8 CAPÍTULO

JSPELLANDO NAS MORFOLIMPÍADAS

SOBRE A PARTICIPAÇÃO DO JSPELL NAS MORFOLIMPÍADAS

JOSÉ JOÃO ALMEIDA E ALBERTO SIMÕES

1. INTRODUÇÃO AO JSPELL

Como resultado natural da participação do JSPELL (Almeida & Pinto, 1995) nas Morfolimpíadas, aqui será apresentado o JSPELL e as suas principais preocupações, sendo tecidas algumas considerações acerca de outras facetas ligadas à análise morfológica, que pensamos que poderão ser incluídas em futuras avaliações.

A apresentação do JSPELL descreve brevemente a história da ferramenta, como são estruturados os dicionários JSPELL e respectivas regras morfológicas, bem como os seus diferentes modos de funcionamento.

Ao contrário do que o nome possa indicar, o JSPELL[1] é essencialmente um analisador morfológico, desenvolvido com base no código do ISPELL, um corrector ortográfico de código aberto.

À semelhança do que acontece com o ISPELL, no JSPELL há uma separação entre um motor de análise genérica e os dicionários e regras morfológicas, permitindo que o motor seja independente de língua, e que sejam criados dicionários para diversas línguas.

O JSPELL encaixa-se na zona das linguagens flexivas. Presentemente, existe publicamente disponível um dicionário e respectivas regras morfológicas para as línguas portuguesa e inglesa[2].

1.1 **Descrição dos dicionários JSPELL**

Um dicionário Jspell é composto por duas partes: um conjunto de regras que definem esquemas de flexão e derivação, e uma lista de triplos (*PalavrasInf*): palavras às quais são associadas a sua classificação morfológica e conjunto de esquemas de flexão a si aplicáveis:

Dic ⊒≡ *Regras* × *PalavrasInf**
Regras ⊒≡ *RegraId* × *Regra**
PalavrasInf ⊒≡ *Palavra* × *Class* × *RegraId**

A sintaxe concreta usada na definição da lista de triplos do dicionário inclui mecanismos de definição de abreviaturas para facilitar a coerência e tornar a definição mais concisa:

```
#vt = /CAT=v,T=inf,TR=t/
avaliar/#vt/DLMPRXYcu/
```

A primeira linha define uma abreviatura para a informação morfológica de verbos transitivos. A última linha define que «avaliar» é um verbo transitivo (e que herda toda a informação definida na abreviatura #vt) e que lhe podem aplicar 9 esquemas de regras diferentes (cada uma identificada por uma letra).

Para cada uma regra identificada por uma letra, o esquema contém um conjunto de definições que permitem formar (derivar) novas palavras. Por exemplo, no dicionário português, o esquema D aplicado

---

[1] *O nome JSPELL deriva de ter sido um enriquecimento do corrector ortográfico internacional ISPELL (jspell = (i+1)spell).*

[2] *O dicionário português conta actualmente (Maio de 2006) com 37 500 lemas, associados a 1 053 regras (47 esquemas). O dicionário inglês tem 44 251 lemas, associados a 90 regras (24 esquemas).*

à palavra *avaliar* permite gerar as seguintes palavras: *avaliador*, *avaliadora*, *avaliadores*, *avaliadoras*.

A associação de diferentes conjuntos de esquemas de derivação, permite um controlo fino do conjunto das palavras geradas, evitando a sobregeração.[1]

1.2 **Modos de funcionamento**

O JSPELL pode ser usado de diversas maneiras:

- como corrector ortográfico: este modo de funcionamento foi herdado do ISPELL, e alterado de modo a que novas palavras adicionadas pelo utilizador passem a incluir, sempre que possível, a informação morfológica associada;
- como interpretador: permite a interacção directa com o utilizador via linha de comando para a pesquisa de palavras (com ou sem sugestão de palavras próximas, «near-misses») e consulta das respectivas propriedades morfológicas;
- como biblioteca de programação: C, Perl (Almeida & Pinto, 1995), Prolog (Almeida, 1995).

Um analisador morfológico é, na maior parte das vezes, um bloco numa aplicação mais complexa, pelo que é importante a existência de uma API que permita aceder programaticamente ao conteúdo dos dicionários.

Em qualquer um dos modos de funcionamento, há um conjunto de opções que permitem alterar:

- os detalhes do algoritmo de pesquisa, permitindo por exemplo controlar se se pretende ou não calcular sugestões para substituir as palavras desconhecidas.
- os pormenores do algoritmo de aplicação de regras, permitindo por exemplo tentar aplicar regras a palavras às quais não estão associadas, para calcular palavras desconhecidas;
- o formato da saída, para que se possa mais facilmente colar com outras aplicações.

Segue-se um exemplo do funcionamento do JSPELL via linha de comando:

```
[ambs@eremita]$ jspell -d port -a -J -y
International Jspell Version 1.1.3

avaliação
* avaliação 0 :lex(avaliar,[CAT=nc,T=inf,TR=t,G=f,N=s,FSEM=cao])

avaliei
* avaliei 0 :lex(avaliar,[CAT=v,T=pp,TR=t,P=1,N=s])

avaliódromo
& avaliódromo 0 :avaliódromo=lex(avaliar,[CAT=nc,T=inf,TR=t,FSEM=odromo])
```

A opção inicial *-y* indica que, perante palavras desconhecidas, o JSPELL não irá produzir sugestões, mas poderá tentar utilizar regras de derivação sobre palavras existentes.

Repare-se que o primeiro caracter da linha (* ou &) descreve a aceitação ou não da palavra analisada, isto é, indica se a palavra foi ou não encontrada no dicionário.

A palavra *avaliódromo* é dada como desconhecida (&) mas é apresentada uma proposta de interpretação: *avaliódromo* pode ser obtido através da aplicação do esquema de derivação *odromo* à palavra *avaliar*

[1] *Actualmente, apenas os verbos defectivos estão a sobregerar, situação facilmente resolúvel com a adição de regras especiais para estes verbos.*

(embora este esquema não esteja na lista dos esquemas oficialmente ligados a *avaliar*).

Como analisador morfológico que é, o JSPELL tem como funcionamento principal o modo de biblioteca de programação. A sua interface com a linguagem de programação Perl está simplificada com a criação de dois módulos, jspell e jspell::dict (Simões & Almeida, 2002), que permitem a interacção simples quer com o analisador morfológico (análise de palavras, lematização, cálculo de palavras derivadas) quer com os dicionários (criação, consulta e alteração dos dicionários).

1.3 **Não queremos as palavras todas!**

Uma vez que um dos usos do dicionário português do JSPELL é dar origem aos dicionários dos correctores ortográficos do ISPELL e ASPELL, **não nos interessa ter todas as palavras** no nosso dicionário.

Por exemplo, o verbo *arvorar*, quando incluído num corrector ortográfico, torna válida a palavra *arvore*. No entanto, é fácil concluir que é muito mais provável que **arvore** seja a palavra **árvore** sem acento, do que uma forma do verbo *arvorar*.

Do mesmo modo, o uso do analisador morfológico que contenha grandes quantidades de formas muito raras, ou mesmo palavras que tenham saído do domínio activo da língua, torna-se desnecessariamente pesado e complexo.

Considere-se, por exemplo, a palavra *dele*. Esta palavra pode ser vista como 1) a contracção da preposição *de* com o pronome pessoal *ele*, ou como 2) forma do verbo *delir*. Ambas as análises são correctas e descritas na generalidade dos dicionários de língua portuguesa, embora não haja facilidade de encontrar exemplos da segunda análise em corpora. O mesmo acontece com a palavra *delira* que pode ser associada ao verbo *delirar* ou ao verbo *delir*.

Naturalmente, há necessidade de um conjunto variado de analisadores e dicionários para o português que cubram um heterogéneo conjunto de finalidades. É crucial a existência de recursos que contenham um conjunto quase exaustivo de palavras actuais e arcaicas, técnicas e eruditas. No entanto a construção de aplicações de PLN sobre este tipo de dicionário é uma tarefa muito mais complexa, já que terá de lidar com uma muito maior (falsa) ambiguidade.

No caso dos dicionários ligados ao JSPELL tem sido claro que, no geral, não queremos as palavras todas! No entanto, determinar o domínio activo da língua não é, de modo algum, simples.

Considere-se o seguinte grupo de entradas (obtidas de um conhecido dicionário português após saltar as primeiras 5000 entradas):

aliazar::grupo de lezírias circundadas de água
alibânia::tecido de algodão das Índias Orientais
álibi::justificação do réu, que consiste em p…
alibilidade::qualidade do que é alíbil
alíbil::próprio para a nutrição
álica::espécie de trigo ou de cevada de que os…
alicaído::de asa caída
alicanso::licranço
alicante::casta de uva algarvia e andaluza
alicantinador::alicantineiro
alicantina::trapaça no jogo ou nos negócios
alicantineiro::que ou aquele que faz ou vive de alicanti…
alicário::o que fabrica ou vende álica
alicatão::grande tenaz para segurar a peça que se pret…

```
alicate::ferramenta formada por duas barras ou p…
alicece::alicerce
```

Embora seja inquestionável o interesse de dispor da totalidade desta informação, parece-nos óbvio que uma parte significativa destas entradas não pertence ao domínio activo da língua portuguesa.

1.4 **Programação usando JSPELL**

Dada a grande importância do nível de programação do JSPELL, esta pequena secção exemplifica o seu uso a partir da linguagem Perl.

Um programa Perl que use o JSPELL começa por incluir o módulo JSPELL e criar um objecto que irá representar o analisador morfológico para determinada língua:

```
use jspell;
$jspell = jspell::new("port");
```

Este objecto, armazenado na variável $jspell, pode ser consultado usando vários métodos. O método rad, por exemplo, retorna uma lista de possíveis lemas para determinada palavra:

```
@a = $jspell->rad("pode");
# @a irá conter ('poder','podar')
```

Outro método bastante útil é o fea, que retorna uma lista de possíveis análises para determinada palavra.

```
@b = $jspell->fea("Porto");
```

Neste caso, a lista iria conter:

```
( {rad=>'porto', CAT=nc, G=m,  N=s},
  {rad=>'Porto', CAT=np, LA=1, SEM=cid, G=m, N=s},
  {rad=>'portar',CAT=v,  T=p,  TR=t,    P=1, N=s})
```

Além de estes e outros métodos, o módulo inclui ainda pequenas funções úteis, nomeadamente a função onethat que, dada uma lista de análises, retorna uma que esteja de acordo com uma determinada restrição:

```
$one = onethat({CAT=>"np"},$jspell->fea("Porto"))
```

Este exemplo retornaria uma análise de *Porto* cuja categoria seja nome próprio.

Outro exemplo que a seguir se apresenta faz a marcação de tempos compostos simples em textos de língua portuguesa. O programa, quando aplicado ao seguinte texto: *O João tem comido muito* e *a Joana tem comida.*, produz o seguinte texto **anotado**:

```
O João tem_comido muito e a Joana tem comida.
```

```
1  use jspell;
2  jspell_dict("port");

3  while(<>){
4    s{(\w+) (?=(\w+))}
5     { if(onethat({rad=>"ter"},                fea($1)) and
6          onethat({CAT=>"v",T=>"ppa",G=>"m",N=>"s"},fea($2)))
7       { "$1_"} else { "$1 "} }eg;
```

```
8     print;
    }
```

Notas:

linha 1,2 – inicia o JSPELL

linha 3 – itera sobre todas as linhas do texto

linha 4 – substitui cada palavra que…

linha 5 – tenha como lema o verbo *ter* e…

linha 6 – seja seguida de um particípio passado, masculino, singular…

linha 7 – pôr ela própria acrescida de uma marca (_)

2. PARTICIPAÇÃO NA AVALIAÇÃO CONJUNTA

2.1 **Avaliação: comentários gerais**

Integrado na participação na sessão de avaliação conjunta, houve naturalmente algum empenho em melhorar a ferramenta disponível. Nesse sentido, o caminho mais natural é:

- melhorar o léxico no sentido de aumentar a cobertura;
- corrigir regras mal associadas a certas palavras.

Este caminho foi apenas em parte explorado. No entanto optou-se também por uma vertente mais ambiciosa: aumentar a sofisticação da resposta, juntando novas características morfológicas ao analisador, de modo a criar diálogo acerca da sua relevância e importância no processo da análise morfológica e deste modo contribuir para uma mais rica avaliação conjunta. As principais características acrescentadas ao JSPELL foram:

- subcategorização de advérbios;
- subcategorização de nomes próprios, no sentido de indicar a utilização ou não de artigos;
- indicador de frequência (ainda em fase de protótipo).

Não se tratando de questões consensuais nem mesmo questões **bem assentes**, não foi possível incluir algumas destas características na avaliação conjunta, mas, por certo, poderão ser estudadas em futuras edições. Embora seja uma questão muito complexa e de difícil concretização, parece-nos importante que em sessões futuras haja espaço para modelos que fomentem o aumento da sofisticação das respostas.

Em todas as submissões de respostas houve sempre melhoramentos diversos, levando à criação de novas versões dos programas/dicionários. Ou seja, a avaliação conjunta contribui naturalmente para fazer evoluir as ferramentas e para evidenciar as suas fraquezas.

2.2 **Derivação**

O motor JSPELL usa regras para derivar novas palavras por flexão e derivação: inclui nos dicionários apenas lemas ou radicais das palavras, e associa a cada uma um conjunto de regras que permitam derivar novas palavras. A alternativa seria o uso de um dicionário definido por enumeração de todas as palavras existentes.

Para além do uso de flexão, acreditamos na importância do uso de regras de derivação. Parece-nos que esta abordagem permite obter resultados interessantes, nomeadamente no que respeita a:

- redução do tamanho do dicionário, já que muitas palavras não serão incluídas, por serem geradas por derivação;
- obter o mesmo radical para palavras com semântica aparentada, do que advêm vantagens

nomeadamente para aplicações de PLN que precisem de informação semântica sobre as palavras:

avaliar avaliador avaliação

- reconhecer palavras novas que não estejam incluídas no dicionário, mas que de algum modo possam ser derivadas a partir de palavras existentes, aplicando-lhes regras habituais da morfologia portuguesa:

avaliómetro avaliacionismo
avaliadamente avaliódromo

- o conhecimento da regra ou regras que deram origem a determinada palavra permite obter propriedades sintáctico-semânticas dessa palavra. Por exemplo, a palavra *rebentamento* – derivada de *rebentar* – pode reger um sintagma preposicional com *de* seguido do objecto (ou seja do elemento que rebenta). Esta situação aparece sempre que esta regra morfológica se aplica a verbos transitivos.

2.3 **Como avaliar morfologia com derivação?**

Não queremos apresentar um método de avaliação, mas apresentar um conjunto de tipos de testes que nos parecem ser exequíveis e talvez sensatos:

- testes de derivação: definir um conjunto de palavras e suas derivadas, e analisar até que ponto as ferramentas são capazes de, a partir da segunda, obter o seu lema (a primeira);
- testes de co-radicalidade: definir conjuntos de palavras para os quais os analisadores deveriam chegar a um mesmo lema:

avaliei avaliação avaliou (avaliar)

e também definir consensualmente um conjunto de palavras e respectivos lemas, para validar se a ferramenta as lematiza correctamente.

- testes de sobrevivência a palavras novas: uma das grandes vantagens do uso de derivação, como vimos, é a possibilidade de reconhecer palavras novas a partir de palavras e regras já existentes. Neste conjunto de testes pretende-se obrigar a ferramenta a «aprender» novas palavras.

O avaliómetro usado em Faro…

2.4 **Frequências**

Como vimos, o domínio activo da língua é muito mais restrito do que pode parecer ao folhear um dicionário, embora seja muito difícil definir a sua linha limítrofe.

Menos drástico do que eliminar as palavras menos activas na língua, sugere-se a inclusão de indicadores de frequência em cada lema do dicionário, permitindo que várias medidas possam ser tomadas, estática ou dinamicamente.

Com base em léxicos com informação de frequência, é possível:

- construir léxicos que sejam a restrição do léxico original a um tamanho estabelecido pelo utilizador (seleccionando os K lemas mais frequentes);
- ordenar as análises obtidas em função da sua frequência;
- eventualmente, rejeitar análises menos frequentes, quando existirem outras muito mais prováveis.

2.4.1 *Como calcular frequências?*

Normalmente, as frequências ligadas a cada palavra ou lema são expressas numa escala logarítmica.

O seu cálculo é normalmente feito com base em corpora etiquetados ou seguindo métodos como o apresentado em Rocha *et al.* (2002).

2.5 **Como avaliar a análise morfológica com atributos de frequência?**

Para a criação de padrões de referência, haverá naturalmente todo o interesse em partir dos corpora etiquetados disponíveis, como por exemplo a Floresta Sintá(c)tica descrita em Bick *et al.* (neste volume). Sempre que houver dúvidas relevantes poderá ser feita uma análise manual com base numa amostra.

A comparação entre as respostas a ser avaliadas e o padrão de referência não poderá ser uma comparação de igualdade habitual, mas sim uma comparação que calcule um erro médio obtido.

3. CONCLUSÕES

A avaliação de um conjunto de ferramentas ajuda no seu desenvolvimento e da respectiva área, pelo que qualquer exercício de avaliação é bem-vindo.

Em relação à avaliação de analisadores morfológicos, há espaço para uma avaliação mais rica, onde, por exemplo, se inclua:

- avaliação de morfologia baseada em regras de derivação;
- avaliação de indicadores de frequência de palavras;
- avaliação da capacidade de os analisadores morfológicos serem usados para resolver problemas (nesta modalidade seriam colocados problemas de processamento de linguagem natural envolvendo morfologia, que os concorrentes teriam de resolver usando as suas ferramentas).