربخش و ...

برای نمایش رابطه بین دو فرد (یا دو موجودیت هم‌نوع) از قاب انگلیسی have-rel-role-91 استفاده می‌کنیم:

```
(h / have-rel-role-91
  :ARG0 (x / او)
  :ARG1 (x2 / من)
  :ARG2 (x3 / باجنق))
```

او باجنق من است.

نقش‌های اصلی have-rel-role-91 در جدول زیر آمده‌اند:

have-rel-role-91

Arg0: شخص یا موجودیت اول
 Arg1: شخص یا موجودیت دوم
 Arg2: نقش الف (اجباری)
 Arg3: نقش ب (معمولا تعریف نشده)
 Arg4: پایه رابطه (کاری، پروژه، که به ندرت استفاده می‌شود)

برخی نقش‌های have-rel-role-91 عبارتند از: پدر، مادر، خواهر، برادر، عمو، خاله، عمه، دایی، نوه، پدربزرگ، مادربزرگ، عروس، داماد، پدرخوانده، مادرزن، نامزد، دوست‌پسر، دوست‌دختر، باجنق و ...

اعداد

نام‌ها اعداد را یک‌دست می‌کند.

(x / سیب :quant 40000)

چهل هزار سیب
۴۰۰۰۰ سیب

(x / اتم :quant 1500000000)

یک و نیم میلیارد اتم
۱۵۰۰۰۰۰۰۰ اتم
یک میلیارد و نیم اتم
۵.۱ میلیارد اتم

نیاز به این یکسان سازی برای نمایش صحیح در زمان ترجمه میان زبانهایی که از سیستم عددی کوچک و زبانهایی که از سیستم عددی بزرگ استفاده می کنند مشهود است. برای مثال در نمونه بالا مقیاس این عدد در انگلیسی بیلیون است.

اعداد تقریبی

اعداد تقریبی با opN : نمایش داده می شوند.

(x / سیب
:quant (x2 / چند :op1 100))

چندصد سیب

(x / سیب
:quant (x2 / بیش از :op1 4000))

بیش از چهار هزار سیب
بیش از ۴۰۰۰ سیب

```
(x / سیب
  :quant (x2 / بین :op1 4000 :op2 5000))
```

بین چهار تا پنج هزار سیب
بین ۴۰۰۰ تا ۵۰۰۰ سیب

مقادیر و کمیت‌ها

مقادیر مشخص به وسیله نوع‌شان و آرگومان‌های unit و quant: نمایش داده می‌شوند.

```
(q / distance-quantity :quant 10
  :unit (x / کیلومتر))
```

ده کیلومتر
۱۰ کیلومتر

همانند اعداد تقریبی، مقادیر تقریبی نیز با opN: نمایش داده می‌شوند.

```
(x / حدود
  :op1 (q / distance-quantity :quant 10
    :unit (x2 / کیلومتر)))
```

حدود ده کیلومتر
۱۰ کیلومتر

نامفا عبارات کمیت‌داری مثل «دو لیتر شیر» را به صورت زیر نمایش می‌دهد:

```

(x / خریدن
  :ARG0 (x2 / دلچک)
  :ARG1 (x3 / شیر
    :quant (q / volume-quantity :quant 2
      :unit (x4 / لیتر))))

```

دلچک دو لیتر شیر خرید.

نامفا برای نمایش طول زمان و یا زمان نسبی temporal-quantity : (برای مقادیر زمانی دقیق از date-entity استفاده می شود. در بخش بعد به آن خواهیم پرداخت).

```

(t / temporal-quantity :quant 30
  :unit (x / سال))

```

سی سال

```

(x / قبل
  :op1 (x2 / حال)
  :duration (t / temporal-quantity :quant 30
    :unit (x3 / سال)))

```

طی سی سال گذشته

```

(x / قبل
  :op1 (x2 / حال)
  :quant (t / temporal-quantity :quant 30
    :unit (x3 / سال)))

```

سی سال قبل
سی سال پیش

```

    قبل (x /
      :op1 (x2 / حال)
      :quant (x3 / بیشتر
        :op1 (t / temporal-quantity :quant 30
          :unit (x4 / سال))))

```

بیش از سی سال قبل

در ترکیبات فصلی، «یا» در ریشه می‌آید:

```

    یا (x /
      :op1 (t / temporal-quantity :quant 3
        :unit (x2 / سال))
      :op2 (t2 / temporal-quantity :quant 4
        :unit (x3 / سال)))

```

سه یا چهار سال
سه چهار سال

```

    یا (x /
      :op1 (t / temporal-quantity :quant 6
        :unit (x2 / ماه))
      :op2 (t2 / temporal-quantity :quant 1
        :unit (x3 / سال)))

```

شش ماه یا یک سال

فواصل نسبی معمولا به همراه با مقادیر کمیتی می‌آیند:

سقوط کردن (x5 /

:ARG1 (x4 / هواپیما)

:location (r / relative-position

:op1 (x / شهر :name (n / name :op1 "تهران"))

:quant (d / distance-quantity :quant 50

:unit (x2 / کیلومتر))

:direction (x3 / شرق)))

هواپیما در پنجاه کیلومتری شرق تهران سقوط کرد.

سقوط هواپیما در پنجاه کیلومتری شرق تهران بود.

پنجاه کیلومتر به سمت شرق تهران سقوط هواپیما رخ داد.

برای مقادیر نامشخص از X-quant: استفاده نمی‌کنیم. quant: کفایت می‌کند.

جمع شدن (x /

:ARG0 (x2 / دلچک

:quant (x3 / تعداد

:mod (x4 / زیاد))))

تعداد زیادی دلچک جمع شده بودند.

گاهی اوقات مفهوم اصلی جمله خود مقدار است.

افزایش یافتن (x /

:ARG1 (x2 / تعداد

:quant-of (x3 / فرد)))

تعداد افراد افزایش یافت.

انواع مقادیر کمیتی:

monetary-quantity, distance-quantity, area-quantity, volume-quantity, temporal-quantity, frequency-quantity, speed-quantity, acceleration-quantity, mass-quantity, force-quantity, pressure-quantity, energy-quantity, power-quantity, voltage-quantity, charge-quantity, potential-quantity, resistance-quantity, inductance-quantity, magnetic-field-quantity, magnetic-flux-quantity, radiation-quantity, concentration-quantity, temperature-quantity, score-quantity, fuel-consumption-quantity, seismic-quantity.

```
(q / monetary-quantity :quant 20
  :unit (x / دلار
    :mod (x2 / کشور :name (n2 / name :op1 "کانادا"))))
```

بیست دلار کانادا.
C\$20

برای کمیت‌هایی که صفر مطلق ندارند و `:quant` 0 به معنا مقدار صفر آن نیست، به جای `:unit` از `:scale` استفاده کنید.

```
(q / seismic-quantity :quant 7.9
  :scale (x / ریشتر))
```

زلزله ۹.۷ ریشتری

رویداد سازی `:quant` با استفاده از قاب انگلیسی `have-quant-91` انجام می‌شود. ساختار موضوعی آن با `have-degree-91` هم‌عرض است. زمانی که تنها در مقدار چیزی توصیف نشده، مقایسه، و ساختهای برترین به کار رفته است. از این قاب استفاده می‌کنیم.

have-quant-91

Arg1: آنچه کمیتش آمده
 Arg2: کمیت (اعداد و یا صفات کمیت ساز مثل خیلی)
 Arg3: درجه کمیت (بیشتر، کمتر، برابر)
 Arg4: مرجع قیاس
 Arg5: مجموعه در برگیرنده، برترین
 Arg6: نتیجه

فروختن / x

:ARG0 (x2 / او)

:ARG1 (x3 / ماشین

:ARG1-of (h2 / have-quant-91

:ARG3 (x4 / برابر)

:ARG4 (x5 / ماشین

:ARG1-of (x6 / فروختن

:ARG0 (x7 / شخص

:ARG0-of (x8 / رقابت کردن

:ARG1 x2))))))

او به تعداد رقیبش ماشین فروخت.
 او همان مقدار ماشین فروخت که رقیبش.

فروختن / x

:ARG0 (x2 / او)

:ARG1 (x3 / ماشین

:ARG1-of (h2 / have-quant-91

:ARG3 (x4 / اکثر)

:ARG5 (x5 / ماشین

:poss (x6 / بنگاه))))))

او اکثر ماشین های بنگاه را فروخت.

داشتن / x)

:ARG0 (x2 / من)

:ARG1 (x3 / آب

:purpose (x4 / نوشیدن

:ARG0 x2)

:ARG1-of (h3 / have-quant-91

:ARG3 (x5 / کافی

:mod (x6 / سخت))

:ARG6 (x7 / ماندن

:ARG1 x3

:ARG2 (t / temporal-quantity :quant 1

:unit (x8 / هفته))

:ARG3 x2))))

به سختی آب آشامیدنی کافی داشتم که یک هفته برایم بماند.

عملگرهای ریاضی

دو مفهوم خاص product-of و sum-of می‌توانند عملگرهای ریاضی که از معنا جمله برداشت می‌شوند را پوشش دهند.

رسیدن / x)

:ARG0 (x2 / سرعت

:poss (x3 / هواپیما))

:ARG1 (p / product-of :op1 3

:op2 (x4 / سرعت

:poss (x5 / صوت))))

سرعت هواپیما به سه برابر سرعت نور رسید.

(x / دویدن

:ARG0 (x2 / شخص

:name (n2 / name :op1 "پاتریک" :op2 "ماکاو"))

:ARG1 (x3 / ماراتون)

:duration (s2 / sum-of

:op1 (t2 / temporal-quantity :quant 2

:unit (x4 / ساعت))

:op2 (t3 / temporal-quantity :quant 3

:unit (x5 / دقیقه))

:op3 (t4 / temporal-quantity :quant 38

:unit (x6 / ثانیه))))

پاتریک ماکااو ماراتون را در دو ساعت، سه دقیقه و سی و هشت ثانیه دوید.

دیگر موجودیت‌ها: تاریخ، زمان، درصد، شماره تلفن، رایانامه، URLs

این موجودیت‌ها دارای استاندارد پیش فرض و یکسان سازی هستند. برای مثلا تقویم استاندارد میلادی است و برای اینکه در یکسان سازی شماره ماه‌ها به مشکل بر نخوریم نوع تقویم را مشخص می‌کنیم:

(d / date-entity

:year 1396

:month 1

:day 13

:calendar (s / solar-hijri))

سیزده فروردین ۱۳۹۶

۱۳/۱/۱۳۹۶

(d / date-entity
:year 2012
:month 2
:day 29)

۲۹ فوریه ۲۰۱۲
۲۰۱۲/۲/۲۹

February 29, 2012
29 February 2012
2/29/2012

(d / date-entity :month 4)

فوریه

(d / date-entity
:month 1
:calendar (s / solar-hijri))

فروردین

(d / date-entity
:weekday (x / جمعه))

جمعه

(d / date-entity :year 2012 :month 2)

فوریه دوهزار و دوازده

```
(d / date-entity :year 1396 :month 2  
  :calendar (s / solar-hijri))
```

اردیبهشت نودوشش

```
(d / date-entity :month 11 :day 29  
  :weekday (x / یکشنبه)  
  :calendar (s / solar-hijri))
```

یکشنبه، ۲۹ بهمن

```
(d / date-entity :day 29)
```

بیست و نهم

```
(d / date-entity :month 2 :day 29 :time "16:30"  
  :weekday (x / چهارشنبه)  
  :timezone (z / PST))
```

چهارشنبه، ۲۹ فوریه، ساعت ۱۶:۳۰ به وقت PST

برای وقت‌های محلی و غیر استاندارد از مفاهیم استفاده می‌شود.

```
(d / date-entity :month 2 :day 29 :time "16:30"  
  :weekday (x / شنبه)  
  :calendar (s / solar-hijri)  
  :timezone (x2 / شهر :name (n / name :op1 "تهران"))))
```

شنبه، ۲۹ اردیبهشت، ساعت ۱۶:۳۰ به تهران.

(d / date-entity :time "16:45")

ساعت ۱۶:۴۵
ساعت چهار و چهل و پنج بعد از ظهر
یک ربع به پنج

(d / date-entity :year 24 :month 2 :day 29
:era (x / هیشه‌ای)
:calendar (x2 / کشور :name (n2 / name :op1 "ژاپن")))

بیست و نهم فوریه سال بیست و چهار دوره هیشه‌ای

(d / date-entity :year 1396 :quarter 4)

ربع چهارم سال ۱۳۹۶
سه‌ماه‌ی چهارم ۱۳۹۶

(d / date-entity :year 2011
:season (x / تابستان))

تابستان ۲۰۱۱

(d / date-entity :year 2011 :year2 2012
:season (x / زمستان))

زمستان ۲۰۱۲ – ۲۰۱۱

```
(d / date-entity :year 1396 :year2 1397
  :calendar (x / سال
    :mod (x2 / تحصیلی))
  :calendar (s / solar-hijri))
```

سال تحصیلی ۹۶ - ۹۷

```
(d / date-interval
  :op1 (d2 / date-entity :year 1396 :month 3 :day 8
    :calendar (s / solar-hijri))
  :op2 (d3 / date-entity :year 1396 :month 3 :day 9
    :calendar (s2 / solar-hijri)))
```

هشتم تا نهم خرداد ۱۳۹۶

```
(d / date-interval
  :op1 (d2 / date-entity :year 1395 :month 3 :day 8
    :calendar (s / solar-hijri))
  :op2 (d3 / date-entity :year 1396 :month 11 :day 9
    :calendar (s2 / solar-hijri)))
```

هشتم خرداد ۱۳۹۵ تا نهم بهمن ۱۳۹۶

```
(p / percentage-entity :value 25)
```

۲۵ درصد
بیست و پنج درصد
۲۵٪

(p / phone-number-entity :value "021-4448-1234")

۰۲۱۴۴۴۸۱۱۰۹

۰۲۱-۴۴۴۸۱۱۰۹

(e / email-address-entity :value "president@president.ir")

president@president.ir

(u / url-entity :value "www.president.ir")

www.president.ir