

Língua Natural 2020/21 | Mini-Projeto nº 2

Grupo 52 | 89413 Andreia Matos (50%) | 89533 Raquel Santos (50%)

Justificação % - dividimos a parte inicial equitativamente e fizemos o restante em simultâneo.

1. Introdução

Este projeto consiste na criação de um conversor temporal utilizando apenas transdutores, de modo a converter uma hora no formato textual à sua representação numérica e vice-versa.

2. Descrição da Solução e Comentários

Inicialmente, desenvolvemos quatro transdutores: **horas** - conversão de horas textuais nos dois dígitos que as representam, **minutos** - conversão de minutos textuais nos dois dígitos que os representam, **meias** - conversão do termo *meia* para o equivalente *trinta* e **quartos** - conversão de termos com *quarto(s)* para as expressões equivalentes. Nos transdutores **horas** e **minutos** optámos pela utilização de pesos (com o valor -1) a fim de priorizar a escrita da hora com os termos *hora(s)* e *minuto(s)*, uniformizando todos os valores.

Utilizando como *sources* os transdutores já referidos criámos alguns mais complexos, recorrendo também a dois outros transdutores adicionais que considerámos necessários: **e** - conversão do *e* para o símbolo “:” e **lazye** - conversão do vazio para :00, para os casos em que não são explicitados os minutos.

O primeiro, **text2num**, converte expressões do tipo *X horas e Y minutos* na forma *hh:mm*, o que conseguimos com a concatenação do **horas** com **e** e na subsequente concatenação com **minutos**, utilizando a função *fstconcat*.

Segue-se o transdutor **lazy2num** que, para além das expressões aceites pelo anterior, aceita ainda expressões do tipo *X horas*, que converte na forma *hh:00*. Fizemos esta alteração incorporando a concatenação do **lazye** que cobre a situação extra em causa.

O transdutor **rich2text** converte expressões que utilizam os termos *meia(s)* e *quarto(s)* nas expressões escritas equivalentes. Optámos por recorrer à função *fstproject* para reescrever as horas dadas, seguindo-se de uma concatenação destas, *fstconcat*, com a união, *fstunion*, do **meias** e **quartos**, considerando assim estes casos em particular.

O **rich2num** converte uma qualquer expressão de horas para a sua forma numérica *hh:mm*, incluindo os casos com *meia(s)/quarto(s)*. Para tal, fizemos uma composição, com *fstcompose*, entre os dois últimos transdutores, previamente *sorted*, utilizando como input o **rich2text**, seguindo o modelo do **lazy2num**. Por fim fizemos uma união com o **lazy2num** para incluir todos os casos restantes.

Por fim, o transdutor **num2text** converte uma expressão numérica na sua forma textual, com os termos *hora(s)* e *minutos(s)*. Para isso é usada a função *fstinvert* com o transdutor anterior de modo a trocar as fitas de input e output e efetuar a conversão contrária.

3. Bibliografia

<http://www.openfst.org/>