# CRF+LG: uma abordagem híbrida para o reconhecimento de entidades nomeadas em português

Juliana Pinheiro Campos Pirovani

Juliana Pinheiro Campos Pirovani

# CRF+LG: Uma Abordagem Híbrida para o Reconhecimento de Entidades Nomeadas em Português

**Vitória, ES**

**2019**

Juliana Pinheiro Campos Pirovani

# CRF+LG: Uma Abordagem Híbrida para o Reconhecimento de Entidades Nomeadas em Português

Tese apresentada ao Programa de Pós-Graduação em Informática (PPGI) da Universidade Federal do Espírito Santo (UFES) como requisito parcial para obtenção do Grau de Doutor em Ciência da Computação.

Centro Tecnológico

Programa de Pós-Graduação em Informática

Universidade Federal do Espírito Santo – UFES

Orientador: Prof. Dr. Elias Silva de Oliveira

Vitória, ES

2019

Ficha catalográfica disponibilizada pelo Sistema Integrado de Bibliotecas - SIBI/UFES e elaborada pelo autor

---

Pirovani, Juliana Pinheiro Campos, 1984-
P671c CRF+LG: Uma Abordagem Híbrida para o Reconhecimento de Entidades Nomeadas em Português / Juliana Pinheiro Campos Pirovani. - 2019.
114 f. : il.

Orientador: Elias Silva de Oliveira.
Tese (Doutorado em Informática) - Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Processamento de linguagem natural (Computação). 2. Processamento de textos (Computação). 3. Reconhecimento de Entidades Nomeadas. 4. Campos Aleatórios Condicionais. 5. Gramáticas Locais. I. Oliveira, Elias Silva de. II. Universidade Federal do Espírito Santo. Centro Tecnológico. III. Título.

CDU: 004

---

# *CRF+LG: Uma Abordagem Híbrida para o Reconhecimento de Entidades Nomeadas em Português*

***Juliana Pinheiro Campos Pirovani***

Tese submetida ao Programa de Pós-Graduação em Informática da Universidade Federal do Espírito Santo como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação. Aprovada em 07 de fevereiro de 2019 por:

**Prof. Dr. Elias Silva de Oliveira** (Orientador)
UFES/ES

**Profª. Drª. Claudine Santos Badue Gonçalves** (Examinador Interno)
UFES/ES

**Prof. Dr. Patrick Marques Ciarelli** (Examinador Externo)
UFES/ES

**Profª. Drª. Priscila Machado Vieira Lima** (Examinador Externo)
UFRJ/RJ

**Prof. Dr. Éric Laporte** (Examinador Externo)
Université Paris-Est Marne-la-Vallée, France

**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO**

**Vitória-ES, 07 de fevereiro de 2019.**

*Aos meus pais, Francisco e Selma.*
*E ao meu amado esposo Victor.*

# Agradecimentos

Agradeço a Deus, por guiar os meus passos e renovar minhas forças a cada dia.

Ao meu amado esposo Victor, pelo amor, incentivo, paciência e compreensão. Você é um grande exemplo pra mim. Obrigada por estar sempre ao meu lado e pelas palavras carinhosas nos momentos mais difíceis.

Agradeço aos meus pais, Francisco e Selma, pelo amor e apoio incondicional. Obrigada pelas lições de vida e fé que me tornaram uma mulher forte e persistente. Às minhas irmãs, cunhados e sobrinhos, pelo carinho, apoio e tantos momentos de alegria.

Ao José Antônio e a Aulenir, pelo carinho, atenção e incentivo.

A todos os tios e tias em Vitória, por me acolherem com tanto carinho nesses últimos anos. Em especial, agradeço a tia Neusa e ao tio Carlinhos, por me receberem toda semana como filha em sua casa nos primeiros anos dessa jornada.

Agradeço ao meu orientador, Professor Elias, pelo conhecimento compartilhado, pela atenção, por confiar em mim e pelas várias oportunidades que me fizeram crescer na vida acadêmica.

Aos Membros da Banca Examinadora pela participação na defesa deste trabalho e pelas valiosas contribuições.

Aos professores, colaboradores e colegas do Programa de Pós-Graduação em Informática (PPGI) da Universidade Federal do Espírito Santo (UFES), que sempre estiveram dispostos a ajudar. Em especial, agradeço aos colegas do LCAD, que muito me ajudaram no desenvolvimento deste trabalho.

Ao Departamento de Computação (DCOMP) do Centro de Ciências Exatas, Naturais e da Saúde (CCENS) da UFES, pelo apoio prestado na realização deste trabalho. Aos colegas que me acompanharam nessa jornada e estiveram comigo nas muitas viagens a Vitória. Um agradecimento especial ao Professor Edmar, amigo e companheiro de estudo e trabalho.

A todos que contribuíram, direta ou indiretamente, para conclusão dessa etapa da minha vida. Cheguei até aqui porque tive a oportunidade de conhecer, conviver e aprender com pessoas muito especiais.

Muito obrigada!

*"Nas grandes batalhas da vida,*
*o primeiro passo para a vitória*
*é o desejo de vencer."*
*(Mahatma Gandhi )*

# Resumo

O Reconhecimento de Entidades Nomeadas tem como objetivo identificar e classificar automaticamente entidades como pessoas, locais e organizações e é uma tarefa muito importante em Extração de Informação. As abordagens utilizadas no desenvolvimento de sistemas de Reconhecimento de Entidades Nomeadas são: linguística, aprendizado de máquina ou híbrida. Este trabalho propõe o uso de uma abordagem híbrida, denominada CRF+LG, para o Reconhecimento de Entidades Nomeadas em textos em Português buscando explorar as vantagens das abordagens linguística e de aprendizado de máquina.

A abordagem proposta usa Campos Aleatórios Condicionais (*Conditional Random Fields - CRF*) considerando a classificação obtida previamente por uma Gramática Local (*Local Grammar - LG*) como uma característica adicional. Campos Aleatórios Condicionais é um método probabilístico para predição estruturada. Gramáticas locais são regras construídas manualmente para identificar expressões em um texto. O objetivo foi estudar essa forma de incluir a *expertise* humana (Gramática Local) na abordagem de aprendizado de máquina Campos Aleatórios Condicionais e analisar como ela pode contribuir para o desempenho dessa abordagem.

Para alcançar esse objetivo, uma Gramática Local foi construída para reconhecer as 10 categorias de entidades nomeadas do HAREM, um evento de avaliação conjunta para o Reconhecimento de Entidades Nomeadas em Português. Inicialmente, as Coleções Douradas do Primeiro e Segundo HAREM, consideradas bases de referência para essa tarefa em Português, foram utilizadas como bases de treino e teste, respectivamente, para avaliação do CRF+LG. Posteriormente, a abordagem proposta foi avaliada em outras duas bases de dados.

Os resultados obtidos superam os resultados de sistemas reportados na literatura que foram avaliados em condições equivalentes. Esse ganho foi de aproximadamente 8 pontos percentuais em Medida-F em relação a um sistema que também usou CRF e de 2 pontos percentuais em relação a um sistema que usou Redes Neurais. Alguns sistemas que usaram Redes Neurais apresentaram resultados superiores, mas usando *corpora* massivo para aprendizado não supervisionado de características, o que não foi utilizado neste trabalho.

A Gramática Local construída pode ser utilizada individualmente quando não há *corpus* de treino disponível e em conjunto com outras técnicas de aprendizado de máquina para melhorar o seu desempenho. Também foram analisados os limites (inferior e superior) da abordagem proposta. O limite inferior indica o desempenho mínimo e o limite superior

indica o ganho máximo que pode ser obtido para a tarefa em questão ao usar esta abordagem.

**Palavras-chaves**: Reconhecimento de Entidades Nomeadas. Campos Aleatórios Condicionais. Gramáticas Locais.

# Abstract

Named Entity Recognition involves automatically identifying and classifying entities such as persons, places, and organizations, and it is a very important task in Information Extraction. Named Entity Recognition systems can be developed using the following approaches: linguistics, machine learning or hybrid. This work proposes the use of a hybrid approach, called CRF+LG, for Named Entity Recognition in Portuguese texts in order to explore the advantages of both linguistics and machine learning approaches.

The proposed approach uses Conditional Random Fields (CRF) considering the term classification obtained by a Local Grammar (LG) as an additional informed feature. Conditional Random Fields is a probabilistic method for structured prediction. Local grammars are handmade rules to identify expressions within the text. The aim was to study this way of including the human expertise (Local Grammar) in the machine learning Conditional Random Fields approach and to analyze how it can contribute to the performance of this approach.

To achieve this aim, a Local Grammar was built to recognize the 10 named entities categories of HAREM, a joint assessment for the Named Entity Recognition in Portuguese. Initially, the Golden Collection of the First and Second HAREM, considered as a reference for Named Entity Recognition systems in Portuguese, were used as training and test sets, respectively, for evaluation of the CRF+LG. After that, the proposed approach was evaluated in two other datasets.

The results obtained outperform the results of systems reported in the literature that were evaluated under equivalent conditions. This gain was approximately 8 percentage points in F-measure in comparison to a system that also used CRF and 2 points in comparison to a system that used Neural Networks. Some systems that used Neural Networks presented superior results, but using massive *corpora* for unsupervised learning of features, which was not the case of this work.

The Local Grammar built can be used individually when there is no training set available and in conjunction with other machine learning techniques to improve its performance. We also analyzed the boundaries (lower bound and upper bound) of the proposed approach. The lower bound indicates the minimum performance and the upper bound indicates the maximum gain that we can achieve for the task in question when using this approach.

**Keywords**: Named Entity Recognition. Conditional Random Fields. Local Grammars.

# Lista de figuras

# Lista de tabelas

# Lista de abreviaturas e siglas

| | |
|---|---|
| NER | Named Entity Recognition |
| IE | Information Extraction |
| NLP | Natural Language Processing |
| NE | Named Entity |
| MUC | Message Understanding Conference |
| CoNLL | Conference on Natural Language Learning |
| HAREM | Avaliação de Reconhecimento de Entidades Mencionadas |
| CD | Coleção Dourada |
| CRF | Conditional Random Fields |
| LG | Local Grammar |
| ACE | Automatic Content Extraction |
| TAC | Text Analysis Conference |
| EDL | Entity Discovery and Linking |
| W-NUT | Workshop on Noisy User-Generated Text |
| HMM | Hidden Markov Model |
| MEMM | Maximum Entropy Markov Model |
| LGG | Local Grammar Graph |
| ETL | Entropy Guided Transformation Learning |
| DNN | Deep Neural Network |
| SVM | Support Vector Machine |
| TBL | Transformation Based Learning |
| LSTM | Long Short-Term Memory |
| Bi-LSTM | Bidirectional Long Short-Term Memory |
| POS | Part of speech |

# Sumário

# 1 Introdução

Uma grande quantidade de informação disponível atualmente encontra-se em textos de escrita livre, ou seja, não estruturados. O tratamento desses textos é relevante para muitas aplicações que buscam informações específicas a partir deles. A Extração de Informação (*Information Extraction - IE*) é uma área que busca obter textos estruturados, facilitando a identificação dessas informações.

O Reconhecimento de Entidades Nomeadas (*Named Entity Recognition - NER*) é uma tarefa importante para as áreas de IE e Processamento de Linguagem Natural (*Natural Language Processing - NLP*). Essa tarefa tem como objetivo identificar e classificar entidades automaticamente em textos de escrita livre. As entidades identificadas correspondem a nomes de pessoas, lugares, organizações, obras, dentre outras consideradas relevantes em domínios específicos como proteínas (WEI et al., 2013) e bacias sedimentares (AMARAL, 2017).

Nomes de pessoas, por exemplo, aparecem com frequência em textos e podem ser considerados uma fonte de informação essencial. Eles podem ser úteis para identificar a quem o texto se refere e compreender melhor o assunto do texto, possibilitando sua classificação. Muitas aplicações buscam informações sobre indivíduos e seus relacionamentos e as pessoas estão cada vez mais interessadas em saber o que os outros falam delas e onde elas são citadas.

Segundo Jiang (2012), NER é uma das tarefas fundamentais da IE pois, além de ter várias aplicações, outras tarefas como a extração de relações e eventos, sistemas de pergunta e resposta e busca orientada a entidades dependem dela como um passo do pré-processamento. Um exemplo é a aplicação apresentada em Pirovani, Spalenza e Oliveira (2017) que gera questões automaticamente a partir das entidades nomeadas (*Named Entities - NEs*) previamente extraídas e classificadas em textos didáticos.

O NER não é uma tarefa simples. Várias categorias de entidades nomeadas são escritas de forma semelhante e aparecem em contextos semelhantes. Por exemplo, nomes de pessoas e lugares começam com uma letra maiúscula, assim como expressões temporais e valores contém números. Além disso, a mesma NE pode ser classificada em categorias diferentes dependendo do contexto em que aparece. A NE Washington pode se referir a uma pessoa em um contexto e a um local em outro.

Grandes listas de NEs (*gazetteers*) são utilizadas por vários sistemas de NER, o que nem sempre corresponde a um grande ganho em desempenho como apresentado por Mikheev, Moens e Grover (1999). Isso ocorre porque nomes de pessoas, organizações e outras NEs fazem parte de uma classe de palavras conhecida como *open word class*

que é uma classe com grande número de palavras e que cresce a cada dia (MANNING; SCHÜTZE, 1999). Pode-se dizer que a identificação de NEs depende não só do idioma, mas também do *corpus* e domínio considerado.

A sexta *Message Understanding Conference* (MUC-6) (GRISHMAN; SUNDHEIM, 1996), um evento para promover e avaliar novos métodos de extração de informação, adicionou a tarefa de NER pela primeira vez para o Inglês, em 1995. A partir daí, a tarefa de NER foi avaliada em outros eventos como o programa ACE (*Automatic Content Extraction*) (DODDINGTON et al., 2004), a tarefa compartilhada da CoNLL (*Conference on Natural Language Learning*) (SANG; MEULDER, 2003), o HAREM (Avaliação de Reconhecimento de Entidades Mencionadas)(SANTOS; CARDOSO, 2007; MOTA; SANTOS, 2008), dentre outros eventos como a TAC (*Text Analysis Conference*) (NIST, 2018) que avalia a tarefa desde 2009.

O HAREM (SANTOS; CARDOSO, 2007; MOTA; SANTOS, 2008), um evento organizado pela Linguateca (Linguateca, 2018), foi um grande incentivo ao desenvolvimento de sistemas de NER para a língua Portuguesa. O HAREM adota uma classificação de 10 categorias de NEs: Abstração, Acontecimento, Coisa, Local, Obra, Organização, Pessoa, Tempo, Valor e Outro. Os *corpora* anotados usados no Primeiro e Segundo HAREM, conhecidos como Coleção Dourada (CD), são referência para trabalhos recentes em NER.

Sistemas de NER podem ser desenvolvidos utilizando as seguintes abordagens: linguística, aprendizado de máquina ou híbrida. Na abordagem linguística, regras onde NEs podem aparecer são identificadas e construídas manualmente, ou seja, as regras permitem a detecção de NEs. No aprendizado de máquina, sistemas aprendem a identificar e classificar NEs a partir de um *corpus* de treinamento. A abordagem híbrida combina as duas abordagens anteriores.

Descrever as regras da abordagem linguística requer *expertise* humana e esforços manuais. Contudo, sistemas baseados em aprendizado de máquina dependem de grande quantidade de *corpora* anotado para treinamento e não levam em consideração a *expertise* humana que poderia capturar regras que não aparecem nos exemplos desse *corpora*.

Este trabalho buscou explorar o potencial das abordagens aprendizado de máquina e linguística construindo um sistema híbrido para o NER em Português. A estratégia apresentada, CRF+LG, utiliza a técnica de aprendizado de máquina Campos Aleatórios Condicionais (*Conditional Random Fields - CRF*) (LAFFERTY; MCCALLUM; PEREIRA, 2001) e Gramáticas Locais (*Local Grammars - LGs*) (GROSS, 1997) que são uma forma de representar as regras contextuais da abordagem linguística.

## 1.1 Motivação

Os primeiros sistemas de NER utilizavam a abordagem linguística. Em 2004, a abordagem linguística foi considerada por Friburger e Maurel (2004) como a abordagem mais utilizada pelos sistemas de NER. Já em 2012, Jiang (2012) afirmou que as soluções mais recentes para NER utilizavam métodos de aprendizado de máquina estatísticos tais como Modelos de Markov Ocultos (*Hidden Markov Model - HMM*), Modelos de Markov de Entropia Máxima (*Maximum Entropy Markov Models - MEMM*) e Campos Aleatórios Condicionais (*Conditional Random Fields - CRF*). A partir de 2015, começaram a surgir soluções com desempenho competitivo para o NER em Português (SANTOS; GUIMARAES, 2015) usando Redes Neurais Profundas (*Deep Neural Network - DNN*).

Apesar disso, Amaral et al. (2014) comparou quatro ferramentas com o objetivo de extrair NEs em Português. Das ferramentas avaliadas, duas são baseadas em uma abordagem linguística, LanguageTasks (Language Tasks, 2019) e PALAVRAS (BICK, 2000), e duas são baseadas em aprendizado de máquina, NERP-CRF (AMARAL, 2013; AMARAL; VIEIRA, 2014) e FreeLing (FreeLing, 2018). Os experimentos foram realizados no *corpus* do Segundo HAREM e foram usadas as categorias Pessoa, Local e Organização existentes em todos os sistemas comparados. Os resultados indicam vantagens para as diferentes ferramentas no reconhecimento de diferentes classes de NEs. Por exemplo, LanguageTasks e PALAVRAS obtiveram melhor desempenho para classe Pessoa, o que pode indicar que a abordagem usada nessas ferramentas (linguística) seja mais apropriada para essa categoria.

Esses resultados sugerem que diferentes abordagens podem ser adequadas a diferentes categorias do NER e justificam um estudo mais profundo do potencial da abordagem híbrida para o NER em Português, já que essa abordagem combina as vantagens das abordagens linguística e de aprendizado de máquina. Apesar da abordagem linguística ser simples, as regras são escritas por um humano e podem capturar mais evidências indicando a presença de entidades nomeadas que poderiam não ser notadas por outras estratégias (ZHOU; SU, 2002). Com o aprendizado de máquina, outras regras podem ser aprendidas a partir do conjunto de textos usado no treino. Ou seja, as duas abordagens se complementam.

Além disso, a pesquisa em NER para o Português ainda é escassa comparada a outros idiomas como Inglês e Francês que tem mais recursos para NLP (AMARAL et al., 2014). De fato, "uma boa parte da pesquisa NER é dedicada ao estudo do Inglês devido à sua importância como uma língua dominante usada internacionalmente" (SHAALAN, 2014).

Os valores de desempenho obtidos pelos sistemas que usam as bases de referência do HAREM ainda são mais baixos comparados aos demais. Isso acontece não só devido a essa

escassez de recursos para NLP, mas também porque o HAREM apresenta uma tarefa mais exigente (SANTOS; CARDOSO, 2007). De acordo com Mota e Santos (2008), o HAREM difere dos outros eventos semelhantes em dois aspectos: a classificação de uma NE depende exclusivamente do seu uso em contexto (não se prende a nenhum dos atributos a que possa estar associada em dicionários ou ontologias) e mais de uma classificação pode ser atribuída a uma NE. O HAREM também possui um *corpus* de textos mais variado e classifica mais categorias de NE. O HAREM classifica 10 categorias que possuem no total 34 tipos e 17 subtipos, enquanto a MUC (GRISHMAN; SUNDHEIM, 1996), por exemplo, classifica 5 (Pessoa, Local, Organização, Tempo e Quantidade) e a CoNLL (SANG; MEULDER, 2003) classifica apenas 4 (Pessoa, Local, Organização e Misc representando diversas NEs).

## 1.2 Hipótese da Pesquisa

A **hipótese** de pesquisa deste trabalho pode ser definida da seguinte forma: é possível melhorar o desempenho de sistemas de NER em textos escritos em Português usando uma abordagem híbrida que insira o conhecimento humano capturado pela abordagem linguística durante o aprendizado de máquina.

Durante a revisão de literatura, não foi encontrado um sistema que faça a combinação das abordagens linguística e de aprendizado de máquina para o NER em Português da forma proposta neste trabalho. As regras da abordagem linguística identificadas e construídas durante este trabalho podem ser utilizadas posteriormente com outras técnicas de aprendizado de máquina.

## 1.3 Objetivos

O **objetivo principal** deste trabalho é propor, implementar e avaliar uma abordagem híbrida para o Reconhecimento de Entidades Nomeadas em Português capaz de classificar as 10 categorias de NEs do HAREM. Tipos e subtipos também foram classificados no HAREM, mas apenas as 10 categorias foram utilizadas neste trabalho porque os sistemas mais recentes classificam só as categorias.

A idéia é estudar uma forma de melhorar o desempenho de sistemas de NER que utilizam a abordagem de aprendizado de máquina dependendo de menos *corpora* para treino e também obter uma forma de realizar o NER usando a abordagem linguística quando não há *corpus* de treino disponível.

Para alcançar esse objetivo, foi construída uma Gramática Local (*Local Grammar - LG*) para reconhecer as 10 categorias de NEs do HAREM. Campos Aleatórios Condicionais (*Conditional Random Fields - CRF*) também foi utilizado para reconhecer essas 10 categorias e, neste trabalho, é mostrado como combinar CRF e LG em uma abordagem

híbrida para o NER em Português. Para avaliar o desempenho da abordagem híbrida proposta, comparações com abordagens apresentadas na literatura foram realizadas.

## 1.4 Contribuições e Publicações

Além de contribuir com a apresentação de uma abordagem híbrida para o NER em Português, outras contribuições deste trabalho foram:

- Construção de uma LG para a classificação de 10 categorias de NEs (Abstração, Acontecimento, Coisa, Local, Obra, Organização, Pessoa, Tempo, Valor e Outro) em Português. Essa LG pode ser utilizada individualmente e em conjunto com outras técnicas. Além disso, ela pode ser adaptada para outros domínios.
- Identificação e apresentação das relações da teoria de conjuntos (Tabela 8) que podem ser observadas através da ferramenta de comparação de concordâncias do Unitex. Essa tabela possibilita tomar algumas decisões ao analisar comparações de concordâncias, auxiliando o humano na composição de LGs e reduzindo o seu esforço manual.
- Apresentação das inconsistências identificadas entre as CDs do HAREM e remoção das principais inconsistências.
- Anotação de nomes de pessoas em textos do jornal A Tribuna. O *corpus* aTribuna construído contém 100 documentos anotados com 2714 nomes de pessoas. A ideia é usar os textos do jornal A Tribuna para construir futuramente uma grande base de referência para o Português incluindo outras categorias, tipos e subtipos de NEs relevantes.
- Investigação do impacto de algumas decisões de pré-processamento no desempenho do CRF, verificando que essas decisões podem afetar muito o desempenho final.
- Estudo dos limites (*lower bound e upper bound*) da abordagem proposta considerando resultados aleatórios e corretos vindos de um outro classificador como uma característica adicional.

Os seguintes artigos foram publicados durante o desenvolvimento deste trabalho:

- PIROVANI, J. P. C.; OLIVEIRA, E. de. Portuguese Named Entity Recognition using Conditional Random Fields and Local Grammars. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). Miyazaki, Japan. European Language Resources Association (ELRA), 2018. Qualis: A1.

- PIROVANI, J. P. C.; OLIVEIRA, E.; LAPORTE, E. Concordance Comparison as a Means of Assembling Local Grammars. In: Villavicencio A. et al. (eds) Computational Processing of the Portuguese Language. PROPOR 2018. Lecture Notes in Computer Science, vol 11122. Canela, RS: Springer, Cham, 2018. p. 57–65. Qualis: B3.

- PIROVANI, J. P. C.; NOGUEIRA, M.; OLIVEIRA, E. Indexing Names of Persons in a Large Dataset of a Newspaper. In: Villavicencio A. et al. (eds) Computational Processing of the Portuguese Language. PROPOR 2018. Lecture Notes in Computer Science, vol 11122. Canela, RS: Springer, Cham, 2018. p. 147–155. Qualis: B3.

- PIROVANI, J. P. C.; OLIVEIRA, E. de. CRF+LG: A Hybrid Approach for the Portuguese Named Entity Recognition. In: Abraham A., Muhuri P., Muda A., Gandhi N.(eds) Intelligent Systems Design and Applications (ISDA 2017). Advances in Intelligent Systems and Computing. Delhi, India: Springer, Cham, 2017. v. 736, p. 102–113. Qualis: B1.

- PIROVANI, J. P. C.; SPALENZA, M. A.; OLIVEIRA, E. Geração Automática de Questões a partir do Reconhecimento de Entidades Nomeadas em Textos Didáticos. In: XXVIII Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE 2017). 2017. v. 28, n. 1, p. 1147–1156. Disponível em: <http://dx.doi.org/10.5753/cbie.sbie.2017.1147>. Qualis: B1.

- PICOLI, L.; PIROVANI, J.; OLIVEIRA, E.; LAPORTE, E. Uso de uma Ferramenta de Processamento de Linguagem Natural como Auxílio à Coleta de Exemplos para o Estudo de Propriedades Sintático-Semânticas de Verbos. Linguamática, v. 7, n. 2, p. 35–44, 2015. Qualis: B5.

- PIROVANI, J. P. C.; OLIVEIRA, E. de. Extração de Nomes de Pessoas em Textos em Português: uma Abordagem Usando Gramáticas Locais. In: Computer on the Beach 2015. Florianópolis, SC: SBC, 2015. p. 1–10. Qualis: B4.

## 1.5 Organização do Trabalho

Esta Tese está organizada em 8 capítulos, sendo este o primeiro.

No Capítulo 2, a fundamentação teórica do trabalho é apresentada. Os conceitos básicos da área de pesquisa são definidos e as técnicas utilizadas neste trabalho (LG e CRF) são apresentadas.

O Capítulo 3 apresenta os trabalhos correlatos.

As bases de dados utilizadas e a metodologia de avaliação são apresentadas no Capítulo 4.

Em seguida, o Capítulo 5 faz uma descrição da abordagem proposta: CRF+LG. É apresentada, portanto, uma visão geral do sistema e todos os seus componentes.

Um estudo sobre o uso da ferramenta de comparação de concordâncias como auxílio à composição manual de LGs é apresentado no Capítulo 6.

O Capítulo 7 apresenta os experimentos realizados, bem como os resultados da avaliação de desempenho e a análise e discussão desses resultados.

Por último, no Capítulo 8, são apresentadas as conclusões do trabalho incluindo as sugestões para trabalhos futuros.

# 2 Fundamentos Teóricos

Neste capítulo são apresentados os fundamentos teóricos necessários para a compreensão deste trabalho. Inicialmente, a tarefa de Reconhecimento de Entidades Nomeadas é definida e um breve histórico das principais conferências de avaliação na área é apresentado. Em seguida, as duas técnicas utilizadas neste trabalho são apresentadas: Gramáticas Locais e Campos Aleatórios Condicionais.

## 2.1 Reconhecimento de Entidades Nomeadas (NER)

Segundo Jiang (2012), uma entidade nomeada é uma sequência de palavras que designa alguma entidade do mundo real. O termo em inglês *Named Entities* foi traduzido como Entidades Mencionadas (EM) no Primeiro HAREM representando entidades com nome próprio. A organização do Primeiro e Segundo HAREM afirma que uma EM deve conter pelo menos uma letra maiúscula e/ou algarismos (MOTA; SANTOS, 2008), embora apresentem várias exceções a essa regra em Santos e Cardoso (2007) como nomes de meses e formas de tratamento.

A tarefa de NER consiste em processar automaticamente um *corpus* de textos de escrita livre, identificando e anotando as NE. As anotações contém a classificação da NE a partir de um conjunto de categorias predefinidas. Ou seja, o NER produz textos estruturados como apresentado no exemplo da Figura 1. Nesse exemplo, as NEs *Joaninha Sampaio e Melo* e *Lourinhã* foram identificadas como nomes de pessoa e de local, respectivamente. A anotação utilizada segue a formatação proposta no HAREM.

Pois, o resto, tinha grandes amigas sempre. Constante, até era, mais até que as minhas primas... tinha as minhas primas, e tinha a Joaninha Sampaio e Melo, que era filha da maior amiga da minha mãe, da Lourinhã.

Pois, o resto, tinha grandes amigas sempre. Constante, até era, mais até que as minhas primas... tinha as minhas primas, e tinha a <EM **ID**="H2-Efin-201" **CATEG**="PESSOA" **TIPO**="INDIVIDUAL">Joaninha Sampaio e Melo</EM>, que era filha da maior amiga da minha mãe, da<EM **ID**="H2-Efin-202" **CATEG**="LOCAL" **TIPO**="HUMANO" **SUB-TIPO**="DIVISAO">Lourinhã</EM>.

Figura 1 – Exemplo de NER em um trecho da CD do Segundo HAREM

NEs aparecem com frequência nos textos. A quantidade de nomes próprios em jornais, por exemplo, é de aproximadamente 10% e sua qualidade informativa os torna relevantes para várias aplicações (FRIBURGER; MAUREL, 2004). NEs foram utilizadas por Friburger, Maurel e Giacometti (2002) no cálculo de similaridade, apresentando melhores resultados no processo de clusterização. Uma outra aplicação é a construção de índice de nomes apresentada em Pirovani, Nogueira e Oliveira (2018).

Extrair NEs é um grande desafio. O problema pode se tornar mais complexo dependendo do domínio dos textos considerados. Esse é o caso dos textos nas redes sociais, onde não há um padrão estrito para se referir às NEs.

A Figura 2 apresenta algumas dificuldades no reconhecimento da NE Pessoa. Nomes de pessoas podem aparecer de diversas formas: a) completos: onde todas as partes (primeiro nome, nome do meio e último nome) são apresentadas; b) parcial: onde somente uma parte do nome é apresentada ou uma combinação dessas partes que podem incluir abreviações; c) referência: quando ao invés de usar o nome real da pessoa, um apelido é usado.

a) E tinha a **Joaninha Sampaio e Melo**,...
b) Hoje foi um ótimo dia para **Mercedes**.
b)... e surgem os heterónimos **H. M. F. Lecher** e ...
c) **Fenômeno** não participava de um coletivo...

Figura 2 – Exemplos de NEs da categoria Pessoa

Observe que, no primeiro exemplo, a conjunção *e* aparece como parte do nome *Joaninha Sampaio e Melo*, o que não é algo comum. Além disso, *Mercedes* pode ser um nome de pessoa ou organização no segundo exemplo.

## 2.2 Conferências de Avaliação Conjunta para o NER

A sexta *Message Understanding Conference* (MUC-6) (GRISHMAN; SUNDHEIM, 1996) formalizou a tarefa de NER em 1995. Na MUC, a *tag* ENAMEX foi utilizada para anotação de nomes de pessoas, lugares e organizações; TIMEX para expressões temporais como data e hora; e NUMEX para expressões numéricas como valores monetários e porcentagens. A MUC-6 realizou uma avaliação conjunta para o NER em inglês. Nesse tipo de avaliação, vários sistemas participantes realizam o NER em um mesmo *corpus* de textos e são avaliados utilizando as mesmas métricas. Assim, é possível comparar o desempenho desses sistemas. Na MUC-7 (MUC-7, 2016) foi inserida a tarefa de identificação de relações com organizações (*employee_of, product_of, location_of*).

Desde a MUC-6, a pesquisa em NER foi constante e a avaliação conjunta dessa tarefa foi realizada em várias conferências (NADEAU; SEKINE, 2007) com o objetivo de

avaliar o estado da arte e impulsionar as pesquisas na área. Cada conferência classifica tipos diferentes de NEs e possui suas próprias técnicas de avaliação. A seguir, serão apresentadas as conferências mais relevantes.

O programa *Automatic Content Extraction* (ACE) (DODDINGTON et al., 2004) realizou avaliações conjuntas de NER entre 2000 e 2008 (LDC, 2018). Durante esse tempo, explorou a extração de informações como entidades, relações e eventos a partir de fontes multimídia (texto puro, áudio e imagem). Além do inglês, avaliações foram realizadas para os idiomas árabe, chinês e espanhol. Na tarefa de detecção de entidades (*Entity Detection and Tracking* - EDT), não só o nome da entidade deve ser anotado, mas todas as menções a uma entidade como uma descrição ou pronome. As categorias de NEs anotadas são: Pessoa, Organização, Local, Instalação, Arma, Veículo e Entidade geopolítica.

A Conferência CoNLL (*Conference on Natural Language Learning*) (SANG; MEULDER, 2003) realizou a tarefa compartilhada *Language-independent Named Entity Recognition* em 2002 e 2003. Em 2002, os idiomas avaliados foram o espanhol e o holandês; já em 2003, os idiomas foram o inglês e o alemão. Ferramentas diferentes foram utilizadas para pré-processar os textos em inglês e em alemão e os participantes tiveram acesso ao *corpus* depois que esse pré-processamento foi feito. As entidades anotadas pelos participantes da tarefa compartilhada foram: Pessoa, Local, Organização e Miscelânea (entidades diversas que não pertencem a uma das categorias anteriores). Os organizadores estavam interessados em abordagens que exploravam recursos extras como dicionários geográficos e dados não anotados.

O HAREM foi uma iniciativa semelhante para o Português. Inspirado na MUC, o HAREM se baseou em um modelo semântico diferente no que diz respeito à importância do contexto e à possibilidade de atribuir mais de uma classificação a uma NE (SANTOS; CARDOSO, 2007). O objetivo do HAREM é identificar e classificar as NEs do texto em 10 categorias: Abstração, Acontecimento, Coisa, Local, Obra, Organização, Pessoa, Tempo, Valor e Outro. A tarefa de identificação avalia se a *string* reconhecida é realmente uma NE, ou seja, verifica os limites da NE; a tarefa de classificação verifica, além dos limites, se a categoria atribuída está correta.

A primeira edição do HAREM aconteceu entre 2004 e 2006 e incluiu dois eventos de avaliação: o primeiro HAREM (10 participantes) e o Mini-HAREM (5 dos 10 participantes do primeiro HAREM). A segunda edição (Segundo HAREM) aconteceu entre 2007 e 2008 e também contou com a participação de 10 sistemas. O Segundo HAREM incluiu a normalização de expressões temporais e o reconhecimento de relações semânticas entre NEs. As relações anotadas são: identidade (NEs que se referem à mesma entidade), inclusão (NE que inclui outra) e localização (NE que corresponde à localização de um evento ou organização) (MOTA; SANTOS, 2008).

*Text Analysis Conference* (TAC) (NIST, 2018) é uma série de *Workshops* de

avaliação que sucedeu a ACE. Idiomas explorados na TAC incluem inglês, chinês e espanhol e a tarefa *Entity Discovery and Linking* (EDL) tem como objetivo identificar NEs em textos escritos nesses idiomas e incorporá-las em uma base de conhecimento em inglês. As NEs são classificadas nas seguintes categorias: Pessoa, Organização, Local, Instalação e Geopolítico. Em 2018, a TAC propôs estender essas cinco categorias para milhares definidas na ontologia YAGO (SUCHANEK; KASNECI; WEIKUM, 2007). Foram selecionadas 7309 entidades que incluem doenças, alimentos e entidades biomédicas.

Como grande parte dos sistemas de NER foram desenvolvidos para textos de notícias e não tem um desempenho satisfatório em gêneros mais informais, em 2015, o *Workshop on Noisy User-generated Text* (W-NUT) inseriu uma tarefa compartilhada de NER em textos do *Twitter* (BALDWIN et al., 2015). W-NUT classificou 10 tipos de NEs: Pessoa, Local, Organização, Instalação, Filme, Artista, Produto, Equipe de Esporte, Programa de TV e Outro. Em 2017, o foco dessa tarefa compartilhada foi avaliar a capacidade de identificar e classificar NEs incomuns ou novas em textos com ruído já que essas NEs são difíceis até para um especialista humano classificar (XU et al., 2018).

Tabela 1 – Conferências de Avaliação Conjunta para o NER

| **Conferência** | **Anos de avaliação** | **Categorias de NEs** | **Idiomas** |
|---|---|---|---|
| MUC | 1996 e 1998 | Pessoa, Local, Organização, Data, Hora, Valor Monetário e Porcentagem | Inglês |
| ACE | 2000 a 2008 | Pessoa, Local, Organização, Instalação, Arma, Veículo e Entidade Geopolítica | Inglês, Árabe, Chinês e Espanhol |
| CoNLL | 2002 e 2003 | Pessoa, Local, Organização e Miscelânea | Inglês, Alemão, Espanhol e Holandês |
| HAREM | 2004 e 2007 | Pessoa, Local, Organização, Abstração, Acontecimento, Coisa, Obra, Tempo, Valor e Outro | Português |
| TAC | 2009 a 2018 | Pessoa, Local, Organização, Entidade Geopolítica e Instalação até 2017. Em 2018, inclusão de 7309 NEs. | Inglês, Chinês e Espanhol |
| W-NUT | 2015 a 2017 | Pessoa, Local, Organização, Instalação, Filme, Artista, Produto, Equipe de Esporte, Programa de TV e Outro | Inglês |

Um resumo de algumas informações desses eventos é apresentado na Tabela 1. Como é possível observar, todos classificam as categorias Pessoa, Organização e Local que

são os tipos mais estudados de NEs (JIANG, 2012).

## 2.3 Gramáticas Locais

Uma forma de representar as regras da abordagem linguística são as gramáticas locais (*Local Grammar - LG*), formalismo introduzido por Maurice Gross (GROSS, 1997). "Gramáticas locais são gramáticas de estados finitos ou autômatos de estados finitos que representam conjuntos de expressões de uma língua natural" (GROSS, 1999). Essas gramáticas são construídas manualmente e são uma forma de agrupar ou capturar expressões que possuem características comuns, sejam elas sintáticas ou semânticas, que têm sido usadas em tarefas como NER (PIROVANI; OLIVEIRA, 2015) e análise de sentimentos (WILLIAMS et al., 2015).

A ferramenta Unitex (Unitex, 2018), um conjunto de *software* livres para NLP, permite representar uma LG como um conjunto de um ou vários grafos, chamados de grafos de gramáticas locais (*local grammar graph - LGG*). O Unitex possui várias ferramentas para NLP e permite, além da construção de LGGs e sua aplicação para extração de informações, pré-processamento de *corpora*, aplicação de dicionários e comparação de concordâncias. Ele é um sistema *open-source* desenvolvido inicialmente na Universidade Marne-La-Vallée (França) e disponível gratuitamente. O artigo Muniz et al. (2005) apresenta o desenvolvimento dos recursos linguísticos para o Português Brasileiro (UNITEX-PB) nessa ferramenta.

O LGG da Figura 3 reconhece *Quem é*, *Quem era* ou *Quem foi* seguido de palavras que iniciam com letra maiúscula, identificadas pelo código `<FIRST>` nos dicionários do Unitex. Símbolos que aparecem entre < e > são interpretados pelo Unitex como código de propriedade lexical nos dicionários ou como lema. Entre as palavras iniciadas com letra maiúscula podem aparecer preposições ou abreviações cujo reconhecimento foi detalhado previamente nos grafos *Preposicao.grf* e *Abreviacoes.grf*, incluídos nesse como subgrafo. Referências a subgrafos são representados em nós com fundo cinza pelo Unitex. Exemplos de expressões reconhecidas pelo grafo (ocorrências) são *Quem é Albert Einstein* e *Quem foi Antônio de Oliveira Salazar*.

O Unitex permite incluir saídas no grafo, representadas em negrito sob setas. Grafos com saídas são chamados de transdutores. Ao aplicar grafos para extrair padrões em um texto, as saídas podem ser: ignoradas (opção *are not taken into account*), usadas para substituir a sequência reconhecida no arquivo de concordância (opção *REPLACE recognized sequences*) ou anexadas ao arquivo de concordância (opção *MERGE with input text*).

Na Figura 3, `<PESSOA>` e `</PESSOA>` embaixo das setas representam saídas que serão anexadas à lista de ocorrências identificadas pelo Unitex, chamada concordância, se

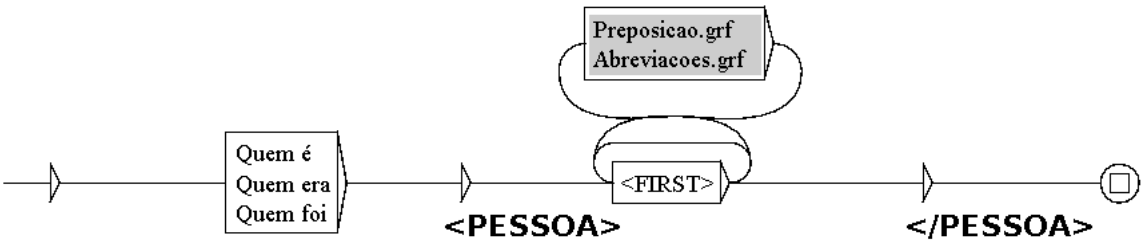


Figura 3 – LGG criado no Unitex (ReconhecePerguntasComNomes.grf)

o grafo for aplicado no modo *MERGE with input text*. Assim, os nomes identificados serão apresentados entre essas *tags* no arquivo de concordância. Um exemplo de concordância para esse grafo, quando aplicado à CD do Segundo HAREM, pode ser visto na Figura 4. As expressões completas reconhecidas pelo grafo aparecem sublinhadas no arquivo de concordância, que também apresenta uma parte do contexto à esquerda e do contexto à direita das expressões.

ram no atentado na Oktoberfest? </P> <P>Quem é<PESSOA> Henning Mankell</PESSOA>?{S} Como se cha
u a construção da Torre Eiffel? </P> <P>Quem é<PESSOA> Samuel Hahnemann</PESSOA>?{S} Em homeopa
Christian von Holst em Gdansk? </P> <P>Quem é<PESSOA> Werner Herzog</PESSOA>? </P> <P>Em que c
é que tem lugar a Semana Verde? </P> <P>Quem era<PESSOA> Herbert Erhardt</PESSOA>?{S} Quando fo
? </P> <P>Que significa a sigla RAF?{S} Quem era<PESSOA> Rolf Heissler</PESSOA>?{S} Diga três m

Figura 4 – Exemplo de concordância para o LGG da Figura 3

Observe que esse LGG identifica nomes através do contexto à esquerda, ou seja, de palavras à esquerda do nome que indicam, de alguma forma, que o que aparece depois é um nome de pessoa. Outras regras que identificam nomes capturam o contexto à direita.

Segundo Friburger e Maurel (2004), transdutores de estados finitos são os melhores formalismos para representar fenômenos linguísticos complexos e precisos e são fáceis de usar. Alguns trabalhos que usaram o formalismo de LGs são: (BAPTISTA, 1998) para representar propriedades linguísticas de nomes de pessoas, (PICOLI et al., 2015) para coletar exemplos para o estudo de propriedades sintático-semânticas de verbos, (HAN et al., 2018) para reconhecer e extrair *multiword expressions* coreanas para análise de sentimentos. Além disso, vários trabalhos usaram LGs para representar as regras da abordagem linguística para o NER conforme apresentado na Seção 3.1.

## 2.4 Campos Aleatórios Condicionais

Campos Aleatórios Condicionais (LAFFERTY; MCCALLUM; PEREIRA, 2001) (*Conditional Random Fields - CRF*) é uma técnica de aprendizado de máquina para

predição estruturada que vem sendo utilizada com sucesso em diversas atividades de Processamento de Linguagem Natural (*Natural Language Processing - NLP*), incluindo o NER. O NER é tratado como um problema de rotulação de sequências e um modelo condicional é construído a partir de uma base de treino para predizer qual a melhor sequência de rotulação dada uma sentença de entrada.

Seja X = ($x_1$, $x_2$, ..., $x_T$) uma sequência de T palavras em um texto, deseja-se determinar a melhor sequência de rótulos Y = ($y_1$, $y_2$, ..., $y_T$) para essas palavras, correspondentes às categorias de NEs ou o rótulo Outro (O) nesse trabalho. O objetivo é atribuir um rótulo $y_i$ a cada observação $x_i$ representada normalmente como um vetor de características. Na rotulação de sequências, assume-se que o rótulo $y_i$ depende não apenas de sua observação $x_i$ correspondente, mas também possivelmente de outras observações e outros rótulos na sequência (JIANG, 2012).

O CRF é um modelo gráfico. Ele é utilizado para modelar a estrutura de dependência condicional entre variáveis aleatórias que podem ser representadas como um grafo não direcionado. A estrutura mais comum de dependências entre as variáveis é apresentada em um CRF de cadeia linear que representa essas dependências em uma sequência temporal. Ou seja, um CRF de cadeia linear prediz as variáveis de saída como uma sequência.

A Figura 5 representa a estrutura de um CRF de cadeia linear.

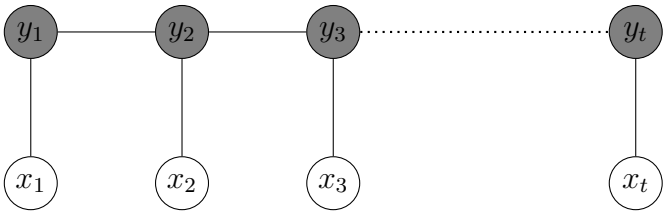


Figura 5 – Estrutura gráfica de um CRF de cadeia linear

O CRF modela uma distribuição condicional p(Y|X) que representa a probabilidade de obter a saída Y dada a entrada X. Segundo Sutton e McCallum (2012), um CRF de cadeia linear é uma distribuição condicional como apresentada na Equação 2.1:

$$p(Y|X) = \frac{1}{Z(X)} \prod_{t=1}^{T} \exp\left\{ \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\} \tag{2.1}$$

onde Z(X) é uma função de normalização, garantindo que a soma das probabilidades seja 1, dada pela Equação 2.2:

$$Z(X) = \sum_{Y} \prod_{t=1}^{T} \exp\left\{ \sum_{k=1}^{K} \theta_k f_k(y_t, y_{t-1}, \mathbf{x}_t) \right\} \tag{2.2}$$

$F = \{f_k(y_t, y_{t-1}, \mathbf{x}_t)\}_{k=1}^{K}$ é um conjunto de K funções características que devem ser definidas de acordo com o problema e $\theta = \{\theta_k\}_{k=1}^{K}$ é um vetor de pesos que deve ser

estimado a partir de um conjunto de treino $D = \{X^{(i)}, Y^{(i)}\}_{i=1}^{N}$ contendo N exemplos.

As funções $f_k$ recebem como entrada um par de estados adjacentes ($y_t$ e $y_{t-1}$) e o vetor $\mathbf{x}_t$ que contém todas as componentes das observações globais X que são necessárias para computar características no tempo t. Uma função $f_k$ pode depender somente de $y_t$ ou de $y_t$ e $y_{t-1}$. Uma função característica que depende apenas de $y_t$ é chamada característica de estado (*state feature*) e quando depende de $y_t$ e $y_{t-1}$ é chamada de característica de transição (*transition feature*) (FIELDMAN; SANGER, 2006). Um exemplo de função característica de transição é apresentado em 2.3. Essa função retorna 1 se a palavra está escrita em letras maiúsculas (componente do vetor $\mathbf{x}_t$), seu rótulo é Organização ($y_t$) e o rótulo anterior ($y_{t-1}$) é Outro; caso contrário retorna 0.

$$f_1 = \begin{cases} 1, & \text{se } y_t = \text{Organização, } y_{t-1} = \text{Outro e uma componente de} \\ & \mathbf{x}_t \text{ indica que a palavra está escrita em letras maiúsculas} \\ 0, & \text{caso contrário} \end{cases} \tag{2.3}$$

Os pesos $\theta_k$ dependem de cada função característica e quanto mais discriminativa for a função, mais alto será seu peso computado. O modelo CRF combina os pesos de cada função característica para determinar a probabilidade de certo valor $y_t$. Isto é, se duas funções características são válidas ao mesmo tempo para uma sentença, as duas se sobrepõem e aumentam a probabilidade de $y_t$.

Uma forma de estimar o vetor de pesos é usar estimativa de máxima verossimilhança (*maximum likelihood*) (BUSSAB; MORETTIN, 2010) que maximiza a probabilidade condicional do conjunto de treino. Uma forma de predizer a melhor sequência de rotulação (sequência de maior probabilidade) $Y^* = argmax_Y P(Y|X)$ para uma dada entrada X é o algoritmo de Viterbi (SUTTON; MCCALLUM, 2012).

HMM (*Hidden Markov Model*) e MEMM (*Maximum Entropy Markov Models*) são outros métodos de aprendizado de máquina estatísticos utilizados no NER. HMM são modelos gerativos que modelam o conjunto completo de probabilidade de observações e estados ocultos e, por isso, computam probabilidades que não são necessárias. Já o CRF modela a probabilidade condicional diretamente (modelo discriminativo). Além disso, HMM não pode usar funções características sobrepostas como o CRF (RUSSEL; NORVIG, 2004). Já os MEMM são modelos discriminativos como os CRF, porém sofrem do problema de viés de rótulo (*label bias problem*) (FIELDMAN; SANGER, 2006).

Constant e Tellier (2012) afirmam que os CRFs de cadeia linear são os melhores modelos estatísticos atualmente para aprender a anotar sequências. Alguns trabalhos que usaram CRF em tarefas que envolvem segmentação e rotulação de sequências são: segmentação de imagens para área médica (KARIMAGHALOO; ARNOLD; ARBEL, 2016), rotulação de seções em textos (RAMESH et al., 2016) e NER em diversos idiomas

(COPARA et al., 2016; JIA et al., 2016; SEKER; ERYIGIT, 2012).

# 3 Trabalhos correlatos

Este capítulo apresenta os principais trabalhos correlatos, tendo como foco os trabalhos aplicados a bases de dados em Português. A primeira seção apresenta os trabalhos correlatos que usaram a abordagem linguística para o NER, enquanto a segunda seção menciona os que usaram a abordagem de aprendizado de máquina e a terceira os que usaram a abordagem híbrida. O capítulo é finalizado apontando as principais diferenças entre alguns desses trabalhos e o trabalho apresentado nesta Tese.

## 3.1 NER usando Abordagem Linguística

No Primeiro e Segundo HAREM, vários sistemas que realizam NER para o Português foram avaliados, sendo que a maioria deles usou abordagem linguística. No Primeiro HAREM, o sistema Stencil (SANTOS; CARDOSO, 2007) usou LGs construídas no NooJ (SILBERZTEIN, 2018) para classificar cinco categorias de NEs (Pessoa, Local, Organização, Tempo e Valor). Seu objetivo foi obter uma alta precisão sem utilizar *gazetteers*. Partes das NEs identificadas pelas LGs inicialmente foram usadas para identificar novas NEs em um segundo passo.

No Segundo HAREM, o sistema XIP (MOTA; SANTOS, 2008) também usou LGs para classificar as NEs, exceto as categorias Abstração e Coisa. O reconhecedor de NEs foi integrado ao sistema XIP, uma ferramenta de análise sintática. Além das LGs, XIP também usa a atribuição de sintagmas e dependências para classificar NEs em contextos mais complexos. É importante destacar que tanto o sistema Stencil do Primeiro HAREM quanto o XIP do Segundo, obtiveram o melhor desempenho para a categoria Tempo usando LGs, o que indica que essa abordagem deve ser explorada para essa categoria. Stencil também obteve bons resultados para a categoria Valor e XIP para Acontecimento.

Os sistemas Priberam (MOTA; SANTOS, 2008) e Rembrandt (CARDOSO, 2008) obtiveram os melhores resultados no Segundo HAREM. Os dois sistemas usaram algum tipo de léxico e regras contextuais para classificar as 10 categorias de NEs. O sistema Priberam, por exemplo, usa um léxico bastante completo que já possui uma classificação semântica. Essa classificação pode ser alterada posteriormente por regras contextuais. O Rembrandt utiliza a Wikipedia como fonte de conhecimento. As candidatas a NEs são identificadas e classificadas pela SASKIA, uma interface própria que interage com a Wikipedia. As NEs são classificadas novamente usando as regras gramaticais que consideram o contexto da NE e auxilia o processo de desambiguação.

Pirovani e Oliveira (2015) usaram o formalismo de LG para o NER em Português.

LGs foram construídas para extrair nomes de pessoas a partir de estudos linguísticos desses textos. Uma LG construída para o livro Senhora, de José de Alencar, foi aplicada também a um conjunto de artigos de um jornal local do Espírito Santo chamado A Tribuna. A idéia foi observar a apropriação de uma LG construída para um contexto em outro. Adaptações foram necessárias e o desempenho obtido para os artigos de jornal foi menor já que a LG não foi construída para eles, indicando que a identificação automática de nomes é dependente do *corpus*.

O trabalho de Rocha et al. (2016) apresentou apenas a identificação de NEs em Português utilizando léxicos, regras contextuais e *POS-tagging*. Esse sistema não identifica as categorias Tempo e Valor. O sistema PAMPO identifica entidades candidatas usando palavras iniciadas com letras maiúsculas e palavras-chave como formas de tratamento. Posteriormente, o sistema descarta algumas dessas NEs usando *POS-tagging*. Por exemplo, NEs candidatas às quais não foram atribuídas as *tags* N (substantivo) ou Prop (nome próprio) são excluídas.

Alguns trabalhos que também usaram LGs para NER em outros idiomas são apresentados em: Friburger e Maurel (2004), Bayraktar e Temizel (2008), Traboulsi (2009) e Krstev et al. (2011).

## 3.2 NER usando Aprendizado de Máquina

Milidiú, Duarte e Cavalcante (2007) avaliaram sete abordagens diferentes baseadas em *Hidden Markov Models (HMM), Support Vector Machine (SVM) e Transformation Based Learning (TBL)* para reconhecimento das NEs Local, Pessoa e Organização. Um *Baseline System (BLS)* baseado em *gazetteers* e em uma heurística simples que relaciona cada preposição à entidade que mais ocorre após ela foi utilizado para classificação inicial. De acordo com esse trabalho, SVM e TBL são boas alternativas quando linguistas podem participar da construção do sistema.

As técnicas *Naive Bayes*, SVM e árvores de decisão foram avaliadas em Pellucci et al. (2011) com o objetivo de reconhecer nomes de pessoas, lugares e organizações. Os experimentos foram realizados usando a ferramenta Weka que já tem todos os algoritmos implementados. Eles usaram 3 arquivos diferentes para representar o vetor de características (*features*) dos dados: valores inteiros para cada característica; vetor de 0 e 1 indicando a presença ou não de uma característica e variáveis nominais representando as características. Os resultados ficaram abaixo do esperado quando comparados aos resultados de Milidiú, Duarte e Cavalcante (2007). Cada técnica obteve melhor desempenho em um experimento, mas os melhores resultados foram obtidos com a técnica *Naive Bayes*.

O sistema NERP-CRF, baseado em CRF, é apresentado em Amaral (2013) e Amaral e Vieira (2014) com o objetivo de identificar e classificar as 10 categorias de NEs

do HAREM. Inicialmente as NEs são identificadas usando a notação BILOU (RATINOV; ROTH, 2009). Essa notação, a etiquetagem POS, as categorias do HAREM e um vetor de *features* são utilizados como entrada para a etapa de treino. Na etapa de teste, o sistema cria o vetor de POS e de *features* para os textos de entrada e submete para o modelo CRF gerado que treina e classifica as NEs. Os *corpora* do HAREM foram utilizados para treino e teste. O sistema proposto obteve os melhores resultados de precisão e medida-F comparado aos sistemas do Segundo HAREM para as 10 categorias.

Um modelo de NER foi criado utilizando a classe NameFinder do OpenNLP em Fonseca, Chiele e Vanin (2015) para reconhecer as NEs no *corpus* do Segundo HAREM. Diferente da maioria dos trabalhos, esse modelo foi treinado no *corpus* Amazônia (FREITAS; ROCHA; BICK, 2008). Assim, o primeiro passo foi a anotação desse corpus. Os resultados foram comparados com os resultados das ferramentas NERP-CRF (AMARAL, 2013; AMARAL; VIEIRA, 2014), LanguageTasks (Language Tasks, 2019), Freeling (FreeLing, 2018) e Palavras (BICK, 2000) para as categorias Pessoa, Lugar e Organização, apresentando resultados compatíveis com os modelos dessas ferramentas. Uma importante observação desse trabalho é a dificuldade geral dos modelos em obter bons resultados para a categoria Organização.

Pires, Devezas e Nunes (2017) usaram vários algoritmos de aprendizado de máquina, disponíveis em ferramentas de NLP, para treinar modelos de NER para o Português. Seu objetivo era descobrir a melhor abordagem e configuração para realizar o NER nas notícias do SIGARRA, o sistema de informação da Universidade de Porto. As ferramentas de NLP usadas foram Stanford CoreNLP (Stanford CoreNLP, 2019), OpenNLP (Apache OpenNLP, 2019), Spacy (spaCy, 2019) e NLTK (BIRD; KLEIN; LOPER, 2009). Inicialmente, os algoritmos foram avaliados no corpus do Segundo HAREM e os melhores resultados foram obtidos com o Stanford CoreNLP que usa o CRF como técnica de aprendizado de máquina para NER. Na sequência, foi realizado um estudo dos hiperparâmetros, obtendo melhorias em cada ferramenta. Por fim, o autor anotou um *corpus* em Português, denominado SIGARRA News Corpus[1], para avaliar o desempenho da ferramenta no domínio pretendido.

*Corpora* em Português e Inglês foi usado por Santos (2017) para comparar classificadores baseados em CRF, HMM e MEMM na tarefa de NER. A pesquisa concluiu que o CRF é o classificador que obtém melhores resultados tanto para o Português, quanto para o Inglês. O autor também investigou a contribuição de *features* individualmente e em conjunto, concluindo que afixos, *POS-tagging* e capitalização são as que mais contribuem. Avaliando o tamanho da janela de *features* de contexto, a de 5 posições apresentou melhor resultado em relação às de 3 e 7.

Santos e Guimaraes (2015) apresentou um sistema independente de linguagem

[1] https://rdm.inesctec.pt/dataset/cs-2017-004

baseado na rede neural profunda CharWNN. Essa rede usa vetores de representação de palavras (*word embeddings*) e caracteres (*character embeddings*) para executar classificação sequencial. O sistema foi testado para o Português e Espanhol e, no caso do Português, a CD do Primeiro HAREM foi usada como treino e a do Mini-HAREM como teste. A abordagem apresentou melhores resultados comparados ao sistema $\text{ETL}_{CMT}$ (SANTOS; MILIDIÚ, 2012), um método que usa *Entropy Guided Transformation Learning (ETL)*.

O trabalho de Castro, Silva e Soares (2018) também apresentou uma arquitetura de rede neural profunda com vetores de representação de palavras e caracteres para o NER em Português. Para isso, foi utilizado o Bi-LSTM-CRF (*bidirectional Long Short-Term Memory with Conditional Random Fields*). Os autores testaram vários parâmetros para treinamento como modelos de *word embedding*, notações de rotulação, capitalização das palavras e número de unidades ocultas para a rede, obtendo um conjunto de valores ótimos para esses parâmetros. Esse sistema obteve 5 pontos percentuais de ganho em relação ao resultado de Santos e Guimaraes (2015).

Costa e Paetzold (2018) obteve resultados muito próximos ao de Castro, Silva e Soares (2018) pois usou uma abordagem muito semelhante. A abordagem combina LSTM e CRF para duas tarefas de rotulação de sequência: *POS-tagging* e NER.

Alguns trabalhos realizaram o NER para o Português em domínios específicos como os de Goulart e Lima (2009) em textos de biomedicina, Silva (2012) em notícias do governo e Amaral (2017) em textos da área de Geologia.

Konkol, Brychcín e Konopík (2015), Lample et al. (2016), Yang, Zhang e Dong (2017), Zhang et al. (2017) e Poostchi e Piccardi (2018) apresentaram trabalhos relevantes utilizando aprendizado de máquina para o NER em outros idiomas. Yang, Zhang e Dong (2017) apresentaram resultados estado da arte para o inglês usando uma estratégia interessante de *reranking* para melhorar resultados já obtidos a partir de soluções *baseline* como Bi-LSTM-CRF. Zhang et al. (2017) também usou Bi-LSTM-CRF para tarefa EDL na conferência TAC, mas usou um Bi-LSTM adicional para consumir representações de características (*features*) externas das palavras que são concatenadas com as representações de palavras antes da camada de saída.

## 3.3 NER usando Abordagem Híbrida

Ferreira, Balsa e Branco (2007) apresentaram uma ferramenta combinando uma abordagem baseada em regras para classificar números, medidas, tempo e endereços com uma abordagem de aprendizado de máquina para classificar outras NEs, incluindo Pessoa, Local, Organização, Obra, Evento e Miscelânea em textos em Português. Testes foram realizados com classificadores baseados em HMM e MEMM e diferentes notações para rotulação. Os classificadores foram treinados no *corpus* descrito em Barreto et al. (2006).

Foi realizada uma avaliação parcial do módulo baseado em regras e do melhor modelo obtido (HMM) com a coleção do HAREM visto que o sistema não classifica as 10 categorias.

O trabalho apresentado em Mota (2010) combina LGs criadas no NooJ (SILBERZTEIN; MULLER; ROYAUTÉ, 2004) com *Co-training* para classificar as NEs Pessoa, Local e Organização em Português. As LGs são usadas inicialmente para identificar NEs candidatas que são usadas para iniciar o classificador *Co-training* na fase de treino e são rotuladas pela lista de decisão inferida pelo *Co-training* na fase de teste. Após a classificação, as NEs classificadas são usadas para reconhecer outras ocorrências de mesmo nome que não tinham contexto a princípio. O corpus CETEMPublico (ROCHA; SANTOS, 2000) foi usado para treino e teste.

Uma combinação de *K-Nearest Neighbors (KNN)* e CRF para identificar NEs em *tweets* em Inglês foi proposto em Liu et al. (2011). Devido a informação insuficiente em um *tweet* e a indisponibilidade de dados de treinamento, uma aprendizagem semi-supervisionada e 30 *gazetteers* foram usados. O KNN conduz uma classificação a nível de palavra e os resultados rotulados por ele são inseridos como uma das características enviadas para o CRF. Os modelos KNN e CRF são treinados repetidamente com um conjunto de treino aumentado incrementalmente. O método apresentou vantagens sobre os *baselines*.

Em Constant e Tellier (2012), os autores propõem avaliar o impacto de usar recursos léxicos externos em um CRF a fim de executar a tarefa conjunta de segmentação *multiword* e *part-of-speech tagging (POS-tagging)* em Francês. A informação obtida a partir de dicionários e gramáticas locais reconhecendo números e algumas NEs como organização e local foram enviadas como características para um CRF de duas maneiras diferentes: concatenando cada possível categoria POS (*Learn-concat*) e considerando cada possível categoria nos recursos como uma nova propriedade booleana (*Learn-bool*). Eles obtiveram um ganho de 0.5 pontos percentuais em medida-F e mostraram que a integração das características baseadas em léxico compensa significativamente o uso de um corpus pequeno de treino.

## 3.4 Comparação com o Trabalho Apresentado nesta Tese

O presente trabalho tem como objetivo realizar o NER para as 10 categorias do HAREM utilizando CRF como foi realizado em Amaral (2013). Porém, o pré-processamento dos textos e a rotulação das NEs foram realizados de formas diferentes. Além disso, a principal diferença é que uma informação inicial sobre o rótulo de cada palavra foi obtida por uma LG e acrescentada ao conjunto de características enviadas ao CRF. Essa informação adicional é uma possibilidade de melhorar o desempenho de sistemas NER que utilizam aprendizado de máquina, incorporando o conhecimento humano.

Este trabalho também difere do apresentado em Liu et al. (2011) e Constant e Tellier (2012) por combinar uma abordagem baseada em regras com o CRF para o NER em Português e por não utilizar *gazetteers* ou *dicionários*. A construção dessas listas de NEs é custosa e, como discutido no início do Capítulo 1, a existência de grandes *gazetteers* não corresponde necessariamente a um maior desempenho. A LG construída neste trabalho pode compensar a ausência de dicionários já que suas regras podem capturar uma grande quantidade de NEs, inclusive NEs que não aparecem nesses dicionários.

Algumas estratégias usadas por sistemas híbridos para rotular NEs são: manter as NEs extraídas por cada uma das abordagens utilizadas e usar alguma estratégia para resolução de ambiguidades (ROCKTÄSCHEL; WEIDLICH; LESER, 2012), utilizar uma sequência de classificadores específicos para cada classe (SRIHARI; NIU; LI, 2000; FERREIRA; BALSA; BRANCO, 2007), utilizar uma estratégia para identificação e outra para classificação de NEs (MOTA, 2010) e combinar classificadores usando uma estratégia de votação (KOZAREVA et al., 2007). Diferente dessas estratégias, o rótulo atribuído pela LG neste trabalho não é usado como rótulo final. Evidências capturadas previamente pela LG permitem o CRF realizar a rotulação final corretamente de um número maior de NEs, conforme apresentado na Seção 7.1.

# 4 Bases de Dados e Avaliação

Este capítulo apresenta as bases de dados utilizadas nos experimentos, a metodologia utilizada para avaliação da abordagem proposta e o teste estatístico aplicado sobre os resultados.

## 4.1 Bases de Dados

Os *corpora* usados como referência para avaliação dos sistemas participantes durante as duas edições do HAREM foram usados como bases de dados neste trabalho. Esses *corpora* foram anotados manualmente por humanos e são conhecidos como Coleções Douradas (CD) do HAREM. São eles: CD do Primeiro HAREM, CD do Mini-HAREM e CD do Segundo HAREM. Essas CDs incluem documentos de diferentes gêneros textuais como técnico, político, jornalístico, e-mails e entrevistas e estão escritos em Português, principalmente do Brasil e de Portugal.

A Tabela 2 apresenta algumas informações sobre a quantidade de documentos, palavras e NEs disponíveis nas CDs do HAREM.

Tabela 2 – Caracterização das CDs do HAREM

| **Coleção** | **Documentos** | **Palavras** | **NEs** |
|---|---|---|---|
| CD Primeiro HAREM | 129 | 80.060 | 4.997 |
| CD Mini-HAREM | 128 | 54.074 | 3.612 |
| CD Segundo HAREM | 129 | 147.991 | 7.836 |

A Tabela 3 apresenta a distribuição de NEs por categoria nas CDs do HAREM. A diferença no total de NEs nas Tabelas 2 e 3 se referem às NEs vagas (possuem mais de uma interpretação).

Na primeira edição do HAREM, as categorias mais frequentes foram Local, Pessoa e Organização, nessa ordem. Já na segunda edição, as categorias mais frequentes foram Pessoa, Local e Tempo.

As CDs do HAREM utilizadas foram obtidas no *site* da Linguateca[1].

Além das CDs do HAREM, outros dois *corpus* foram usados em experimentos neste trabalho: aTribuna e SIGARRA[2]. Informações gerais sobre eles são apresentadas na Tabela 4.

O *corpus* aTribuna foi anotado durante este trabalho. Ele é um *corpus* que contém

[1] http://www.linguateca.pt/HAREM/
[2] https://rdm.inesctec.pt/dataset/cs-2017-004

Tabela 3 – Distribuição de NEs por categoria nas CDs do HAREM

| Categoria | CD Primeiro HAREM | CD Mini-HAREM | CD Segundo HAREM |
|---|---|---|---|
| Pessoa | 1.029 | 836 | 2.036 |
| Local | 1.286 | 895 | 1.311 |
| Organização | 956 | 622 | 961 |
| Tempo | 434 | 364 | 1.189 |
| Valor | 484 | 328 | 353 |
| Abstração | 449 | 326 | 286 |
| Acontecimento | 128 | 63 | 300 |
| Obra | 222 | 130 | 449 |
| Coisa | 82 | 180 | 308 |
| Outro | 40 | 14 | 79 |

100 artigos de notícias extraídos do jornal local A Tribuna[3] do Espírito Santo. Os artigos foram aleatoriamente selecionados dentre os artigos de política e economia. Alunos de graduação anotaram os nomes de pessoas nesses artigos usando a ferramenta Etiquet(H)arem[4]. Cada artigo foi anotado por um aluno e as anotações foram revisadas posteriormente pela autora desta Tese.

SIGARRA[5] é um conjunto de textos de notícias da Universidade de Porto, escrito em Português de Portugal. Esse *corpus* é anotado com 8 categorias de NEs: Pessoa, Localização, Organização, Data, Hora, Curso, Evento e Unidadeorganica. Unidadeorganica se refere a uma NE do modelo organizacional da Universidade do Porto (Ex: FADEUP - Faculdade de Esporte e FAUP - Faculdade de Arquitetura). Data é a categoria mais frequente nesse *corpus* que contém 12644 NEs anotadas.

Tabela 4 – Caracterização dos *corpus* aTribuna e SIGARRA

| Coleção | Documentos | Palavras | NEs | Categorias de NEs |
|---|---|---|---|---|
| aTribuna | 100 | 101.733 | 2.714 | Pessoa |
| SIGARRA | 905 | 185.000 | 12.644 | Pessoa, Lugar, Organização, Data, Curso, Evento, Unidadeorganica, Hora |

## 4.2 Métricas de Avaliação e Validação Cruzada

Com a finalidade de avaliar o desempenho da abordagem proposta neste trabalho, as métricas comumente usadas para avaliar o desempenho de sistemas de NER foram

[3] https://tribunaonline.com.br/
[4] http://www.linguateca.pt/poloCoimbra/recursos/etiquetharem.zip
[5] https://rdm.inesctec.pt/dataset/cs-2017-004

utilizadas. São elas (MOTA; SANTOS, 2008):

$$Precisão = \frac{\text{Total de NEs identificadas corretamente}}{\text{Total de NEs identificadas}} \tag{4.1}$$

$$Abrangência = \frac{\text{Total de NEs identificadas corretamente}}{\text{Total de NEs realmente existentes no corpus}} \tag{4.2}$$

$$Medida - F = \frac{\text{2 x Precisão x Abrangência}}{\text{Precisão + Abrangência}} \tag{4.3}$$

A precisão representa a quantidade de acertos no total de NEs identificadas. Quanto maior a precisão, menor o número de NEs identificadas de forma errada (falso-positivos). A abrangência representa a quantidade de acertos no total de NEs existentes. Quanto maior, menor a quantidade de NEs não identificadas (falso-negativos). A medida-F é uma média harmônica das outras duas.

Neste trabalho, as métricas foram computadas usando os *scripts* de avaliação do Segundo HAREM[6]. Esses *scripts* usam uma fórmula (MOTA; SANTOS, 2008) para computar o que está correto (Total de NEs identificadas corretamente). É necessário identificar corretamente a EN completa, todas as suas partes, para receber o valor 1. O valor total da medida é obtido somando esse valor às parcelas relativas aos acertos e erros de classificação (categoria, tipo e subtipo) de acordo com o modo de avaliação escolhido pelo usuário. Ou seja, o HAREM penaliza classificações erradas.

Uma estratégia comum para avaliar a maioria dos sistemas que realizam o NER em Português é usar uma das CDs do HAREM para treino e uma outra para teste. Neste trabalho, a CD do Primeiro HAREM foi utilizada inicialmente para treino.

Métodos de amostragem (FACELI et al., 2011) foram utilizados para obter uma estimativa de desempenho mais confiável ao usar a mesma CD para treino e teste. A ideia é dividir a base de dados usando uma parte para treino e outra parte para teste, garantindo que os dados de teste não foram utilizados durante o aprendizado. Assim, é possível obter uma boa estimativa de desempenho do algoritmo (ARLOT; CELISSE et al., 2010).

A estratégia *holdout* consiste em dividir a base de dados em dois subconjuntos mutuamente exclusivos. Uma divisão comum é considerar 2/3 dos dados para treino e 1/3 dos dados restante para teste.

Já a estratégia de validação cruzada *k-fold cross-validation* consiste em dividir a base de dados em $k$ subconjuntos (*folds*) mutuamente exclusivos de tamanhos aproximadamente iguais e realizar $k$ iterações de treino e teste sendo que, em cada uma delas, um subconjunto diferente é usado para teste e os outros $k$-1 subconjuntos são usados para treino. O

[6] www.linguateca.pt/harem/avaliacao/Av_HAREM_XML.zip

desempenho final é uma média das métricas computadas em cada subconjunto de teste.

## 4.3 Teste dos Postos Sinalizados de Wilcoxon

O teste estatístico dos postos sinalizados de Wilcoxon (BUSSAB; MORETTIN, 2010) tem como objetivo comparar duas amostras com observações pareadas, resultantes das medidas obtidas da aplicação de dois métodos ao mesmo conjunto de dados. É utilizado em experimentos do tipo "controle" versus "tratamento" com o interesse de verificar o efeito do tratamento.

Sendo assim, esse é um teste que pode ser utilizado para comparar diferentes sistemas e verificar qual deles obtém o melhor resultado. Nesse caso, a Hipótese nula ($H_0$) especifica que os dois sistemas são equivalentes, ou seja, que o novo sistema (tratamento) não tem efeito.

O teste é não paramétrico, o que significa que não é necessário fazer uma suposição sobre a distribuição dos valores analisados. Dadas duas amostras com observações pareadas, o teste é baseado nos postos (*rank*) sinalizados das diferenças dessas observações. A partir desses valores, é possível calcular a estatística do teste e concluir se a Hipótese é ou não aceitável.

Considerando $n$ o tamanho da amostra (número de pares (X,Y) observados), segue o passo a passo do teste (TRIOLA, 2014):
Para cada par de dados, faça

1. Calcule a diferença D = X - Y e descarte os pares cuja diferença seja 0.
2. Obtenha o posto (*rank*) de |D|: ordene as diferenças D, da menor para a maior, independente do sinal. O posto de |D| é a posição obtida nessa ordenação (1, 2, ... , $n$). Caso aconteça um empate, atribua a média dos postos correspondentes.
3. Obtenha o posto sinalizado: atribua a cada posto o sinal da diferença que o originou.
4. Calcule a estatística T do teste: Sendo $T^+$ a soma dos postos positivos e $T^-$ o valor absoluto da soma dos postos negativos, considere como estatística do teste o menor valor entre $T^+$ e $T^-$.
5. Determine o valor crítico com base no tamanho amostral considerando o número de pares cuja diferença é diferente de 0.

Rejeita-se $H_0$ se o $T$ calculado for menor do que ou igual ao valor crítico tabelado (consulte Apêndice A).

# 5 A abordagem proposta: CRF+LG

CRF+LG combina uma rotulação obtida por um CRF de cadeia linear com uma classificação obtida através de LG. Sutton e McCallum (2012) afirmam que um tipo interessante de característica para o CRF pode ser o resultado de métodos mais simples para a mesma tarefa. Assim, as LGs realizam uma pré-rotulação capturando evidências gerais de NEs nos textos e o CRF realiza uma rotulação sequencial utilizando essa pré-rotulação. A pré-rotulação é enviada para o CRF junto das outras características de entrada e pode ser vista como uma sugestão para o CRF.

A Figura 6 apresenta uma visão geral da metodologia utilizada para treino.

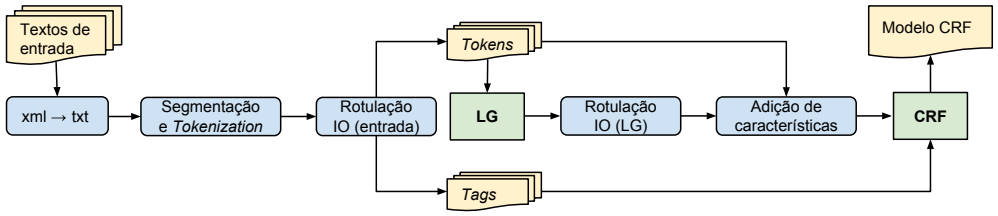


Figura 6 – Fluxograma da metodologia usada para treino

Inicialmente, os textos de entrada (.xml anotados) são convertidos em arquivos texto (.txt), mantendo apenas as categorias de NEs utilizadas, no formato apresentado na Figura 7.

> O software pode ser personalizado, e é oferecido gratuitamente na <Web_LOCAL>. <Cerca_de_400_VALOR> projetos em <30_-VALOR> países estão usando o sistema, afirma <Liebenberg_PESSOA>.

Figura 7 – Exemplo de texto de entrada retirado da CD do Segundo HAREM

Cada arquivo texto passa pelo processo de segmentação em sentenças e *tokenization*. A segmentação foi realizada pela ferramenta Unitex[1]. O Unitex utiliza LGs para descrever os diferentes contextos para o fim de uma sentença. Neste trabalho, a LG que realiza segmentação de sentenças no Unitex foi alterada para não segmentar as sentenças em dois-pontos (:) e ponto e vírgula (;). Essa flexibilidade é um ponto forte da ferramenta.

Os arquivos segmentados passam pelo processo de *tokenization* utilizando a biblioteca OpenNLP[2]. Essa biblioteca é baseada em aprendizagem de máquina e realiza tarefas

[1] http://unitexgramlab.org/
[2] http://opennlp.apache.org/

comuns de NLP como segmentação, *tokenization*, *POS-Tagging*, etc. O *script* desenvolvido para realizar esse processo não permite a *tokenization* de uma lista de abreviações importantes para o NER como *D.*, *Prof.* e *R.*.

A Figura 8 ilustra o resultado do pré-processamento realizado nessa etapa. Observe que o Unitex insere a saída {S} para delimitar sentenças e o OpenNLP utiliza espaços para delimitar *tokens*.

O software pode ser personalizado , e é oferecido gratuitamente na <Web_LOCAL> .{S} <Cerca_de_400_VALOR> projetos em <30_-VALOR> países estão usando o sistema , afirma <Liebenberg_PES-SOA> .{S}

Figura 8 – Exemplo de arquivo de entrada segmentado e tokenizado

Com o objetivo de representar o NER como um problema de rotulação de sequência, um rótulo deve ser atribuído a cada *token* do texto. As anotações das NEs precisam ser convertidas em uma sequência de rótulos. Várias notações podem ser usadas para essa delimitação de NEs e identificação dos *tokens* no texto (KONKOL; BRYCHCÍN; KONOPÍK, 2015), mas a IO foi escolhida por apresentar resultados melhores em testes prévios realizados durante este trabalho. Além disso, Amaral, Buffet e Vieira (2015) apresentaram melhores resultados com a notação IO em relação à BILOU na CD do Segundo HAREM.

Tabela 5 – Exemplo de rotulação IO

| ***Tokens*** | ***Tags*** |
|---|---|
| Cerca | I-VALOR |
| de | I-VALOR |
| 400 | I-VALOR |
| projetos | O |
| em | O |
| 30 | I-VALOR |
| países | O |
| estão | O |
| usando | O |
| o | O |
| sistema | O |
| , | O |
| afirma | O |
| Liebenberg | I-PESSOA |
| . | O |

A notação IO é utilizada da seguinte forma: todos os *tokens* que fazem parte de uma NE são rotulados com I (*Inside*) e todos os demais *tokens* com O (*Outside*). Nesse

caso, a classe da NE também é mencionada no rótulo I como ilustrado na Tabela 5 para a sentença *Cerca de 400 projetos em 30 países estão usando o sistema, afirma Liebenberg.*

Como os processos de segmentação e *tokenization* foram realizados nos arquivos anotados, a rotulação IO desses arquivos de entrada retorna dois arquivos para cada sentença: um contendo os *tokens* e um contendo as *tags* correspondentes ao rótulo final esperado (classificação correta) de cada *token* no arquivo de treino.

Na próxima etapa, a LG construída neste trabalho é aplicada aos arquivos sem nenhuma marcação (sentenças tokenizadas) e as NEs identificadas por ela são anotadas. Um exemplo é apresentado na Figura 9.

<VALOR>Cerca de 400</VALOR> projetos em 30 países estão usando o sistema, afirma<PESSOA> Liebenberg</PESSOA>.

Figura 9 – Exemplo de aplicação da LG

A rotulação IO também é utilizada para converter as anotações das NEs atribuídas pela LG em uma sequência de rótulos.

Em seguida, várias características são adicionadas para cada *token* dos arquivos, como o *POS-Tagging*, se a palavra inicia com letra maiúscula ou não, inclusive o rótulo da NE atribuído pela LG anteriormente. Observe que os rótulos atribuídos pela LG (chamados neste trabalho de *tips*) e os rótulos dos arquivos de entrada (chamados aqui de *tags*) são independentes. *Tip* corresponde à classificação obtida através da LG e *tag* corresponde à classificação correta existente no conjunto de treino. Um exemplo dessa etapa é apresentado na Seção 5.2.

Os arquivos obtidos durante a adição de características e as *tags* são utilizados durante a aprendizagem supervisionada do modelo de predição CRF.

A metodologia usada para teste é semelhante. A diferença é que a etapa de rotulação IO para os arquivos de entrada não existe, pois eles não possuem as marcações das NEs. Além dos arquivos contendo os *tokens* e características, o CRF recebe o modelo treinado previamente para predizer um rótulo (*tag*) para cada *token*.

A Figura 10 apresenta uma visão geral da metodologia utilizada para teste.

As etapas foram executadas por um conjunto de *scripts* (Shell Script e Python). Todas as ferramentas necessárias para implementação da abordagem apresentada oferecem uma interface de linha de comando que foi utilizada. Algumas diretrizes para execução das ferramentas utilizadas e exemplos são apresentados no Apêndice B.

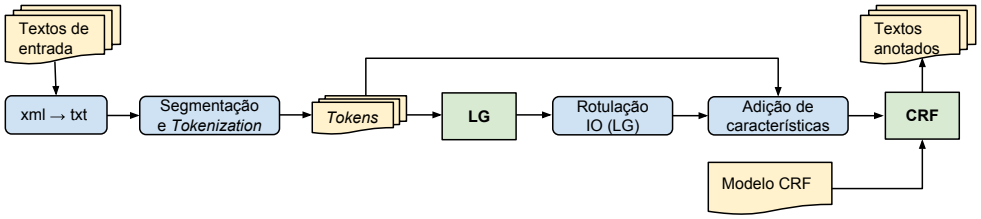


Figura 10 – Fluxograma da metodologia usada para teste

## 5.1 Construção da LG

A LG construída neste trabalho é constituída por 10 LGGs principais, um para cada categoria de NE considerada pelo HAREM. Cada LGG invoca outros LGGs representando regras para identificar NEs.

Para construir cada grafo, foram observadas as evidências internas e externas no conjunto de treino para auxiliar na identificação e classificação das NEs. As evidências internas aparecem dentro da NE, pode ser uma palavra que pertence a ela ou o seu padrão. Já as evidências externas fazem parte do contexto da NE, são palavras que aparecem à direita ou à esquerda dela e que poderiam indicar de alguma forma a existência da NE (FRIBURGER; MAUREL, 2004).

Alguns exemplos de evidências internas observadas no *corpus* de treino foram:

- Palavras como *Universidade*, *Faculdade* e *Departamento* iniciando NEs da categoria Organização: ***Universidade*** *de Lisboa*, ***Departamento*** *de Matemática*.
- Abreviações e formas de tratamento nas NEs da categoria Pessoa: *Diogo Costa* ***V. T.*** *Pereira*, ***Sr.*** *José Fragelli*.
- Unidades de medida no fim das NEs da categoria Valor: *5.000* ***m2***, *3,3* ***kg***.

Alguns exemplos de evidências externas detectadas no *corpus* de treino foram:

- Preposição *em* antes de NEs da categoria Local: ***em*** *Lisboa*, ***Em*** *Dublin*.
- Palavras como *denominada*, *chamava* e *professor de* antes de NEs da categoria Abstração: ***denominada*** *eBook Iniciative*, ***professor de*** *Ética Económica e Política*.
- Palavras como *disse*, *afirma*, *aconselha* após NEs da categoria Pessoa: *Afonso Camões* ***afirma***, *doutora Kawas* ***aconselha***.

Foi observado que algumas palavras são consideradas evidência interna ou externa dependendo se inicia com letra maiúscula ou não. Um exemplo é a palavra *rua* que não faz

parte da NE quando está em minúsculas, mas faz parte quando inicia com letra maiúscula, conforme exemplo a seguir:

- Evidência interna: *... na velha casa da **Rua** 1º de Dezembro.*
- Evidência externa: *... na quadra 35 da **rua** Araújo Leite.*

No exemplo de evidência interna acima, *1º de Dezembro* poderia ser identificada pelo LGG da categoria Tempo e *Rua 1º de Dezembro* pelo da categoria Local. O modo de reconhecimento *Longest matches* do Unitex foi utilizada neste trabalho para reconhecer a sequência mais longa. Quando a ocorrência é identificada por mais de um LGG, o fato de escolher a sequência mais longa (contexto maior) indica também maior probabilidade de que a classificação seja correta, já que considera mais evidências (palavras) no reconhecimento.

Não foram observadas evidências internas ou externas para identificar NEs da categoria Coisa. O LGG criado para essa categoria reconhece apenas algumas palavras que foram anotadas várias vezes com essa categoria no *corpus* de treino como nomes de planetas e signos.

Assim, os LGGs criados capturam algumas heurísticas simples para reconhecimento de NEs identificadas no *corpus* de treino. Um exemplo de regra no grafo criado para a categoria Pessoa é apresentado na Figura 11.

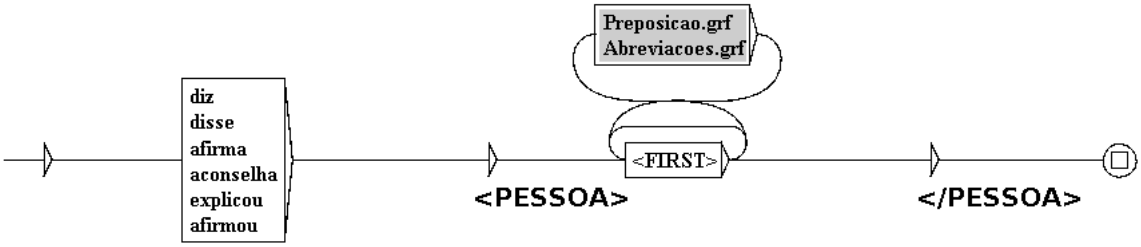


Figura 11 – Exemplo de regra no grafo que reconhece a categoria Pessoa

Esse grafo reconhece palavras como *diz* ou *afirmou* seguida de palavras que iniciam com letra maiúscula. Preposições e abreviações podem aparecer entre as palavras iniciadas com letra maiúscula. Exemplos de ocorrências identificadas por esse grafo foram:
*diz <PESSOA> Moncef Kaabi </PESSOA>*
*afirmou <PESSOA> José SÓCRATES</PESSOA>*
*afirma <PESSOA> Jason Knight </PESSOA>.*

Os nomes de pessoas identificados aparecem entre as *tags* <PESSOA> e </PESSOA> porque a opção *MERGE with input text* do Unitex foi utilizada. Assim, os arquivos obtidos após a aplicação da LG incluem as *tags* correspondentes às NEs identificadas por ela.

O Apêndice C apresenta apenas alguns exemplos de regras incluídas nos LGGs construídos neste trabalho. Descrever essas regras requer experiência humana e pode ser simples para linguistas, mas nem sempre esses profissionais são envolvidos na construção de sistemas de IE e NLP. É necessário buscar formas de auxiliar os profissionais da computação nessa tarefa. Por esse motivo, foi realizado o estudo apresentado no Capítulo 6.

## 5.2 CRF e Adição de características

A biblioteca MALLET [3] foi utilizada para inferir o modelo CRF de cadeia linear a partir do conjunto de treino e posteriormente aplicar esse modelo para rotular o conjunto de teste. A interface de linha de comando *SimpleTagger* foi utilizada. Para treinamento do modelo deve ser enviado um arquivo contendo, em cada linha, as características de um *token* e seu rótulo final esperado (*tag*), separados por espaço, sendo que esse rótulo deve estar na última coluna. Para testes (aplicação do modelo) deve ser enviado o arquivo contendo, em cada linha, as características de cada *token*.

As características são adicionadas ao arquivo contendo os *tokens* como já apresentado na Figura 6. As características usadas foram as mesmas propostas por (AMARAL, 2013), além da característica correspondente ao rótulo atribuído pela LG (*tip*). O conjunto de características é apresentado na Tabela 6.

Tabela 6 – Conjunto de características atribuídas a cada *token*

| **Características** | **Descrição** |
|---|---|
| word | palavra atual (posição p) |
| ptag | *POS-Tagging* da palavra correspondente à sua classe gramatical |
| cap | se a palavra é composta apenas de letras maiúsculas, apenas minúsculas ou maiúsculas e minúsculas |
| ini | se a palavra inicia por letra maiúscula, minúscula ou símbolos |
| simb | se a palavra é composta por símbolos, dígitos ou letras |
| preW, prevT, prevCap | *word, ptag* e *cap* para a palavra na posição p-1 |
| pre2W, prev2T, prev2Cap | *word, ptag* e *cap* para a palavra na posição p-2 |
| nextW, nextT, nextCap | *word, ptag* e *cap* para a palavra na posição p+1 |
| next2W, next2T, next2Cap | *word, ptag* e *cap* para a palavra na posição p+2 |
| tip | rótulo atribuído pela LG à palavra |

O *POS-Tagging* de uma palavra corresponde à sua classe gramatical e também foi

[3] http://mallet.cs.umass.edu/

atribuído pela biblioteca OpenNLP. Quando uma palavra não possui uma das palavras anteriores (p-1, p-2) ou posteriores (p+1, p+2), os valores das características correspondentes são *null*.

Um exemplo de atribuição de características ao *token* Liebenberg na sentença *Cerca de 400 projetos em 30 países estão usando o sistema, afirma Liebenberg* é apresentado na Tabela 7.

Tabela 7 – Exemplo de atribuição de características

| ***Token*** | **Características** | **Rotulação IO *tag*** |
|---|---|---|
| Liebenberg | word=Liebenberg ptag=n cap=maxmin ini=cap simb=alpha prevW=afirma prevT=v-fin prevCap=min nextW=. nextT=punc nextCap=null prev2W=, prev2T=punc prev2Cap=null next2W=null next2T=null next2Cap=null tip=I-PESSOA | I-PESSOA |

# 6 Comparação de Concordâncias para compor LGs

Neste capítulo é apresentado um estudo no qual a ferramenta de comparação de concordâncias do Unitex (Unitex, 2018), Concordiff, foi utilizada na composição de uma LG para identificar nomes de pessoas em textos em Português. Um repositório de LGGs que representam regras mais simples foi utilizado com esse objetivo. Analisando as concordâncias identificadas por diferentes LGGs, foram observadas algumas relações da teoria de conjuntos e foi observado que é possível tomar algumas decisões de acordo com essas relações que podem ajudar a obter uma LG maior e mais completa a partir de gramáticas menores (elementares).

A comparação de concordâncias proporcionou uma visão geral do que cada LGG reconhece em um *corpus* específico, permitiu identificar ambiguidades e falso-positivos e foi essencial para obter a LG final.

O estudo apresentado neste capítulo foi realizado com o objetivo de compreender melhor a construção de LGs e sua composição, para que, posteriormente, uma LG pudesse ser construída para as 10 categorias do HAREM conforme proposto na Seção 5.1. Logo, esse estudo explora apenas a combinação de vários LGGs que reconhecem expressões da mesma categoria (Pessoa). Outros tipos de combinações podem ser explorados em outros estudos.

O método descrito aqui propõe um auxílio na composição de LGs, mas como a construção de LGs requer *expertise* humana, é importante considerar a participação de linguistas para construir gramáticas locais mais complexas e de qualidade.

## 6.1 O Programa Concordiff do Unitex

O programa ConcorDiff (PAUMIER, 2016) do Unitex compara dois arquivos de concordância, correspondentes às listas de ocorrências identificadas por dois grafos em um texto de entrada, linha a linha e apresenta suas diferenças. O resultado é uma página HTML que alterna linhas das duas concordâncias e deixa uma linha vazia quando uma ocorrência aparece em apenas uma delas. Um exemplo de comparação de concordâncias é apresentado na Figura 12. As linhas com cor de fundo rosa (primeira, terceira, quinta e sétima linhas) são do primeiro arquivo de concordância passado como parâmetro para o programa e as com cor de fundo verde (segunda, quarta e sexta linhas) do segundo.

As linhas escritas em azul (primeira e segunda linhas) indicam ocorrências comuns

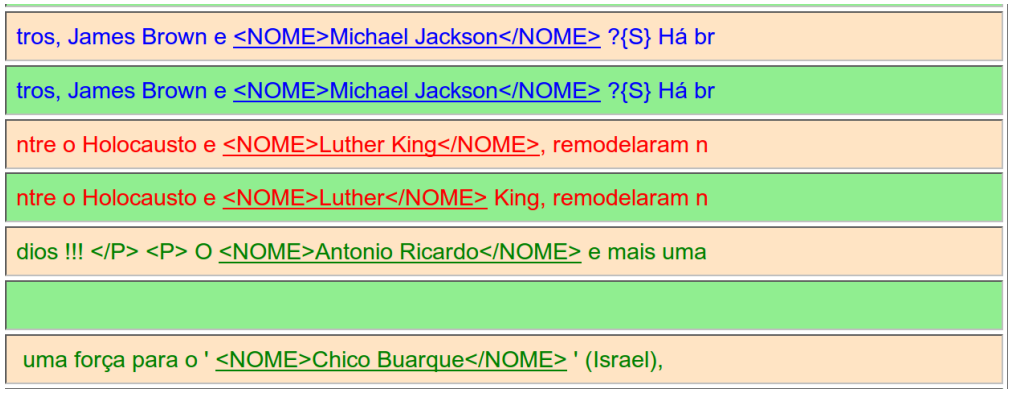


Figura 12 – Exemplo de comparação de concordâncias gerado pelo Unitex

às duas concordâncias. Para o exemplo da Figura 12 significa que *Michael Jackson* foi reconhecido pelos dois LGGs cujas concordâncias foram comparadas. As linhas vermelhas (terceira e quarta linhas) indicam que as ocorrências se sobrepõem parcialmente, ou seja, possuem uma parte comum. No exemplo, um LGG reconheceu *Luther King* como nome e o outro reconheceu apenas *Luther*. As linhas verdes (quinta e sétima linhas) indicam ocorrências que pertencem a somente uma concordância. *Antonio Ricardo* e *Chico Buarque* foram reconhecidos como nomes apenas pelo LGG que gerou o primeiro arquivo de concordância passado para o programa. Embora não exemplificada nessa figura, a cor violeta pode aparecer indicando ocorrências idênticas, mas com saídas (rótulos) diferentes anexadas ao arquivo de concordância.

## 6.2 Metodologia

Inicialmente foi construído um repositório de LGGs criados no Unitex para reconhecer nomes de pessoas. Algumas regras usadas para construir os LGGs desse repositório foram obtidas na literatura e outras foram criadas durante este trabalho. Os LGGs do repositório construído são considerados elementares pois serão usados para compor uma gramática maior para identificar nomes de pessoas. Assim, a gramática local para identificar nomes de pessoas, denotada simplesmente por LG, será constituída por vários LGGs ligados por invocações.

Após a construção desse repositório, os LGGs elementares foram aplicados à Coleção Dourada (CD) do Segundo HAREM gerando um arquivo de concordância para cada LGG. Foram aplicados os dicionários de Português e Inglês pois vários nomes da língua inglesa aparecem nos textos da CD.

As concordâncias obtidas após a aplicação dos LGGs à CD do Segundo Harem foram comparadas, duas a duas, utilizando a comparação de concordâncias realizada pelo programa ConcorDiff do Unitex.

Em seguida, foi realizada a análise dos arquivos gerados pelo ConcorDiff e a composição da LG para reconhecer nomes de pessoas a partir dos LGGs elementares conforme metodologia apresentada na Seção 6.3. A LG é composta por LGGs do repositório inicial por meio de modificações manuais: revisão dos grafos, das invocações entre grafos e inserção de novos grafos.

## 6.3 Composição da LG

Sejam $G_X$ e $G_Y$ dois LGGs e $C_X$ e $C_Y$ os respectivos arquivos de concordância obtidos ao aplicá-los ao mesmo *corpus*. Ou seja, $C_X$ é o arquivo com as ocorrências identificadas por $G_X$ e $C_Y$ o arquivo com as ocorrências identificadas por $G_Y$. $C_X$ x $C_Y$ é o arquivo apresentando as diferenças entre as concordâncias $C_X$ e $C_Y$, obtido pelo programa ConcorDiff do Unitex. Sejam $x_1$, $x_2$, ..., $x_n$ os elementos de $C_X$ apresentados nas linhas com cor de fundo rosa em $C_X$ x $C_Y$ e $y_1$, $y_2$, ..., $y_m$ os elementos de $C_Y$ apresentados nas linhas com cor de fundo verde. Como $C_X$ e $C_Y$ representam conjuntos de ocorrências, é possível observar algumas relações da teoria de conjuntos ao analisar $C_X$ x $C_Y$.

Sejam os LGGs $G_1$ e $G_2$ apresentados nas Figuras 13 e 14.

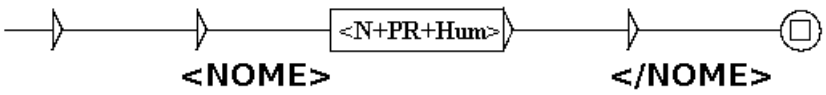


Figura 13 – LGG $G_1$ (ReconheceNomesCompostos.grf)


Figura 14 – LGG $G_2$ (ReconheceFormasDeTratamento.grf)

$G_1$ reconhece substantivos próprios utilizando o código do Unitex <N+PR> com o código semântico <Hum> para se referir a humano. Nomes compostos de pessoas conhecidas como *Marilyn Monroe, Cameron Diaz* e *Albert Einstein* aparecem com esse código ao aplicar o dicionário de Inglês no texto de entrada. Já $G_2$ reconhece nomes precedidos por formas de tratamento como *Sra.*, *D.* e *Sr.* O «..» após <FIRST> indica

a aplicação de um filtro morfológico sobre as palavras iniciadas com letras maiúsculas indicando que elas devem ter pelo menos dois caracteres. Isso evita o reconhecimento de preposições no início de frase por exemplo.

A Figura 15 apresenta parte da comparação de concordâncias $C_1$ x $C_2$. Observe que $y_1$, a primeira linha com fundo verde de $C_1$ x $C_2$ (primeira linha na figura), contém o nome *Manuel* após *D.* reconhecido por $G_2$ . Já $x_2$, a segunda linha com fundo rosa de $C_1$ x $C_2$ (última linha na figura), contém o nome *Ray Bradbury* reconhecido por $G_1$.

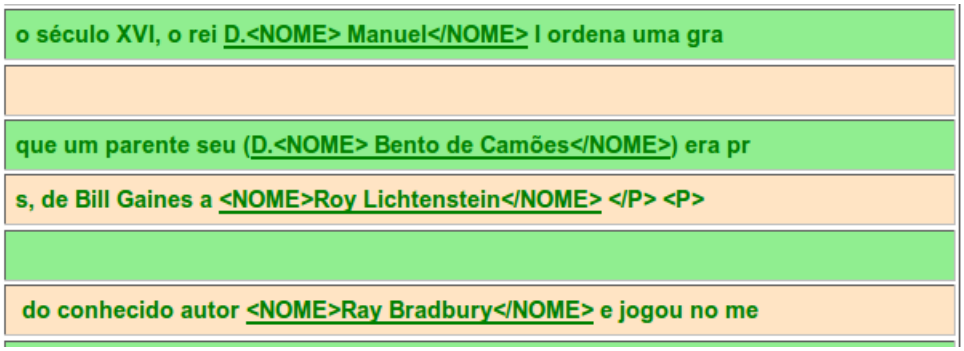


Figura 15 – Parte da Comparação de Concordâncias $C_1$ x $C_2$

Linhas verdes com cor de fundo diferentes indicam ocorrências identificadas por apenas um dos dois grafos. As duas primeiras ocorrências foram identificadas por $G_2$ e as duas últimas por $G_1$. Embora apenas parte da comparação tenha sido apresentada, todas as linhas dela são verdes com cor de fundo diferentes, indicando que as duas regras reconhecem nomes diferentes no *corpus*. Pode-se dizer assim que $C_1$ e $C_2$ são conjuntos **disjuntos** e que é necessário manter os 2 LGGs $G_1$ e $G_2$ como subgrafos em uma gramática com o objetivo de reconhecer nomes de pessoas pois elas reconhecem nomes diferentes. A gramática da Figura 16 mostra a união de $G_1$ e $G_2$.

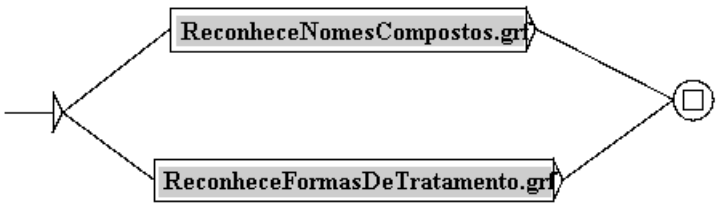


Figura 16 – Grafo composto por $G_1$ e $G_2$

Se apenas uma cor de fundo ocorrer em um arquivo que só tem linhas verdes significa que um dos LGGs não reconheceu nada. Assim, apenas o LGG que identificou alguma ocorrência é relevante e deve ser mantido.

Uma relação óbvia é a de igualdade de conjuntos que pode ser observada, por exemplo, ao analisar a comparação de duas concordâncias obtidas pelo mesmo LGG. O arquivo $C_X$ x $C_X$ terá apenas linhas azuis indicando que todas as ocorrências são comuns às duas concordâncias. Observe a Figura 17 que apresenta parte da comparação de concordâncias $C_1$ x $C_1$.

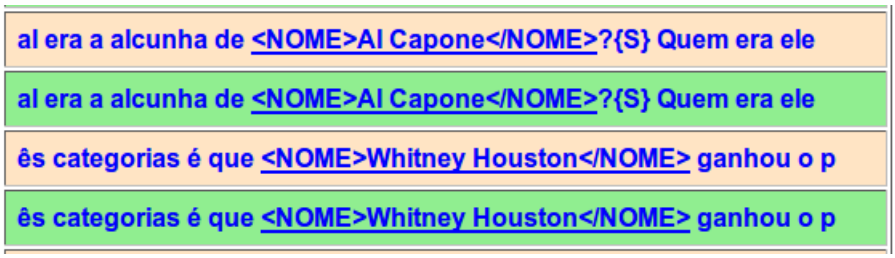


Figura 17 – Parte da Comparação de Concordâncias $C_1$ x $C_1$

O LGG $G_3$ apresentado na Figura 18 reconhece aposto especificativo. Através dele é possível reconhecer *o engenheiro João da Silva* e *a bela Maria Gabriela*. Os códigos <DET> e <A> são utilizados nos dicionários do Unitex indicando determinante e adjetivo, respectivamente. O * verde indica que o contexto à esquerda será usado para extrair ocorrências, mas não fará parte delas. Já o $G_4$ (Figura 19) reconhece nomes precedidos por verbos que se referem a ações humanas como *disse*, *afirmou* e *aconselhou* seguidos por um determinante e uma palavra escrita com letras minúsculas (código <LOWER> no Unitex).

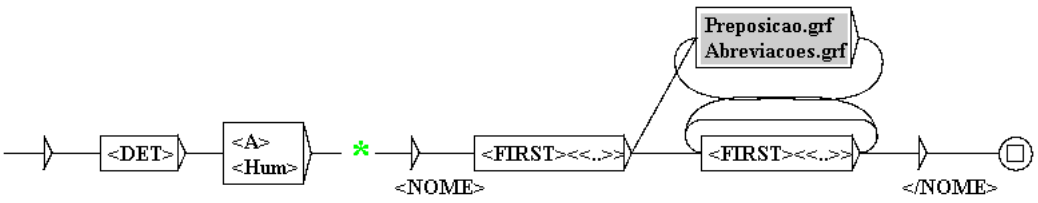


Figura 18 – LGG $G_3$ (ReconheceAposto.grf)

A Figura 20 apresenta parte da comparação $C_3$ x $C_4$. Observe que $x_2$ (segunda linha com fundo rosa e terceira na figura) e $y_1$ (primeira linha com fundo verde e quarta na figura) reconhecem o mesmo nome *Brian Holmes*, porém $G_3$ reconhece *o teórico* antes do nome e $G_4$ reconhece *diz o teórico* antes do nome. Essas palavras não são apresentadas devido o contexto indicado por *.

Todas as outras linhas da comparação $C_3$ x $C_4$ são semelhantes. Linhas azuis e verdes com mesma cor de fundo indicam que algumas ocorrências são idênticas em $C_3$ e $C_4$, mas existem ocorrências que aparecem em somente uma delas. Nesse caso, o LGG $G_3$ reconhece tudo que $G_4$ reconhece (em azul) mais alguma coisa (em verde com fundo

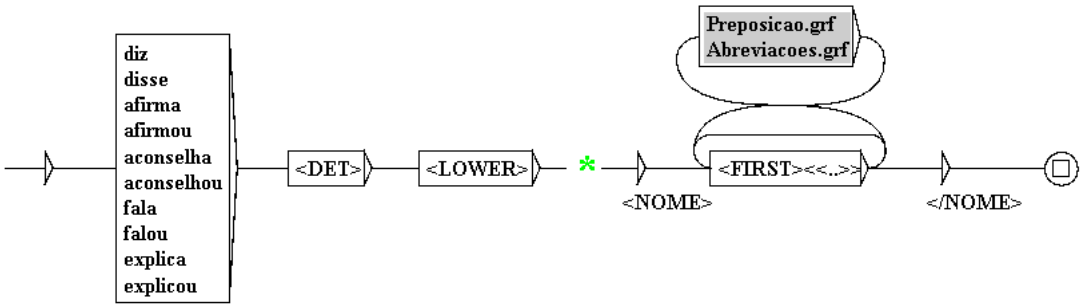


Figura 19 – LGG $G_4$ (ReconheceAcaoSeguidaAposto.grf)

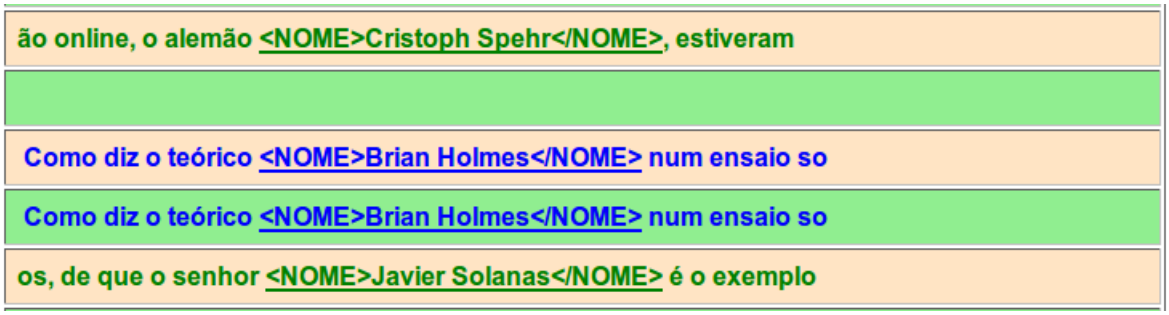


Figura 20 – Parte da Comparação de Concordâncias $C_3$ x $C_4$

rosa). Sendo assim, pode-se dizer que o conjunto $C_4$ está **incluído** em $C_3$ e apenas o LGG $G_3$ pode ser mantido na LG final. Também pode acontecer o contrário, $C_X$ **incluído** em $C_Y$. Nesse caso, a cor de fundo das linhas verdes também será verde e $G_Y$ deve ser mantido na LG final.

Uma situação um pouco diferente pode ser observada ao comparar as concordâncias obtidas a partir dos LGGs $G_5$ e $G_6$.

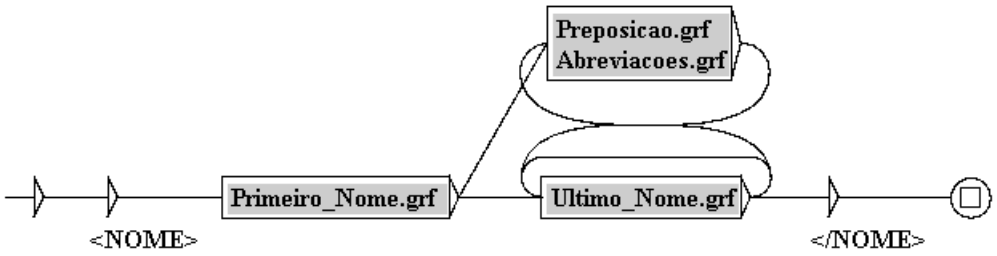


Figura 21 – LGG $G_5$ (Reconhece2NomesProprios.grf)

$G_5$ (Figura 21) utiliza dois subgrafos Primeiro_Nome.grf e Ultimo_Nome.grf para reconhecer palavras consecutivas iniciando com letra maiúscula. A expressão reconhecida por Ultimo_Nome.grf pode ser repetida diversas vezes. A Figura 22 descreve o reconhecimento do primeiro nome. Primeiro_Nome.grf reconhece um substantivo próprio

(<N+Pr>) desde que apenas a primeira letra da palavra seja maiúscula. Ultimo_Nome.grf é semelhante ao LGG da Figura 22, mas não verifica se a palavra que começa com letra maiúscula é substantivo próprio, ou seja, aceita palavras que não estão no dicionário. Isso significa que o primeiro nome sempre deve ser um substantivo próprio que consta no dicionário, mas os sobrenomes podem ser palavras desconhecidas.


Figura 22 – Subgrafo Primeiro_Nome.grf usado no LGG $G_5$ (Figura 21)

$G_6$ (Figura 23) é semelhante ao LGG da Figura 19, porém o nome é precedido apenas por verbos que se referem a ações humanas.

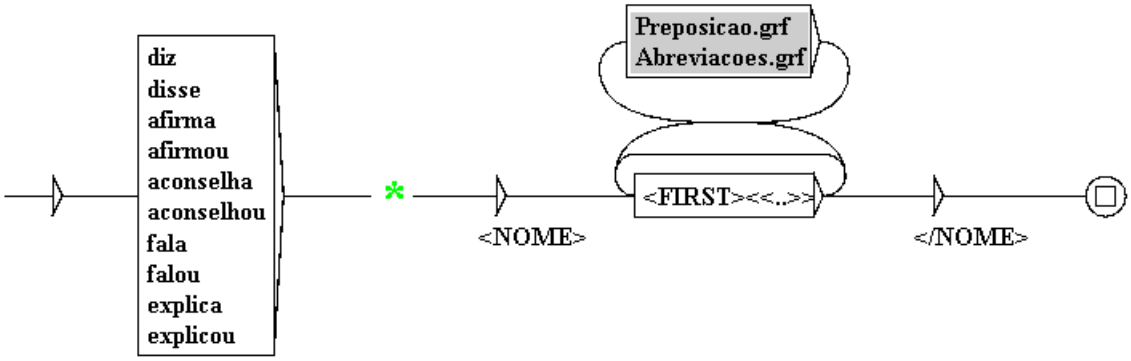


Figura 23 – LGG $G_6$ (ReconheceAcoesHumanasAEsquerda.grf)

A Figura 24 apresenta parte da comparação $C_5$ x $C_6$. $y_1$ (primeira linha na figura) apresenta o reconhecimento do nome *Bjornstjerne Christiansen* identificado por $G_6$ pois aparece após o verbo que representa ação humana *afirmou*, mas não é identificado por $G_5$ pois a palavra *Bjornstjerne* não consta no dicionário. A diferença principal nessa comparação de concordâncias com relação à $C_3$ x $C_4$ são as linhas verdes com cor de fundo diferentes como a primeira e segunda linhas. Isso indica que existe uma pequena **interseção** entre os conjuntos $C_5$ e $C_6$ (em azul), mas existem ocorrências distintas em cada conjunto (linhas verdes) e, por isso, os dois LGGs devem ser mantidos na LG final.

Sempre que aparecerem linhas verdes com cor de fundo diferentes em $C_X$ x $C_Y$, ambos $G_X$ e $G_Y$ devem ser mantidos na LG final pois existem ocorrências exclusivas a cada uma das concordâncias.

Considere agora o LGG $G_3$ que reconhece aposto novamente (Figura 18) e o LGG $G_7$ que reconhece a mesma estrutura sintática, porém sem o * que indica o fim do contexto


Figura 24 – Parte da Comparação de Concordâncias $C_5$ x $C_6$

a esquerda (Figura 25). Nesse caso, a parte do contexto à esquerda (<DET> seguido de <A> ou <Hum>) fará parte da ocorrência identificada. Observe a comparação $C_3$ x $C_7$ apresentada na Figura 26.

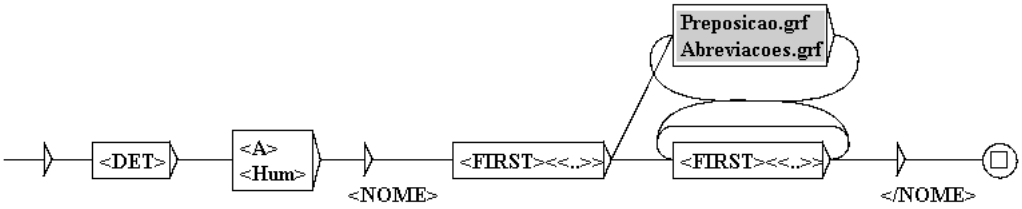


Figura 25 – LGG $G_7$ (ReconheceAposto1.grf)

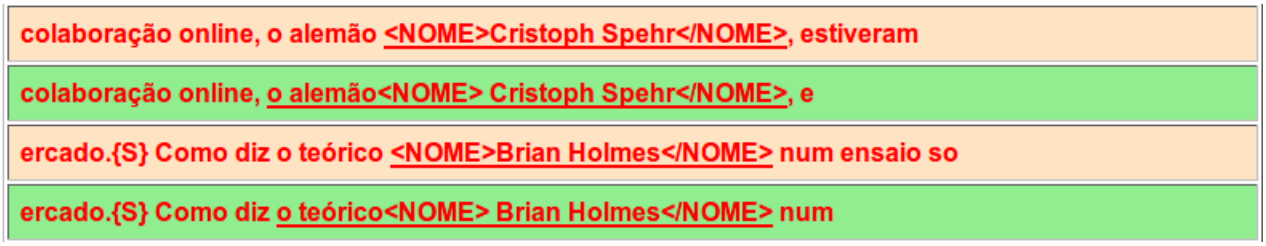


Figura 26 – Parte da Comparação de Concordâncias $C_3$ x $C_7$

Observe que cada ocorrência x[1] de $C_3$ está emparelhada com uma ocorrência y de $C_7$ na Figura 26. $x_1$ (primeira linha na figura) apresenta o reconhecimento de *<NOME>Cristoph Spehr<NOME>* que aparece sublinhado na figura, enquanto $y_1$ (segunda linha na figura) apresenta o reconhecimento de *o alemão <NOME>Cristoph Spehr <NOME>*. Os dois LGGs identificaram o nome *Cristoph Spehr*, embora o LGG $G_7$ tenha mantido o contexto usado para identificar o nome na ocorrência. Ou seja, cada ocorrência

[1] Os índices para x e y somente serão usados quando for necessário se referir a alguma ocorrência específica da concordância

y identificada pelo LGG $G_7$ é mais longa do que as ocorrências x emparelhadas a ela identificadas pelo LGG $G_3$.

Assim como as demais linhas nessa comparação de concordância, todas as linhas na Figura 26 são vermelhas, indicando a ocorrência de uma **disjunção com sobreposição parcial de ocorrências**. Nesse caso, todas as ocorrências em $C_3$ e $C_7$ foram emparelhadas, ou seja, são similares e somente se sobrepõem (*overlap*) parcialmente. Essa sobreposição pode ser à direita ou à esquerda, indicando que em alguma das concordâncias é capturado algum contexto à direita ou à esquerda. No exemplo $C_3$ x $C_7$, a sobreposição acontece à direita das ocorrências, indicando que o contexto à esquerda é capturado pelo $G_7$.

Quando todas as linhas de $C_X$ x $C_Y$ forem vermelhas, a decisão será manter na LG final aquela que extrai as ocorrências mais longas. Essa possibilidade é importante, pois caso a ocorrência seja identificada por mais de um LGG, o fato de escolher o que reconhece a sequência mais longa (contexto maior) indica também maior probabilidade de que a ocorrência seja realmente um nome, já que considera mais evidências (palavras) no reconhecimento. Isso também pode auxiliar na desambiguação da entidade nomeada caso o sistema reconheça outros tipos além de nomes de pessoas. Para reconhecer a sequência mais longa, o modo de reconhecimento *default* do Unitex (*Longest matches*) foi utilizado neste trabalho.

Também podem aparecer linhas vermelhas com linhas de outras cores em $C_X$ x $C_Y$. Observe a concordância $C_5$ x $C_8$ apresentada na Figura 28. Ela foi obtida após a aplicação de $G_5$ já apresentado na Figura 21 e de $G_8$ apresentado na Figura 27. Esse reconhece nomes sucedidos por verbos que se referem a ações humanas como *disse*, *afirmou*, *aconselhou*, etc. Existem algumas ocorrências em $C_5$ x $C_8$ que somente se sobrepõem (*overlap*) parcialmente como a primeira e segunda linhas (em vermelho) e outras ocorrências identificadas apenas por $G_5$ como a terceira e última linha (em verde com cor de fundo rosa).

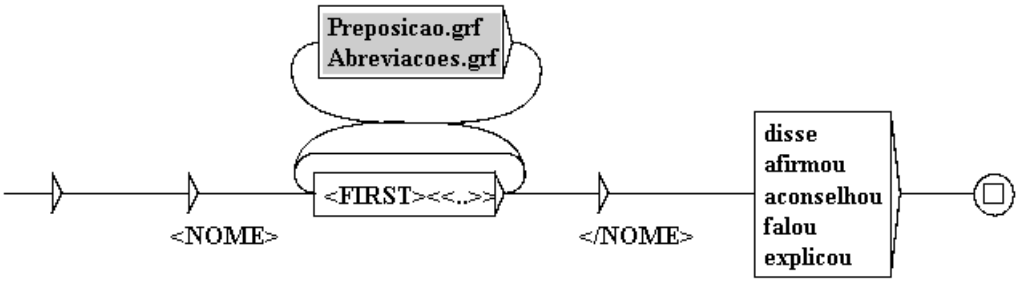


Figura 27 – LGG $G_8$ (ReconheceAcoesHumanasADireita.grf)

Nesse caso é necessário manter os dois LGGs pois as ocorrências identificadas apenas por $G_5$ são relevantes, embora um falso-positivo (*Nobel da Medicina*) tenha sido reconhecido. Portanto, se as linhas de $C_X$ x $C_Y$ são vermelhas e verdes com a mesma cor

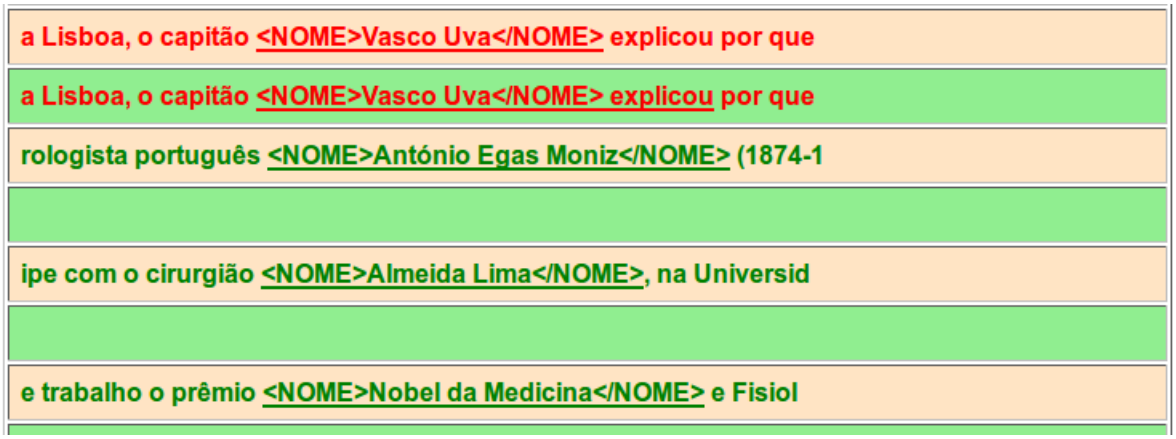


Figura 28 – Parte da Comparação de Concordâncias $C_5$ x $C_8$

de fundo, ambos LGGs devem ser mantidos se as ocorrências em verde são relevantes. Caso contrário, é necessário manter apenas a que captura o maior contexto.

Como foi possível observar nesse exemplo, a comparação de concordâncias também auxilia a identificar erros. Para evitar o reconhecimento de palavras que não são nomes (falso-positivos) como *Nobel da Medicina* em $C_5$ x $C_8$, podem ser incluídos LGGs para reconhecer outras entidades nomeadas. Um exemplo é o LGG $G_9$ apresentado na Figura 29 que reconhece palavras iniciadas com letras maiúsculas precedidas pela palavra *Rua* como <LOCAL> ao invés de <NOME>.

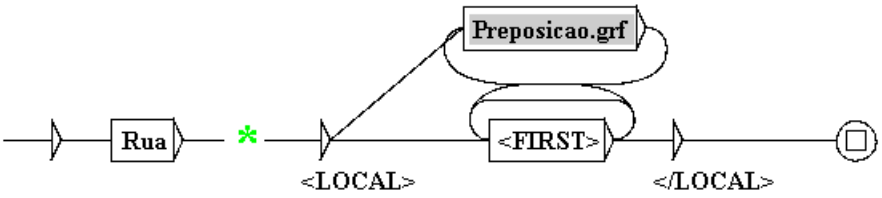


Figura 29 – LGG $G_9$ (ReconheceNomesRua.grf)

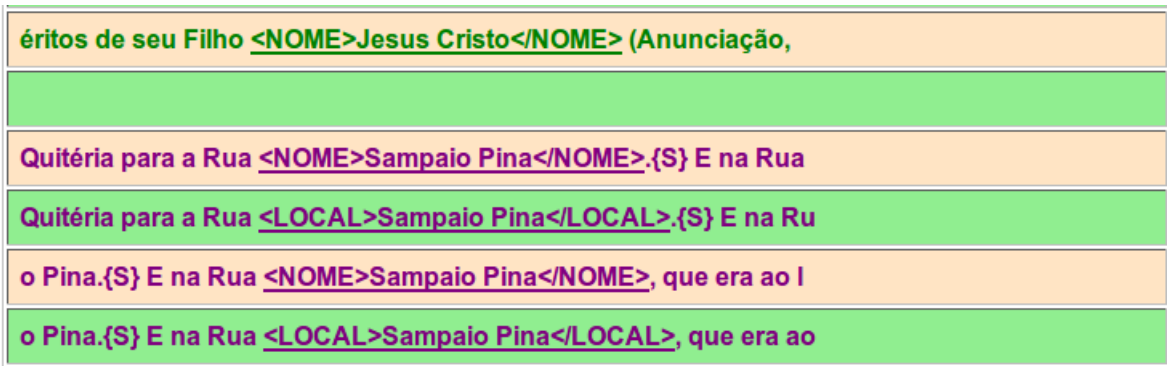


Figura 30 – Parte da Comparação de Concordâncias $C_5$ x $C_9$

Observe que os nomes de Rua também podem ser reconhecidos por $G_5$ (Figura 21) e existe uma ambiguidade para resolver. Na comparação de concordâncias $C_5$ x $C_9$ (Figura 30) *Sampaio Pina* é reconhecido como nome de pessoa e também como local (terceira e quarta linhas). Os dois LGGs deveriam ser mantidos na LG final, já que o $G_5$ reconhece outros nomes de pessoas não identificados por $G_9$, mas a ambiguidade deve ser resolvida posteriormente com esse resultado do Unitex. Sempre que a cor violeta aparecer em uma comparação de concordâncias onde também aparecem outras cores, os dois LGGs devem ser mantidos e a ambiguidade tratada após o uso do Unitex.

Uma forma de resolver a ambiguidade é rotular primeiro os locais e depois os nomes de pessoas. Assim, quando for realizada a anotação dos nomes de pessoas, *Sampaio Pina* já teria sido rotulada como Local e não seria considerado um nome de pessoa, ou seja, não seria anotada novamente.

Tabela 8 – Relações observadas através da Comparação de Concordâncias

| Relação | Representação | Cor das linhas | Decisão |
|---|---|---|---|
| **Inclusão** | $C_X \subset C_Y$ | Azuis e verdes (com cor de fundo verde) | Manter $G_Y$ |
| | $C_Y \subset C_X$ | Azuis e verdes (com cor de fundo rosa) | Manter $G_X$ |
| **Interseção** | $C_X = C_Y$ | Azuis | Manter ou $G_X$ ou $G_Y$ |
| | $C_X = C_Y$ com saídas diferentes | Violeta | Analisar ambiguidade |
| | $C_X \cap C_Y \neq \emptyset$ | Azuis e verdes (com cor de fundo diferentes) | Manter $G_X$ e $G_Y$ |
| **Disjunção** | $C_X \cap C_Y = \emptyset$, com $C_X = \emptyset$ | Verdes (com cor de fundo verde) | Manter $G_Y$ |
| | $C_X \cap C_Y = \emptyset$, com $C_Y = \emptyset$ | Verdes (com cor de fundo rosa) | Manter $G_X$ |
| | $C_X \cap C_Y = \emptyset$ | Verdes (com cor de fundo diferentes) | Manter $G_X$ e $G_Y$ |
| **Disjunção com sobreposição parcial de ocorrências** | $C_X \cap C_Y = \emptyset$, com $C_X \sim C_Y$[1] | Vermelhas | Manter $G_X$ se $\|x_{iv}\| > \|x_{jv}\|$[2], ∀i e ∀j ou $G_Y$ se $\|y_{jv}\| > \|x_{iv}\|$, ∀i e ∀j |
| | $C_X \cap C_Y = \emptyset$, com ($\exists\ x_i \in C_X$) ($\exists\ y_j \in C_Y$), $\|x_{iv}\| \sim \|y_{jv}\|$ | Vermelhas e verdes com mesma cor de fundo | Manter $G_X$ e $G_Y$ se as ocorrências em verde são relevantes. Se não, manter apenas a que captura maior contexto. |

[1] $C_X \sim C_Y \Leftrightarrow |x_{iv}| \sim |y_{jv}|$, ∀i e ∀j. $|x_{iv}| \sim |x_{jv}|$ significa que $x_i$ está emparelhada, ou simplesmente é similar, a $x_j$.
[2] $|x_{iv}| > |x_{jv}|$ significa que $x_i$ é mais longa do que a $x_y$ emparelhada com ela.

Se o arquivo $C_X$ x $C_Y$ apresenta apenas linhas na cor violeta, significa que $G_X$ e $G_Y$ reconheceram as mesmas ocorrências, mas anexaram saídas diferentes a elas. Nesse

caso, todas as ocorrências são ambíguas e essa ambiguidade deve ser tratada analisando melhor a situação.

A Tabela 8 resume as relações da teoria de conjuntos identificadas.

Os LGGs utilizados nos exemplos são muito simples, mas poderiam ser complexos. A Tabela 8 possibilita que um usuário tome suas decisões analisando as concordâncias independente de quais sejam os LGGs, sem a necessidade de conhecê-los ou de um entendimento profundo dos mesmos. As decisões indicadas nessa tabela também podem ser utilizadas no desenvolvimento de ferramentas computacionais que utilizem a comparação de concordâncias.

## 6.4 Resultados da Composição da LG

A gramática local para reconhecer nomes de pessoas obtida após uso da comparação de concordâncias pode ser vista na Figura 31. Os resultados obtidos ao aplicar essa LG na CD do Segundo HAREM foram 59.06% de Precisão, 55.22% de Abrangência e 57.07% de Medida-F.

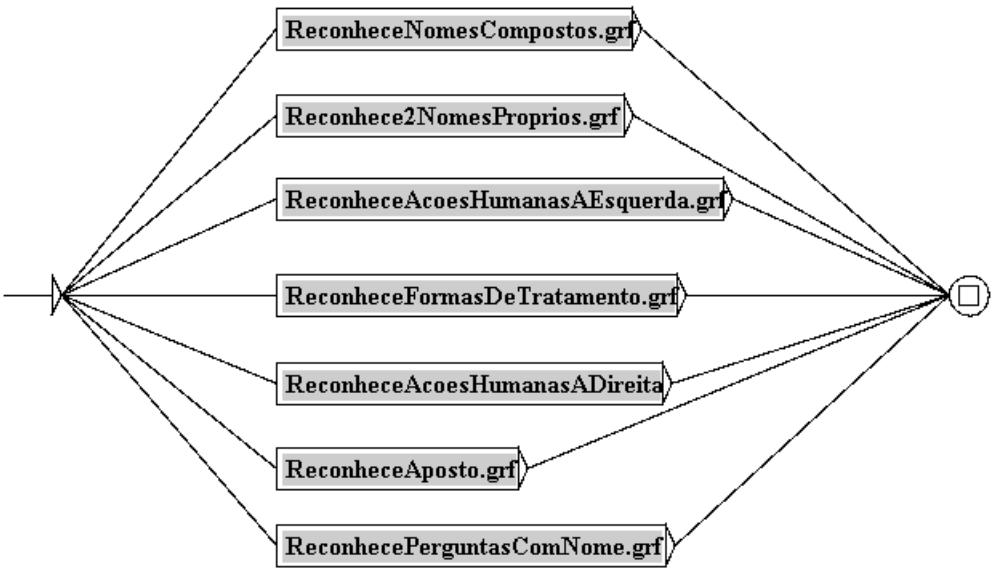


Figura 31 – LG para reconhecer nomes de pessoas

Algumas regras contribuem mais, outras menos, para esse resultado. Por exemplo, a regra representada pelo LGG da Figura 21 (Reconhece2NomesProprios.grf) é responsável pela identificação da maior parte dos nomes de pessoas. Ela identifica cerca de 45% do total de nomes de pessoas no *corpus* corretamente (Abrangência). Porém, outras regras apresentam uma quantidade menor de erros (falso-positivos). A regra apresentada na Figura 3 (ReconhecePerguntasComNomes.grf) identifica apenas 3% dos nomes de pessoas do *corpus*, mas tem precisão alta, cerca de 90% de acerto no total de nomes identificados.

Várias estratégias foram utilizadas para melhorar o desempenho da LG. No Segundo HAREM, algumas palavras ou expressões minúsculas devem fazer parte da NE[2]. Um exemplo são as formas de tratamento reconhecidas pelo LGG da Figura 14 e nomes de cargos ou profissões que aparecem antes de nomes de pessoas e devem fazer parte da NE. Um exemplo disponibilizado pelo HAREM[3] mostra que na frase *A rainha Isabel II surpreeendeu a Inglaterra* não basta anotar o nome *Isabel* como nome de pessoa, mas *rainha Isabel II*.

Para resolver essa questão, o LGG ReconheceFormasDeTratamento.grf (Figura 14) foi alterado simplesmente mudando a posição da saída (rótulo) no grafo para que a forma de tratamento seja parte da NE reconhecida. Ela aparece agora antes da forma de tratamento como mostra a Figura 32. Se LG for utilizada em outra aplicação cujo interesse seja apenas o nome de pessoa sem a forma de tratamento, basta retornar a saída para posição original. Além disso, foi criado um LGG que reconhece as palavras minúsculas disponibilizadas pelo HAREM como contexto à esquerda do nome de pessoa (ReconheceContextoAEsquerda.grf).

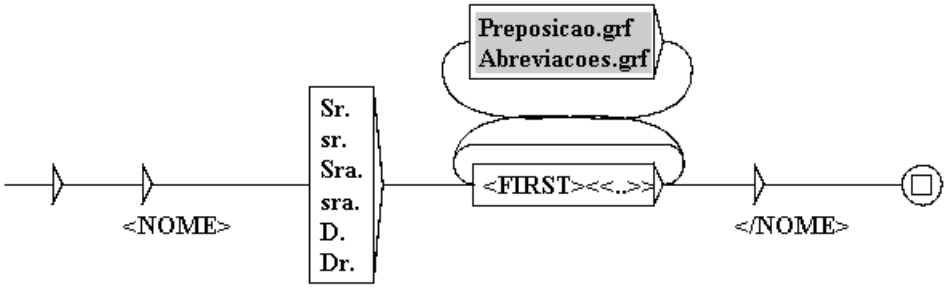


Figura 32 – Alteração realizada no LGG ReconheceFormasDeTratamento.grf

Essas palavras minúsculas também foram utilizadas para reconhecer o tipo cargo da classe Pessoa, representado por PESSOA(CARGO) e para reconhecer nomes de pessoas onde um cargo aparece no contexto à esquerda. Um nome reconhecido dessa forma na CD do Segundo HAREM foi *Nuno Severiano Teixeira* em *Ministro da Defesa Nacional, Nuno Severiano Teixeira*. Reconhecer o tipo cargo foi importante para desambiguar algumas entidades nomeadas rotuladas inicialmente como nome de pessoa e que na verdade se referiam a um cargo.

Como o LGG Reconhece2NomesProprios.grf (Figura 21) reconhece palavras iniciando com letras maiúsculas como nomes de pessoas sem considerar o contexto, em um *corpus* como a CD do Segundo HAREM que possui uma grande diversidade de NEs, muitos falso-positivos foram encontrados. Assim, foram usadas algumas heurísticas simples para desambiguar nomes de pessoas com outras NEs comuns. Por exemplo, NEs antecedidas

[2] http://www.linguateca.pt/aval_conjunta/HAREM/minusculas.html
[3] http://www.linguateca.pt/aval_conjunta/HAREM/ExemplarioSegundoHAREM.pdf

por palavras como *em, no, na, Rua, Praça, Avenida, etc* foram rotuladas como Local; e NEs antecedidas por *da, pela, Universidade, Faculdade, Organização, etc* foram rotuladas como Organização.

Analisando os falso-positivos e falso-negativos, várias outras regras foram identificadas e adicionadas a LG. Uma delas foi uma regra para reconhecer nomes antecedidos por verbo seguido da preposição *por* (ReconheceAcaoRealizadaPor.grf). Um exemplo foi o reconhecimento do nome *Francisco de Almeida* em *comandada por Francisco de Almeida*.

Os resultados obtidos pela LG final para as métricas Precisão (P), Abrangência (A) e Medida-F (F) são apresentados na Tabela 9. Nessa tabela também são apresentados os resultados obtidos pelo Rembrandt (CARDOSO, 2008), sistema que obteve melhor desempenho no reconhecimento da categoria Pessoa no Segundo HAREM. Embora o Rembrandt classifique as 10 categorias do HAREM, o resultado apresentado nessa tabela considera só o tipo individual da classe Pessoa.

Tabela 9 – Comparação: Rembrandt x LG

| **Sistemas** | **P (%)** | **A (%)** | **F (%)** |
|---|---|---|---|
| Rembrandt | 79 | 64.08 | 70.76 |
| LG | 79.75 | 74.18 | 76.86 |

Observe na Tabela 9 que LG supera os resultados do Rembrandt. Os resultados da métrica Abrangência indicam que ela reconhece corretamente cerca de 10% de nomes de pessoas a mais que o Rembrandt.

Embora LG reconheça apenas os tipos individual e cargo da categoria Pessoa, sua avaliação também foi realizada para todos os 8 tipos da categoria. A comparação dos resultados obtidos com os resultados apresentados em Amaral et al. (2014) para as quatro ferramentas é mostrada na Tabela 10. A tabela apresenta os resultados obtidos pelas quatro ferramentas, seguido pelo resultado da LG proposta neste trabalho.

Tabela 10 – Comparação: Sistemas em Amaral et al. (2014) x LG

| **Sistemas** | **P (%)** | **A (%)** | **F (%)** |
|---|---|---|---|
| NERP-CRF | 57 | 51 | 54 |
| Freeling | 55 | 61 | 58 |
| Language-Tasks | 63 | 62 | 62 |
| PALAVRAS | 61 | 65 | 63 |
| LG | 81.28 | 60.36 | 69.28 |

LG apresentou maior Precisão pois identifica menos tipos de NEs. A abrangência obtida é superior à do sistema NERP-CRF apenas. Ainda assim, esse pode ser considerado um resultado promissor, considerando que LG reconhece só dois tipos da categoria Pessoa. Acredita-se que, a adição de regras a LG para reconhecer os outros tipos da categoria Pessoa elevaria a abrangência do sistema, tornando LG superior às demais ferramentas.

Foi observado um problema que influencia os resultados ao analisar os falso-positivos. LG identifica *José Mourinho* como nome de pessoa em *Liderança - As Lições de José Mourinho*, porém o HAREM classifica *Liderança - As Lições de José Mourinho* como Obra. LG tem como objetivo identificar nomes de pessoas e, em uma aplicação com essa finalidade, reconhecer *José Mourinho* como nome de pessoa estaria correto, mas é contabilizado como erro ao computar as métricas. O mesmo acontece com *Adolf Hitler* em *as tropas de Adolf Hitler* que é considerado Organização pelo Segundo HAREM e em todas as situações onde um nome de pessoa aparece no contexto de outras categorias identificadas pelo Segundo HAREM.

# 7 Experimentos e Resultados

Neste capítulo são apresentados todos os experimentos realizados e resultados obtidos durante este trabalho. A LG final, obtida no experimento do capítulo anterior, não foi utilizada nos experimentos deste capítulo pois, além de reconhecer apenas a categoria Pessoa, ela foi construída e adaptada para a CD do Segundo HAREM, utilizada aqui como teste. A LG utilizada nos experimentos a seguir foi construída conforme proposta apresentada na Seção 5.1 considerando apenas as evidências no conjunto de treino.

Na Seção 7.1, o resultado da abordagem proposta, CRF+LG, é apresentado e comparado aos resultados das abordagens LG e CRF aplicadas individualmente. Uma análise de erros obtidos por CRF+LG é realizada na Seção 7.2. Nessa seção, também é apresentada uma avaliação de desempenho do CRF e CRF+LG sem algumas inconsistências identificadas entre as CDs do Primeiro e Segundo HAREM.

Uma comparação dos resultados do CRF+LG com os resultados de dois sistemas reportados na literatura foi realizada na Seção 7.3. Já a Seção 7.4 apresenta uma investigação do impacto de algumas decisões de pré-processamento no resultado do CRF+LG.

A Seção 7.5 mostra o desempenho de LG, CRF e CRF+LG em outros *corpus*. Essa seção também menciona alguns resultados obtidos com pequenas adaptações na LG para esses *corpus*. Por fim, a Seção 7.6 apresenta um estudo dos limites do CRF, apontando os limites inferior e superior da abordagem proposta.

## 7.1 Comparação das técnicas LG, CRF e CRF+LG

Inicialmente, as CDs do Primeiro e Segundo HAREM foram utilizadas como bases de treino e teste, respectivamente e as técnicas LG e CRF foram aplicadas individualmente para avaliar seus efeitos em relação à estratégia combinada CRF+LG. A Tabela 11 apresenta os resultados para as tarefas de identificação (verificação dos limites das NEs) e classificação (verificação das categorias atribuídas às NEs).

Tabela 11 – Comparação: LG x CRF x CRF+LG

| **Sistemas** | **Identificação** | | | **Classificação** | | |
|---|---|---|---|---|---|---|
| | **P (%)** | **A (%)** | **F (%)** | **P (%)** | **A (%)** | **F (%)** |
| **LG** | 71.27 | 28.48 | 40.70 | 64.80 | 25.06 | 36.14 |
| **CRF** | 79.03 | 66.13 | 72.01 | 64.92 | 52.59 | 58.11 |
| **CRF+LG** | **79.86** | **66.76** | **72.73** | **66.52** | **53.85** | **59.52** |

A abrangência obtida pela LG individualmente é bem inferior às demais pois a LG

captura apenas algumas heurísticas gerais para o NER identificadas no *corpus* de treino. Acredita-se que fazendo um estudo linguístico e inserindo novas regras que capturem o conhecimento humano, a abrangência poderia aumentar. Esperava-se obter uma precisão maior com a LG, mas algumas regras muito genéricas e inconsistências entre as CDs de treino e teste interferiram nessa métrica, conforme será discutido na próxima seção.

O ganho obtido por CRF+LG em relação ao CRF foi ligeiramente maior na tarefa de classificação. Esse ganho (1.41 pontos percentuais em medida-F) pode parecer pequeno, mas corresponde a 42 NEs corretamente classificadas a mais (CRF reconhece 4762 e CRF+LG reconhece 4804 NEs corretamente).

O teste dos postos sinalizados de Wilcoxon (Seção 4.3) foi realizado com o objetivo de verificar se existe diferença significativa entre CRF e CRF+LG, ou seja, se CRF+LG realmente obtém resultados melhores do que o CRF. A hipótese nula ($H_0$) de que ambos obtiveram o mesmo resultado estatisticamente foi testada contra a hipótese alternativa ($H_1$) de que os resultados produzidos por CRF são menores do que os obtidos por CRF+LG. O teste foi aplicado aos resultados da métrica Medida-F para as 10 categorias de NEs.

A Tabela 12 apresenta os cálculos realizados. A segunda e terceira colunas apresentam respectivamente os resultados de Medida-F obtidos por CRF e CRF+LG por categoria. Analisando a última coluna da tabela, o valor mínimo considerando a soma dos postos positivos e negativos foi T = 3. Considerando o tamanho da amostra n = 10 e o nível de significância $\alpha = 0.01$ (uma cauda), o valor crítico obtido foi 5 (Apêndice A). Como 3 é menor que 5, $H_0$ é rejeitada. Logo, o teste estatístico confirmou para as 10 categorias de NEs classificadas a diferença significativa entre CRF e CRF+LG.

Tabela 12 – Resultados por Categoria e Cálculos do Teste Estatístico

| **Categorias** | **CRF (X)** | **CRF+LG (Y)** | **D = X-Y** | **Posto de \|D\|** | **Posto sinalizado** |
|---|---|---|---|---|---|
| **Pessoa** | 59.58 | 61.99 | -2.41 | 6 | -6 |
| **Local** | 55.30 | 56.10 | -0.80 | 4 | -4 |
| **Organização** | 49.16 | 50.28 | -1.12 | 5 | -5 |
| **Acontecimento** | 19.11 | 28.32 | -9.21 | 9 | -9 |
| **Obra** | 22.38 | 25.10 | -2.72 | 7 | -7 |
| **Abstração** | 9.60 | 9.50 | 0.10 | 1 | 1 |
| **Coisa** | 5.33 | 5.01 | 0.32 | 2 | 2 |
| **Tempo** | 4.77 | 5.22 | -0.45 | 3 | -3 |
| **Valor** | 54.42 | 61.73 | -7.31 | 8 | -8 |
| **Outro** | 1.40 | 11.11 | -9.71 | 10 | -10 |

Observe na Tabela 12 que CRF+LG obteve resultados ligeiramente inferiores aos do CRF para apenas duas categorias: Abstração e Coisa. De fato, foi observado que algumas regras inseridas nos LGGs dessas categorias reconheceram falso-positivos. No

caso da categoria Coisa, por falta de evidências internas e externas, o LGG reconhece apenas algumas palavras que foram anotadas com essa categoria no *corpus* de treino, como apresentado na Seção 5.1. Porém, no *corpus* de teste algumas dessas palavras não eram NEs ou foram anotadas como parte de NEs de forma errada.

O ganho obtido ao usar CRF+LG foi maior em algumas categorias como Acontecimento, Valor e Outro. No caso da categoria Outro, que é a que tem menos exemplos no *corpus* de treino, a inserção de uma única regra para reconhecer nomes de Prêmios (veja em C.10), aumentou em quase 10 pontos percentuais a medida-F. As NEs da categoria Valor possuem um padrão mais rígido e conhecido que foi capturado e inserido no LGG correspondente mais facilmente. Já as NEs da categoria Acontecimento possuem muitas palavras conhecidas como evidências internas que também foram inseridas no LGG.

Algumas situações onde o CRF errou e CRF+LG fez a classificação correta (dada a sugestão correta pela LG) são descritas a seguir:

1. *... grau de licenciatura em <EM ID="1" CATEG="ABSTRACCAO">Enfermagem</EM> ...*

   CRF classificou como Local porque a NE é precedida pela preposição *em*, mas a LG classificou como Abstração pois *licenciatura em* é uma evidência externa para reconhecer NEs dessa categoria.

2. *<EM ID="2" CATEG="PESSOA">dr. Catanho de Meneses</EM>*

   No *corpus* de treino não tinha exemplo de NE da categoria Pessoa incluindo *dr.* iniciando com letra minúscula e o CRF classificou *Catanho de Meneses* como Abstração. A LG reconheceu corretamente a NE pois *dr.* é uma evidência interna identificada pelo LGG que reconhece Pessoa.

3. *... deixando aliviados os <EM ID="3" CATEG="VALOR">mais de 30</EM> países...*

   No *corpus* de treino existem exemplos onde *mais de* faz parte da NE e outros exemplos onde não faz parte. CRF classificou *30* como Tempo e a LG classificou corretamente pois *mais de* é uma evidência interna para reconhecer a categoria Valor.

4. *Há mais Marias na <EM ID="4" CATEG="LOCAL">Terra</EM>*

   CRF classificou como Coisa pois a palavra *Terra* só aparece como Coisa no treino. Porém, a LG deu a sugestão correta indicando como Local pois a palavra é antecedida por *na*.

5. *Cada edição do <EM ID="5" CATEG="OBRA">Programa de Actividades</EM> assinalará...*

Apesar de ter exemplos semelhantes no treino, CRF classificou a NE como Organização. A LG reconhece a palavra *Programa* como evidência interna para NEs da categoria Obra.

6. *... e dos anéis de <EM ID="6" CATEG="COISA">Saturno</EM>.*

   O *corpus* de treino só possui um exemplo de *Saturno* como Coisa, mas CRF reconheceu como Local provavelmente por conta da preposição *de*. A LG reconhece essa palavra como Coisa, a não ser que alguma evidência externa altere essa classificação.

## 7.2 Análise de Erros e Inconsistências

Analisando os falso-positivos e falso-negativos obtidos por CRF+LG, foram observados alguns erros como:

1. preposições, conjunções e hífens que não são considerados parte de NEs visto que são comuns também fora de NEs. Ex:

   *Joaninha Sampaio* classificada como Pessoa sendo que o nome era *Joaninha Sampaio e Melo*;

   *STN* classificada como Organização sendo que o nome da Organização era *STN - SISTEMA DE TRANSMISSÃO DO NORDESTE*.

2. NEs classificadas em uma categoria sendo que fazem parte de uma NE maior de outra categoria. Ex:

   *José Mourinho* classificado como Pessoa sendo que *Liderança - As Lições de José Mourinho* deveria ser classificado como Obra;

   *cerca de 1520* classificado como Valor sendo que *em cerca de 1520* deveria ser classificado como Tempo.

3. palavras escritas em maiúsculas rotuladas como Organização. Ex:

   *FESTA* que não é NE;

   *DNA* que deveria ser classificada como Coisa.

4. abreviações que não foram incluídas como parte da NE porque foram separadas no processo de segmentação ou *tokenization*. Ex:

   *7000* classificado como valor sendo que *7000 a.C.* deveria ter sido classificado como Tempo;

   *Auro Soares de Moura Andrade* classificado como Local sendo que o nome de local era *Av. Auro Soares de Moura Andrade*.

5. NEs distintas que foram consideradas uma única NE incluindo palavras minúsculas que aparecem entre elas. Ex:

   *Mehrgarh na Índia* classificado como Local sendo que os nomes de locais eram *Mehrgarh* e *Índia*;

   *Conselho e da Comissão* classificado como Organização sendo que *Conselho* e *Comissão* deveriam ser classificados como Organização.

Também foi observado que muitos erros ocorreram devido a inconsistências entre as CDs do Primeiro HAREM e Segundo HAREM. Por exemplo, na CD do Primeiro HAREM, *strings* como *2004* antecedidas pela preposição *em* são consideradas NEs da categoria Tempo e o CRF+LG aprendeu dessa forma e rotulou todas as *strings* semelhantes antecedidas por *em* como Tempo. Porém, na CD do Segundo HAREM a preposição *em* faz parte da NE. Logo, todas essas NEs foram consideradas falso-positivos ao computar as métricas. O mesmo acontece em outras situações das categorias Tempo (NEs antecedidas por *até*, *no*, *a partir de*, *durante o* e *no dia*), Valor (NEs antecedidas por *até*, *aproximadamente*, *quase* e *menos de*) e Pessoa (NEs antecedidas por *presidente*, *chefe*, *vovó*, dentre outras).

Essas inconsistências também afetaram os resultados da LG apresentados na Tabela 11 pois, como a LG foi construída considerando a CD do Primeiro HAREM, essas palavras também não foram incluídas como parte das NEs.

Com o objetivo de avaliar o desempenho do CRF e CRF+LG sem essas inconsistências, a técnica de validação cruzada *10-fold cross-validation* foi utilizada considerando somente a CD do Segundo HAREM. O CRF e CRF+LG foram aplicados usando os mesmos subconjuntos de treino e teste. Os resultados para cada subconjunto de teste são apresentados nas Tabelas 13 e 14 para o CRF e CRF+LG, respectivamente.

Embora esse experimento seja diferente do apresentado na seção anterior, os resultados em medida-F são aproximadamente 12 pontos percentuais superiores tanto para o CRF, quanto para o CRF+LG. Nesse caso, o ganho obtido por CRF+LG em relação ao CRF foi pouco mais de 2 pontos percentuais em medida-F na média.

Um novo experimento foi realizado removendo as principais inconsistências detectadas entre as CDs do Primeiro e Segundo HAREM. A remoção da preposição *em* antes de uma data, por exemplo, aumentou em 4 pontos percentuais a medida-F na tarefa de classificação usando CRF+LG. Isso corresponde a 272 NEs que foram identificadas corretamente (da forma aprendida no treino) e não estavam sendo contabilizadas. As métricas do primeiro experimento (CD do Primeiro HAREM como treino e CD do Segundo HAREM como teste) foram computadas novamente considerando essa nova CD do Segundo HAREM (sem algumas inconsistências) como referência.

Os resultados obtidos são apresentados na Tabela 15.

Tabela 13 – Validação cruzada usando CRF na CD do Segundo HAREM

| **Fold** | **CRF** | | | | | |
|---|---|---|---|---|---|---|
| | **Identificação** | | | **Classificação** | | |
| | **P (%)** | **A (%)** | **F (%)** | **P (%)** | **A (%)** | **F (%)** |
| **1** | 87.92 | 80.41 | 84.00 | 77.01 | 67.90 | 72.17 |
| **2** | 87.14 | 81.95 | 84.47 | 72.83 | 66.80 | 69.68 |
| **3** | 85.04 | 74.77 | 79.58 | 70.73 | 59.74 | 64.78 |
| **4** | 85.06 | 79.90 | 82.40 | 73.32 | 67.13 | 70.09 |
| **5** | 85.50 | 77.38 | 81.24 | 71.87 | 62.83 | 67.05 |
| **6** | 85.98 | 81.40 | 83.63 | 73.46 | 67.99 | 70.62 |
| **7** | 89.29 | 79.14 | 83.91 | 79.16 | 67.45 | 72.84 |
| **8** | 87.07 | 81.67 | 84.63 | 74.68 | 67.13 | 70.71 |
| **9** | 83.07 | 78.53 | 80.74 | 71.05 | 65.14 | 67.97 |
| **10** | 86.80 | 80.64 | 83.61 | 75.30 | 68.40 | 71.69 |
| **Média** | **86.36** | **79.57** | **82.81** | **73.94** | **66.05** | **69.75** |
| **Desvio padrão** | 1.70 | 2.10 | 1.65 | 2.52 | 2.61 | 2.36 |

Tabela 14 – Validação cruzada usando CRF+LG na CD do Segundo HAREM

| **Fold** | **CRF+LG** | | | | | |
|---|---|---|---|---|---|---|
| | **Identificação** | | | **Classificação** | | |
| | **P (%)** | **A (%)** | **F (%)** | **P (%)** | **A (%)** | **F (%)** |
| **1** | 90.15 | 82.35 | 86.07 | 79.75 | 70.24 | 74.69 |
| **2** | 87.25 | 81.63 | 84.35 | 73.66 | 67.22 | 70.29 |
| **3** | 85.34 | 76.98 | 80.95 | 71.31 | 61.81 | 66.22 |
| **4** | 84.57 | 79.68 | 82.05 | 73.25 | 67.27 | 70.13 |
| **5** | 88.24 | 77.59 | 82.57 | 74.72 | 63.45 | 68.63 |
| **6** | 88.12 | 82.10 | 85.01 | 76.80 | 69.93 | 73.20 |
| **7** | 90.86 | 82.24 | 86.33 | 79.58 | 69.26 | 74.06 |
| **8** | 89.45 | 80.55 | 84.77 | 77.66 | 67.55 | 72.25 |
| **9** | 90.27 | 83.02 | 86.50 | 79.06 | 70.50 | 74.54 |
| **10** | 90.68 | 82.80 | 86.56 | 79.00 | 70.54 | 74.53 |
| **Média** | **88.49** | **80.90** | **84.52** | **76.48** | **67.78** | **71.85** |
| **Desvio padrão** | 2.10 | 2.04 | 1.91 | 2.88 | 2.87 | 2.77 |

Analisando essa tabela, é possível observar o quanto as inconsistências influenciaram os resultados apresentados na Tabela 11. A precisão de todas as técnicas em ambas as tarefas (identificação e classificação) contribuiu bastante para o ganho final já que a quantidade de falso-positivos diminuiu sem as inconsistências. Analisando dessa forma, a LG foi a técnica que obteve a maior precisão como esperado a princípio, já que foi construída a partir do conhecimento humano. A precisão da LG aumentou 12 pontos percentuais em relação aos resultados apresentados anteriormente. Além disso, a medida-F de CRF+LG para classificação aumentou quase 6 pontos percentuais alcançando 65.33%.

Tabela 15 – Comparação: LG x CRF x CRF+LG (sem as principais inconsistências)

| Sistemas | Identificação | | | Classificação | | |
|---|---|---|---|---|---|---|
| | P (%) | A (%) | F (%) | P (%) | A (%) | F (%) |
| **LG** | 83.47 | 33.37 | 47.68 | 76.98 | 29.79 | 42.96 |
| **CRF** | 85.45 | 71.58 | 77.90 | 71.18 | 57.73 | 63.75 |
| **CRF+LG** | **86.39** | **72.27** | **78.70** | **72.99** | **59.12** | **65.33** |

Esses resultados são muito importantes para conhecer o potencial real do CRF+LG em uma base de dados consistente.

## 7.3 Comparação com Abordagens Apresentadas na Literatura

Os resultados do CRF+LG foram comparados com os resultados de dois sistemas de NER que realizaram experimentos sob as mesmas condições deste trabalho. O primeiro foi o NERP-CRF (AMARAL, 2013; AMARAL; VIEIRA, 2014). As principais diferenças entre CRF+LG e NERP-CRF foram mencionadas na Seção 3.4.

O arquivo .xml anotado pelo NERP-CRF foi obtido conforme indicado em Amaral et al. (2014)[1]. Os identificadores de cada NE (ID) foram modificados acrescentando um número único para computar as métricas já que o NERP-CRF usa o mesmo ID para todas as NEs em um documento e isso muda o desempenho real do sistema computado usando os *scripts* do Segundo HAREM. Quando um ID único não é atribuído a cada NE, as métricas computadas não consideram todos os falso-positivos, somente um por documento que tem falso-positivo. Isso foi observado ao se estudar a arquitetura de avaliação do Segundo HAREM e analisar os arquivos gerados por cada módulo.

Como é possível observar na Tabela 16, os resultados obtidos superam em mais de 9 pontos percentuais a abrangência do NERP-CRF e em quase 8 pontos percentuais a medida-F na tarefa de classificação, o que representa um bom ganho.

Tabela 16 – Comparação: NERP-CRF x CRF+LG

| Sistemas | Identificação | | | Classificação | | |
|---|---|---|---|---|---|---|
| | P (%) | A (%) | F (%) | P (%) | A (%) | F (%) |
| **NERP-CRF** | 74.83 | 54.86 | 63.31 | 62.13 | 44.08 | 51.57 |
| **CRF+LG** | **79.86** | **66.76** | **72.73** | **66.52** | **53.85** | **59.52** |

CRF+LG também foi comparado com o sistema baseado em CharWNN (SANTOS;

[1] http://www.inf.pucrs.br/linatural/recursos_para_reconhecimento_de_entidades_nomeadas/-NERP_CRF.xml

GUIMARAES, 2015). Como Santos e Guimaraes (2015) não apresentaram os resultados para a CD do Segundo HAREM, CRF+LG foi executado novamente utilizando as CDs que eles usaram para treino (CD do Primeiro HAREM) e teste (Mini-HAREM). O *script* de avaliação da CoNLL-2002[2] que avalia a tarefa de classificação também foi usada como eles fizeram para computar as métricas neste experimento. Observe na Tabela 17 que CRF+LG obteve um ganho de aproximadamente 2 pontos percentuais em cada métrica avaliada.

Tabela 17 – Comparação: CharWNN x CRF+LG

| **Sistemas** | **P (%)** | **A (%)** | **F (%)** |
|---|---|---|---|
| **CharWNN** | 65.21 | 52.27 | 58.03 |
| **CRF+LG** | **67.09** | **54.85** | **60.36** |

Os resultados apresentados na Tabela 17 foram obtidos para um cenário seletivo (categorias Pessoa, Local, Organização, Tempo e Valor) porque os resultados apresentados por Santos e Guimaraes (2015) para as 10 categorias do HAREM foram obtidos usando vetores de representação de palavras (*word embeddings*) previamente treinados sem supervisão por Santos e Zadrozny (2014) com outros três corpus (Wikipedia português, CETENFolha e CETEMPublico) e os resultados para esse cenário seletivo não. Esses *corpus* possuem juntos aproximadamente 400 milhões de palavras. Portanto, a comparação com esse resultado seria injusta visto que o CRF+LG não usa *word embeddings*, nem outros *corpus* para geração de *features* que são pré-definidas.

Como o trabalho de Santos e Guimaraes (2015), outros trabalhos recentes apresentaram resultados superiores (CASTRO; SILVA; SOARES, 2018; COSTA; PAETZOLD, 2018) usando vetores de representação de palavras. A comparação com e entre esses resultados deve ser cuidadosa porque eles fazem o pré-treinamento em *corpus* diferentes e de tamanhos diferentes. Assim, não é possível saber se o resultado é melhor porque o pré-treinamento foi realizado em um *corpora* maior ou porque a técnica usada é realmente superior. Talvez o tamanho do *corpora* usado no pré-treinamento seja o motivo de resultados tão discrepantes (diferença de 24% em medida-F entre o pior e o melhor sistema) obtidos usando a mesma técnica relatada em Ji et al. (2017). Essas técnicas exigem, além do *corpus* anotado, um outro muito grande para realizar o pré-treinamento ou os *word embeddings* pré-treinados.

## 7.4 Avaliação de Ferramentas de Pré-processamento

Analisando os resultados apresentados nas Tabelas 11 e 16, foi observado que apenas o uso do CRF já superava os resultados do NERP-CRF. Esse é um resultado interessante

[2] http://www.cnts.ua.ac.be/conll2002/ner/bin/conlleval.txt

obtido possivelmente por causa do pré-processamento dos textos que foi realizado de uma forma diferente, já que as características usadas foram as mesmas do NERP-CRF neste caso.

Por conta desse resultado, uma investigação do impacto de algumas decisões de pré-processamento foi realizada. CRF+LG foi executado novamente variando algumas possíveis ferramentas de pré-processamento (com sua configuração *default*) na realização das tarefas de segmentação, *tokenization* e *POS-Tagging*. Além das ferramentas OpenNLP e Unitex, a ferramenta Freeling[3] também foi experimentada. Ela também realiza tarefas de NLP em vários idiomas.

As combinações possíveis foram testadas já que o Unitex não realiza *tokenization* e *POS-Tagging* em contexto e Freeling realiza a segmentação e *tokenization* em conjunto. Assim, sempre que Unitex ou OpenNLP foi usado para segmentar, o OpenNLP realizou a *tokenization*; e sempre que Freeling foi usado para segmentar, ele também realizou a *tokenization*.

Tabela 18 – Resultados da Combinação de Ferramentas

| **Combinação de Ferramentas** | | | **Identificação** | | | **Classificação** | | |
|---|---|---|---|---|---|---|---|---|
| **Seg.** | **Tok.** | **PosT.** | **P(%)** | **A(%)** | **F(%)** | **P(%)** | **A(%)** | **F(%)** |
| Unitex | OpenNLP | OpenNLP | 78.88 | **66.15** | 71.96 | 65.62 | **53.28** | 58.81 |
| OpenNLP | OpenNLP | OpenNLP | 77.64 | 64.07 | 70.20 | 64.13 | 51.24 | 56.96 |
| Freeling | Freeling | OpenNLP | **80.28** | 65.71 | **72.27** | **67.13** | 53.20 | **59.36** |
| Unitex | OpenNLP | Freeling | 76.47 | 63.14 | 69.17 | 61.74 | 49.35 | 54.85 |
| OpenNLP | OpenNLP | Freeling | 77.24 | 62.75 | 69.24 | 62.93 | 49.49 | 55.41 |
| Freeling | Freeling | Freeling | 78.81 | 65.57 | 71.58 | 65.57 | 52.83 | 58.52 |

Os resultados são apresentados na Tabela 18. As abreviações Seg., Tok. e PosT. foram utilizadas para segmentação, *tokenization* e *POS-Tagging*, respectivamente. Da pior para a melhor combinação houve um ganho de 4.5 pontos percentuais em Medida-F para a tarefa de classificação e aproximadamente 3 pontos percentuais para a tarefa de identificação. A combinação das ferramentas Freeling para segmentação e *tokenization* e OpenNLP para POS-Tagging foi a que obteve melhores resultados. Observe que a combinação utilizada neste trabalho obteve o segundo melhor resultado em Medida-F considerando a configuração *default*.

## 7.5 Avaliação em outros *Corpus*

Alguns experimentos foram realizados com o objetivo de avaliar o desempenho da LG construída e dos modelos CRF e CRF+LG obtidos em outros domínios.

[3] http://nlp.lsi.upc.edu/freeling/node/1

Inicialmente, o *corpus* aTribuna foi utilizado. O LGG construído para a categoria Pessoa foi aplicado já que esse *corpus* possui a anotação de nomes de pessoas apenas. Os modelos CRF e CRF+LG treinados na CD do Primeiro HAREM foram utilizados. Os resultados são apresentados nas três primeiras linhas da Tabela 19.

A LG obteve um desempenho superior ao CRF e a maior precisão. A abordagem proposta neste trabalho, CRF+LG, obteve a melhor Medida-F e um ganho de 6.4 pontos percentuais em relação ao CRF nesse caso. Acredita-se que esse aumento seja porque algumas evidências capturadas pelo LGG aparecem mais no domínio de notícias de jornal (Ex: palavras como *diz* e *disse* no contexto à direita e à esquerda).

Tabela 19 – Avaliação no *corpus* aTribuna

| **Sistemas** | **P (%)** | **A (%)** | **F(%)** |
|---|---|---|---|
| **LG** | 83.51 | 25.52 | 39.10 |
| **CRF** | 76.10 | 23.41 | 35.80 |
| **CRF+LG** | **76.15** | **29.19** | **42.20** |
| **LG*** | 76.99 | 37.61 | 50.53 |

Analisando alguns outros textos do jornal A Tribuna (não pertencentes ao corpus aTribuna anotado neste trabalho), foram observados alguns padrões bem rígidos para escrita de nomes nesses textos. Algumas adaptações foram inseridas na LG construída neste trabalho para reconhecer esses padrões:

1. Sequências de palavras escritas em maiúsculas sucedidas por / e afiliação profissional (Ex: *CASSIANO ROSÁRIO/AGÊNCIA ESTADO*).
2. Sequências de palavras escritas em maiúsculas sucedidas por - e data (Ex: *RODRIGO GAVINI - 03/10/2016*).
3. Sequências de palavras que iniciam com letra maiúscula seguidas por vírgula e idade. Após a idade, as palavras *ano*, *anos* ou uma vírgula podem aparecer (Ex: *Gerusa Maria Rassch Gaiba, 52 ,*).

Os resultados obtidos pela LG adaptada (LG*) também são apresentados na Tabela 19. Inserindo apenas essas três regras, o resultado da LG é superior aos demais, alcançando um ganho de mais de 8 pontos percentuais em relação ao CRF+LG na Medida-F. Esse resultado indica que, na ausência de *corpus* do mesmo domínio para treino, pode ser melhor adaptar a LG do que usar um modelo treinado em outro domínio.

Alguns experimentos também foram realizados utilizando o *corpus* SIGARRA. Inicialmente, a LG foi aplicada a esse *corpus* reconhecendo as 4 categorias que possuem correspondência direta com as categorias do HAREM (Pessoa, Localização, Organização e

Evento que correspondem a Pessoa, Local, Organização e Acontecimento) e as categorias Data e Hora que estão incluídas na categoria Tempo do HAREM. O LGG Tempo possui grafos distintos para reconhecer Data e Hora; apenas as saídas desses grafos foram alteradas para esse experimento.

O método de amostragem *holdout* foi utilizado para avaliar o desempenho do CRF e CRF+LG no *corpus* SIGARRA, já que os modelos CRF e CRF+LG foram treinados anteriormente para reconhecer as 10 categorias do HAREM. A divisão comum, 2/3 para treino e 1/3 para teste, foi utilizada. A LG não foi modificada para execução do CRF+LG. A Tabela 20 apresenta os resultados para a tarefa de classificação.

Tabela 20 – Avaliação no *corpus* SIGARRA

| **Sistemas** | **P (%)** | **A (%)** | **F(%)** |
|---|---|---|---|
| **LG** | 66.11 | 45.05 | 53.59 |
| **CRF** | 86.30 | 75.71 | 80.66 |
| **CRF+LG** | **86.27** | **76.60** | **81.15** |
| **LG*** | 70.79 | 52.46 | 60.26 |

Os resultados obtidos para LG foram inferiores porque ela não reconhece todas as categorias do SIGARRA e porque a maior parte das NEs da categoria Unidadeorganica são reconhecidas como Organização, fazendo com que a precisão diminua. Apesar da LG não ter sido construída para esse *corpus*, CRF+LG obteve um ganho de aproximadamente 0.5 pontos percentuais em relação ao CRF para Medida-F.

Cinquenta textos desse *corpus* foram selecionados aleatoriamente e analisados. Algumas adaptações foram realizadas na LG para reconhecer os seguintes padrões observados:

1. Sequências de palavras com a primeira letra maiúscula iniciadas por *Departamento de* como Unidadeorganica (Ex: *Departamento de Produção e Sistemas*).

2. Sequências de palavras com a primeira letra maiúscula ou números iniciados por palavras como *Sala*, *Salão*, *Auditório* e *Anfiteatro* como Localização (Ex: *Anfiteatro Nobre*).

3. Sequências de palavras com a primeira letra maiúscula iniciadas por expressões como *Mestrado em*, *Doutoramento em* e *Licenciatura em* como Curso (Ex: *Doutoramento em Segurança e Saúde Ocupacionais*).

Também foi incluído um LGG para reconhecer as NEs do modelo organizacional da Universidade do Porto pré-definidas em Pires (2017). Assim, a LG passou a reconhecer também as categorias Curso e Unidadeorganica, concluindo as 8 categorias do SIGARRA.

Os resultados obtidos são apresentados na Tabela 20 (LG*). Esses resultados foram obtidos aplicando a LG aos outros 855 documentos não analisados durante a sua adaptação.

Embora os resultados da LG adaptada não superem os obtidos por CRF e CRF+LG nesse caso, já que os modelos foram treinados e testados no próprio *corpus* SIGARRA, os resultados mostram o potencial da LG. Com poucas adaptações, ela obteve um ganho de aproximadamente 4.6 pontos percentuais em Precisão, 7.4 pontos percentuais em Abrangência e 6.7 pontos percentuais em Medida-F.

Os resultados apresentados para os *corpus* aTribuna e SIGARRA mostram que a LG pode ser adaptada para uso em outros domínios caso não exista um *corpus* para treino ou no caso do *corpus* de treino ser pequeno.

## 7.6 Estudo dos Limites do CRF

Com o objetivo de estudar os limites do CRF usando o resultado de um outro classificador como uma característica adicional, os seguintes experimentos foram realizados utilizando as CDs do Primeiro e Segundo HAREM como bases de treino e teste respectivamente:

- Limite inferior (*Lower bound*): um rótulo aleatório entre os 11 possíveis (10 categorias do HAREM mais o rótulo O) foi inserido como característica *tip* nos vetores de características em vez do rótulo atribuído pela LG. O experimento foi repetido 100 vezes e a média foi calculada para cada métrica (o desvio padrão obtido foi menor que 0.01 para cada métrica).
- Perturbação: um rótulo errado (contrário à categoria correta da palavra) foi inserido como característica *tip* nos vetores de características em vez do rótulo atribuído pela LG. Se o rótulo correto da palavra era I-X, sendo X uma das categorias do HAREM, a característica *tip* atribuída foi O; se o rótulo era O, a característica *tip* atribuída foi I-X onde a categoria X foi escolhida aleatoriamente entre as 10 categorias do HAREM. Esse experimento também foi repetido 100 vezes e a média foi calculada para cada métrica (o desvio padrão obtido foi menor que 0.01 para cada métrica).
- Limite superior (*Upper bound*): o rótulo correto foi inserido como característica *tip* nos vetores de características em vez do rótulo atribuído pela LG.

Os valores da característica *tip* foram atribuídos dessa forma nos vetores de características do treino e teste.

Os resultados obtidos são apresentados na Tabela 21.

Como esperado, o CRF com característica *tip* aleatória (limite inferior) obteve resultados muito próximos (ligeiramente inferiores) ao CRF sem a característica *tip*

Tabela 21 – Comparação: CRF x CRF+LG x Limite inferior x Limite superior

| **Experimentos** | **Identificação** | | | **Classificação** | | |
|---|---|---|---|---|---|---|
| | **P (%)** | **A (%)** | **F (%)** | **P (%)** | **A (%)** | **F (%)** |
| **CRF** | 79.03 | 66.13 | 72.01 | 64.92 | 52.59 | 58.11 |
| **CRF+LG** | 79.86 | 66.76 | 72.73 | 66.52 | 53.85 | 59.52 |
| **Limite Inferior** | *78.78* | *65.12* | *71.30* | *64.66* | *51.74* | *57.48* |
| **Perturbação** | 91.83 | 87.06 | 89.39 | 72.92 | 67.03 | 69.85 |
| **Limite Superior** | **95.68** | **94.56** | **95.12** | **95.67** | **91.69** | **93.63** |

(apresentado apenas como CRF na Tabela 21). Isso indica que o peso estimado para a função característica *tip* foi muito baixo pois ela não é discriminante.

Inicialmente, havia sido pensado que o CRF com rótulo errado (perturbação) seria um limite inferior já que o rótulo era completamente o oposto do rótulo correto da palavra. Porém, o CRF identificou e aprendeu um padrão pois teve uma informação parcial. Quando a característica *tip* era I-X, o CRF aprendeu que o rótulo seria O, fazendo a distinção entre o que não era NE e o que era. Quando *tip* era O, o CRF aprendeu que a palavra era uma NE (I-X), mas não sabia qual e precisava usar as outras características para predizer o rótulo, cometendo erros.

Obviamente, os resultados para o limite superior são os melhores, já que a característica *tip* era mais discriminante e, consequentemente, teve um peso estimado maior. Observe que, mesmo com a sugestão (*tip*) sempre certa, o CRF não acerta 100% porque considera as 18 características no aprendizado, não apenas a *tip*.

Os resultados apresentados na Tabela 21 para limite inferior e superior representam os limites para o desempenho do CRF combinado com um outro classificador da maneira proposta neste trabalho (inserindo o resultado do classificador como característica adicional para o CRF) e sob as condições dos experimentos (usando as CDs do HAREM como treino e teste e as 18 características propostas).

Os resultados para o limite superior são muito importantes porque nos permitem prever o ganho máximo que podemos alcançar combinando o CRF com uma LG melhorada ou outros classificadores.

# 8 Conclusões e Trabalhos Futuros

Este trabalho apresentou uma abordagem híbrida, CRF+LG, para o Reconhecimento de Entidades Nomeadas em textos em Português usando Campos Aleatórios Condicionais e Gramáticas Locais. A classificação obtida pela LG foi enviada como característica para o processo de aprendizado do modelo de predição CRF junto com outras características. O modelo CRF realiza a rotulação final das NEs. Essa abordagem é uma boa forma de considerar a *expertise* humana que pode capturar regras que não aparecem nos exemplos do *corpus* anotado usado como treino pelo CRF.

Os resultados obtidos nos experimentos permitiram verificar a hipótese de pesquisa apresentada de que é possível melhorar o desempenho de sistemas de NER em textos escritos em Português usando uma abordagem híbrida, inserindo o conhecimento humano capturado pela abordagem linguística durante o aprendizado de máquina. O ganho obtido em Medida-F por CRF+LG em relação ao CRF foi de aproximadamente 1.4 pontos percentuais na CD do Segundo HAREM, 0.5 pontos percentuais no SIGARRA e 6.4 pontos percentuais no aTribuna sendo que, nesse último, o modelo aplicado foi treinado em outro domínio.

Os resultados obtidos por CRF+LG superaram resultados de sistemas reportados na literatura que realizaram testes em condições equivalentes. Esse ganho foi de aproximadamente 8 pontos percentuais em Medida-F em relação a um sistema que também usou CRF e de 2 pontos percentuais em relação a um sistema que usou Redes Neurais. É importante ressaltar que os ganhos podem se tornar mais expressivos ao usar *corpora* maiores para treino.

Além disso, os resultados apresentados na Seção 7.6 mostraram o limite superior de 93.63% em Medida-F para o desempenho do CRF, indicando qual seria o ganho máximo obtido ao combiná-lo com uma LG melhorada ou outros classificadores da forma proposta neste trabalho.

A LG construída neste trabalho pode ser utilizada em outros domínios e adaptada para eles. A Seção 7.5 mostrou que algumas adaptações feitas na LG para um domínio específico possibilitou alcançar um resultado melhor (50.53%) do que usando um modelo CRF+LG treinado em outro domínio (42.20%). Além disso, a LG pode ser utilizada individualmente quando não há *corpus* de treino disponível ou quando esse *corpus* é pequeno. Ela também pode ser utilizada para melhorar o desempenho de outras técnicas de aprendizado de máquina.

A LG completa e outros recursos gerados durante este trabalho estão disponíveis em:

<https://inf.ufes.br/~elias/dataSets/ner/recursosTese-julianaPirovani.zip>

## 8.1 Trabalhos Futuros

Alguns experimentos realizados mostraram o potencial da LG para uso no problema de NER. Como trabalhos futuros, a LG construída pode ser melhorada pois não foi realizado um estudo linguístico aprofundado para sua construção. Novas regras que capturem o conhecimento humano podem ser inseridas para melhorar o seu desempenho.

Também é interessante estudar a viabilidade de identificar regras e construir LGs de forma automática ou semi-automática com o objetivo de minimizar o esforço humano durante a sua construção. A ferramenta de comparação de concordâncias e o resumo das relações da teoria de conjuntos apresentada na Tabela 8 podem auxiliar a tomada de decisões para este fim.

Além disso, podem ser exploradas outras formas de combinar o resultado obtido pela LG com o CRF ou outras técnicas de aprendizado de máquina. Algumas possibilidades de combinação são citadas na Seção 3.4.

O limite superior da estratégia proposta neste trabalho nos incentiva a testar LGs melhoradas e outros classificadores para informar novas características para o processo de aprendizado do CRF. A ideia é obter resultados cada vez mais próximos do limite superior indicado nos experimentos.

Redes neurais também podem ser utilizadas futuramente para aprendizado não supervisionado de características e para avaliação como técnica de aprendizado de máquina em uma abordagem híbrida. Nesse caso, a perspectiva de economia de recursos (menos *corpora* para treino) proposta neste trabalho seria desconsiderada.

# Referências

AMARAL, D. O. et al. Comparative Analysis of Portuguese Named Entities Recognition Tools. In: CHAIR), N. C. C. et al. (Ed.). *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland: European Language Resources Association (ELRA), 2014. p. 2554–2558.

AMARAL, D. O. F.; VIEIRA, R. NERP-CRF: Uma Ferramenta para o Reconhecimento de Entidades Nomeadas por meio de Conditional Random Fields. *Linguamática*, v. 6, n. 1, p. 41–49, 2014.

AMARAL, D. O. F. d. *O Reconhecimento de Entidades Nomeadas por Meio de Conditional Random Fields para a Língua Portuguesa*. Dissertação (Mestrado) — Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, Brasil, 2013.

AMARAL, D. O. F. d. *Reconhecimento de Entidades Nomeadas na Área da Geologia: Bacias Sedimentares Brasileiras*. Tese (Doutorado) — Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, Brasil, 2017.

AMARAL, D. O. F. do; BUFFET, M.; VIEIRA, R. Comparative Analysis between Notations to Classify Named Entities using Conditional Random Fields. In: *Proceedings of the 10th Brazilian Symposium in Information and Human Language Technology - STIL*. Natal, RN: SBC, 2015. p. 27–31.

Apache OpenNLP. 2019. Acesso em: 15/02/2019. Disponível em: <https://opennlp.apache.org/>.

ARLOT, S.; CELISSE, A. et al. A Survey of Cross-validation Procedures for Model Selection. *Statistics surveys*, The author, under a Creative Commons Attribution License, v. 4, p. 40–79, 2010.

BALDWIN, T. et al. Shared tasks of the 2015 Workshop on Noisy User-generated Text: Twitter Lexical Normalization and Named Entity Recognition. In: *Proceedings of the Workshop on Noisy User-generated Text*. Beijing, China: ACL, 2015. p. 126–135.

BAPTISTA, J. A Local Grammar of Proper Nouns. In: *Seminários de Linguística*. Portugal: Faro: Universidade do Algarve, 1998. v. 2, p. 21–37.

BARRETO, F. et al. Open Resources and Tools for the Shallow Processing of Portuguese: the TagShare Project. In: *Proceedings of LREC 2006*. Genoa, Itália: Citeseer, 2006.

BAYRAKTAR, Ö.; TEMIZEL, T. T. Person Name Extraction from Turkish Financial News Text Using Local Grammar-Based Approach. In: IEEE. *Computer and Information Sciences, 2008. ISCIS'08. 23rd International Symposium on*. Istanbul, Turkey, 2008. p. 1–4.

BICK, E. *The Parsing System Palavras: Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. Tese (Doutorado), Arhus, Danemark, 2000.

BIRD, S.; KLEIN, E.; LOPER, E. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. Califórnia, EUA: O'Reilly Media, Inc., 2009.

BUSSAB, W. d. O.; MORETTIN, P. A. *Estatística Básica*. São Paulo: Saraiva, 2010.

CARDOSO, N. REMBRANDT-Reconhecimento de Entidades Mencionadas Baseado em Relaações e Análise Detalhada do Texto. In: *In Cristina Mota and Diana Santos (eds.). Desafios na Avaliaação Conjunta do Reconhecimento de Entidades Mencionadas*. Portugal: Linguateca, 2008. v. 1, p. 195–211.

CASTRO, P. V. Q. d.; SILVA, N. F. F. da; SOARES, A. da S. Portuguese Named Entity Recognition Using LSTM-CRF. In: *Villavicencio A. et al. (eds) Computational Processing of the Portuguese Language. PROPOR 2018. Lecture Notes in Computer Science, vol 11122.* Canela, RS: Springer, Cham, 2018. p. 83–92.

CONSTANT, M.; TELLIER, I. Evaluating the Impact of External Lexical Resources into a CRF-based Multiword Segmenter and Part-of-speech Tagger. In: *8th International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey: European Language Resources Association (ELRA), 2012. p. 646–650.

COPARA, J. et al. Conditional Random Fields for Spanish Named Entity Recognition Using Unsupervised Features. In: *Ibero-American Conference on Artificial Intelligence - IBERAMIA 2016*. San José, Costa Rica: Springer, 2016. p. 175–186.

COSTA, P. d.; PAETZOLD, G. H. Effective Sequence Labeling with Hybrid Neural-CRF Models. In: *Villavicencio A. et al. (eds) Computational Processing of the Portuguese Language. PROPOR 2018. Lecture Notes in Computer Science, vol 11122.* Canela, RS: Springer, Cham, 2018. p. 490–498.

DODDINGTON, G. R. et al. The Automatic Content Extraction (ACE) Program-Tasks, Data, and Evaluation. In: *LREC*. Lisboa, PORTUGAL: European Language Resources Association (ELRA), 2004. v. 2, p. 1.

FACELI, K. et al. *Inteligência Artificial: Uma Abordagem de Aprendizado de Máquina*. Rio de Janeiro: LTC, 2011.

FERREIRA, E.; BALSA, J.; BRANCO, A. Combining Rule-based and Statistical methods for Named Entity Recognition in Portuguese. In: *Actas da 5a Workshop em Tecnologias da Informaçao e da Linguagem Humana - STIL*. Rio de Janeiro, RJ: SBC, 2007.

FIELDMAN, R.; SANGER, J. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. New York, USA: Cambridge University Press, 2006.

FONSECA, E. B.; CHIELE, G. C.; VANIN, A. A. Reconhecimento de Entidades Nomeadas para o Português Usando o OpenNLP. *Anais do Encontro Nacional de Inteligência Artificial e Computacional (ENIAC 2015), s. pp*, 2015.

FreeLing. 2018. Acesso em: 02/03/2018. Disponível em: <http://nlp.cs.upc.edu/freeling/>.

FREITAS, C.; ROCHA, P.; BICK, E. Um mundo novo na Floresta Sintá (c) tica -o treebank do Português. *Calidoscópio*, v. 6, n. 3, p. 142–148, 2008.

FRIBURGER, N.; MAUREL, D. Finite-state Transducer Cascades to Extract Named Entities in Texts. *Theoretical Computer Science*, Elsevier, New york, USA, v. 313, n. 1, p. 93–104, 2004.

FRIBURGER, N.; MAUREL, D.; GIACOMETTI, A. Textual Similarity Based on Proper Names. In: *Proceedings of the Workshop Mathematical/Formal methods in Information Retrieval (MFIR '2002) at the 25th ACM SIGIR Conference*. Tampere, Finland: ACM, 2002. p. 155–167.

GOULART, R. R.; LIMA, V. L. S. de. O Contexto no Reconhecimento de Entidades Nomeadas em Textos de Biomedicina. *Simpósio de Tecnologias da Informação e da Língua (STIL)*, p. 1–10, 2009.

GRISHMAN, R.; SUNDHEIM, B. Message Understanding Conference-6: A Brief History. In: *Proceedings of the 16th Conference on Computational Linguistics - COLING '96*. Stroudsburg, PA, USA: Association for Computational Linguistics, 1996. v. 1, p. 466–471.

GROSS, M. The Construction of Local Grammars. *In ROCHE, E.; SCHABÈS, Y. (eds.). Finite-state language processing, Language, Speech, and Communication, Cambridge, Mass.*, MIT Press, p. 329–354, 1997.

GROSS, M. A Bootstrap Method for Constructing Local Grammars. In: BOKAN, N. (Ed.). *Proceedings of the Symposium on Contemporary Mathematics*. Belgrado, Sérvia: University of Belgrad, 1999. p. 229–250.

HAN, J. et al. DECO-MWE: Building a Linguistic Resource of Korean Multiword Expressions for Feature-Based Sentiment Analysis. In: SHIRAI, K. (Ed.). *Proceedings of the LREC 2018 Workshop "The 13th Workshop on Asian Language Resources"*. Miyazaki, Japan: European Language Resources Association (ELRA), 2018. p. 14–20.

JI, H. et al. Overview of TAC-KBP2017 13 Languages Entity Discovery and Linking. In: *Proceedings of the Tenth Text Analysis Conference (TAC2017)*. Maryland, USA: NIST, 2017.

JIA, Y. et al. A Hybrid Approach Using Maximum Entropy Model and Conditional Random Fields to Identify Tibetan Person Names. *Himalayan Linguistics*, v. 15, n. 1, 2016.

JIANG, J. Information Extraction from Text. In: AGGARWAL, C.; ZHAI, C. (Ed.). *Mining text data*. Boston, MA: Springer, US, 2012. p. 11–41.

KARIMAGHALOO, Z.; ARNOLD, D. L.; ARBEL, T. Adaptive Multi-level Conditional Random Fields for Detection and Segmentation of Small Enhanced Pathology in Medical Images. *Medical image analysis*, Elsevier, v. 27, p. 17–30, 2016.

KONKOL, M.; BRYCHCÍN, T.; KONOPÍK, M. Latent Semantics in Named Entity Recognition. *Expert Systems with Applications*, Elsevier, v. 42, n. 7, p. 3470–3479, 2015.

KOZAREVA, Z. et al. Combining Data-driven Systems for Improving Named Entity Recognition. *Data & Knowledge Engineering*, Elsevier, v. 61, n. 3, p. 449–466, 2007.

KRSTEV, C. et al. E-Dictionaries and Finite-State Automata for the Recognition of Named Entities. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. *Proceedings of the 9th International Workshop on Finite State Methods and Natural Language Processing, FSMNLP 2011*. França, 2011. p. 48–56.

LAFFERTY, J.; MCCALLUM, A.; PEREIRA, F. Conditional Random Fields:

Probabilistic Models for Segmenting and Labeling Sequence Data. In: *Proceedings of the Eighteenth International Conference on Machine Learning, ICML 2001*. San Francisco, CA, USA: ACM, 2001. v. 1, p. 282–289.

LAMPLE, G. et al. Neural Architectures for Named Entity Recognition. *arXiv preprint arXiv:1603.01360*, 2016.

Language Tasks. 2019. Acesso em: 18/02/2019. Disponível em: <https://github.com/ltasks/ltasks4j>.

LDC. *Annotation Tasks and Specifications*. 2018. Acesso em: 24/05/2018. Disponível em: <https://www.ldc.upenn.edu/collaborations/past-projects/ace/annotation-tasks-and-specifications>.

Linguateca. 2018. Acesso em: 02/03/18. Disponível em: <http://www.linguateca.pt>.

LIU, X. et al. Recognizing Named Entities in Tweets. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon: Association for Computational Linguistics, 2011. v. 1, p. 359–367.

MANNING, C. D.; SCHÜTZE, H. *Foundations of Statistical Natural Language Processing*. London, England: MIT press, 1999.

MIKHEEV, A.; MOENS, M.; GROVER, C. Named entity recognition without gazetteers. In: *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*. Bergen, Noruega: Association for Computational Linguistics, 1999. p. 1–8.

MILIDIÚ, R.; DUARTE, J.; CAVALCANTE, R. Machine Learning Algorithms for Portuguese Named Entity Recognition. *Inteligência Artificial, Revista Iberoamericana de Inteligência Artificial*, Asociación Española para la Inteligencia Artificial, v. 11, n. 36, p. 67–75, 2007.

MOTA, C. Combining Nooj with Co-training for NER. In: *Applications of Finite-State Language Processing: Selected Papers from the 2008 International NooJ Conference*. Budapest, Hungaria: Cambridge Scholars Publishing, 2010. p. 76–86.

MOTA, C.; SANTOS, D. *Desafios na Avaliação Conjunta do Reconhecimento de Entidades Mencionadas: O Segundo HAREM*. Linguateca, 2008. Disponível em: <https://www.linguateca.pt/LivroSegundoHAREM/>.

MUC-7. *MUC-7 Proceedings*. 2016. Acesso em: 11/10/2018.

MUNIZ, M. C. et al. UNITEX-PB, a Set of Flexible Language Resources for Brazilian Portuguese. In: *Proceedings of the Workshop on Technology on Information and Human Language (TIL)*. São Leopoldo, Brazil: SBC, 2005. p. 2059–2068.

NADEAU, D.; SEKINE, S. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, John Benjamins publishing company, v. 30, n. 1, p. 3–26, 2007.

NIST. *Text Analysis Conference (TAC)*. 2018. Acesso em: 24/05/2018. Disponível em: <https://tac.nist.gov/2018/index.html>.

PAUMIER, S. *Unitex 3.1 User Manual*. 2016. 377 p. Disponível em: <http://unitexgramlab.org/releases/3.1/man/Unitex-GramLab-3.1-usermanual-en.pdf>.

PELLUCCI, P. R. S. et al. Utilização de Técnicas de Aprendizado de Máquina no Reconhecimento de Entidades Nomeadas no Português. *e-Xacta*, Editora UniBH, v. 4, n. 1, p. 73–81, 2011.

PICOLI, L. et al. Uso de uma Ferramenta de Processamento de Linguagem Natural como Auxílio à Coleta de Exemplos para o Estudo de Propriedades Sintático-semânticas de Verbos. *Linguamática*, v. 7, n. 2, p. 35–44, 2015.

PIRES, A.; DEVEZAS, J.; NUNES, S. Benchmarking Named Entity Recognition Tools for Portuguese. In: *Proceedings of the Ninth INForum: Simpósio de Informática*. Aveiro, Portugal: UA Editora, 2017. p. 111–121.

PIRES, A. R. O. *Named Entity Extraction from Portuguese Web Text*. Dissertação (Mestrado) — Faculdade de Engenharia da Universidade de Porto, Porto, Portugal, 2017.

PIROVANI, J. P. C.; NOGUEIRA, M.; OLIVEIRA, E. Indexing Names of Persons in a Large Dataset of a Newspaper. In: *Villavicencio A. et al. (eds) Computational Processing of the Portuguese Language. PROPOR 2018. Lecture Notes in Computer Science, vol 11122.* Canela, RS: Springer, Cham, 2018. p. 147–155.

PIROVANI, J. P. C.; OLIVEIRA, E. de. Extração de Nomes de Pessoas em Textos em Português: uma Abordagem Usando Gramáticas Locais. In: *Computer on the Beach 2015*. Florianópolis, SC: SBC, 2015. p. 1–10.

PIROVANI, J. P. C.; SPALENZA, M. A.; OLIVEIRA, E. Geração Automática de Questões a Partir do Reconhecimento de Entidades Nomeadas em Textos Didáticos. In: *XXVIII Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE 2017)*. Recife, Brasil: Sociedade Brasileira de Computação - SBC, 2017. v. 28, n. 1, p. 1147–1156.

POOSTCHI, E. Z. B. H.; PICCARDI, M. BiLSTM-CRF for Persian Named-Entity Recognition ArmanPersoNERCorpus: the First Entity-Annotated Persian Dataset. In: CHAIR), N. C. C. et al. (Ed.). *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Paris, France: European Language Resources Association (ELRA), 2018.

RAMESH, S. H. et al. Automatically Identify and Label Sections in Scientific Journals using Conditional Random Fields. In: *Semantic Web Evaluation Challenge*. Grécia: Springer, 2016. p. 269–280.

RATINOV, L.; ROTH, D. Design Challenges and Misconceptions in Named Entity Recognition. In: *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2009. p. 147–155.

ROCHA, C. et al. PAMPO: Using Pattern Matching and Pos-tagging for Effective Named Entities Recognition in Portuguese. 2016. Disponível em: <http://arxiv.org/abs/1612.09535>.

ROCHA, P. A.; SANTOS, D. CETEMPúblico: Um Corpus de Grandes Dimensões de

Linguagem Jornalística Portuguesa. *In Maria das Graças Volpe Nunes (ed) V Encontro para o Processamento Computacional da Língua Portuguesa Escrita e Falada (PROPOR 2000)*, ICMC/USP, São Paulo, 2000.

ROCKTÄSCHEL, T.; WEIDLICH, M.; LESER, U. ChemSpot: A Hybrid System for Chemical Named Entity Recognition. *Bioinformatics*, Oxford University Press, v. 28, n. 12, p. 1633–1640, 2012.

RUSSEL, S.; NORVIG, P. *Inteligência Artificial*. Rio de Janeiro, RJ: Elsevier, 2004.

SANG, E. F. T. K.; MEULDER, F. D. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In: *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2003. p. 142–147.

SANTOS, C. N. d.; GUIMARAES, V. Boosting Named Entity Recognition with Neural Character Embeddings. In: *Proceedings of the Fifth Named Entities Workshop, ACL 2015*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2015. p. 25–33.

SANTOS, C. N. d.; ZADROZNY, B. Learning Character-level Representations for Part-of-Speech Tagging. In: *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*. Beijing, China: ICML, 2014. p. 1818–1826.

SANTOS, C. N. dos; MILIDIÚ, R. L. *Entropy Guided Transformation Learning: Algorithms and Applications*. London, United Kingdom: Springer-Verlag London, 2012.

SANTOS, D.; CARDOSO, N. *Reconhecimento de Entidades Mencionadas em Português: Documentação e Actas do HAREM, a Primeira Avaliação Conjunta na Área*. Linguateca, 2007. 413 p. Disponível em: <http://www.linguateca.pt/aval_conjunta/LivroHAREM/Livro-SantosCardoso2007.pdf>.

SANTOS, J. d. S. *Estudo Comparativo de Diferentes Classificadores baseados em Aprendizagem de Máquina para o Processo de Reconhecimento de Entidades Nomeadas*. Dissertação (Mestrado) — Universidade Estadual de Feira de Santana, Bahia, Brasil, 2017.

SEKER, G. A.; ERYIGIT, G. Initial Explorations on using CRFs for Turkish Named Entity Recognition. In: *Proceedings of COLING 2012*. Mumbai, India: ACL, 2012. p. 2459–2474.

SHAALAN, K. A Survey of Arabic Named Entity Recognition and Classification. *Computational Linguistics*, MIT Press, v. 40, n. 2, p. 469–510, 2014.

SILBERZTEIN, M. *NooJ: A Linguistic Development Environment*. 2018. Acesso em: 10/10/2018. Disponível em: <http://www.nooj4nlp.net/pages/nooj.html>.

SILBERZTEIN, M.; MULLER, C.; ROYAUTÉ, J. NooJ: an Object-Oriented Approach. *INTEX pour la linguistique et le traitement automatique des langues*, Cahiers de la MSH Ledoux. Presses Universitaires de Franche-Comté, p. 359–369, 2004.

SILVA, T. S. d. *Reconhecimento de Entidades Nomeadas em Notícias de Governo*. Dissertação (Mestrado) — Universidade Federal do Rio de Janeiro, RJ, Brasil, 2012.

spaCy. 2019. Acesso em: 18/02/2019. Disponível em: <https://spacy.io/>.

SRIHARI, R.; NIU, C.; LI, W. A Hybrid Approach for Named Entity and Sub-type Tagging. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. *Proceedings of the Sixth Conference on Applied Natural Language Processing*. Seattle, Washington, 2000. p. 247–254.

Stanford CoreNLP. *Stanford CoreNLP – Natural Language Software*. 2019. Acesso em: 18/02/2019. Disponível em: <https://stanfordnlp.github.io/CoreNLP/>.

SUCHANEK, F. M.; KASNECI, G.; WEIKUM, G. Yago: a Core of Semantic Knowledge. In: *Proceedings of the 16th international conference on World Wide Web*. Canadá: ACM, 2007. p. 697–706.

SUTTON, C.; MCCALLUM, A. An Introduction to Conditional Random Fields. *Foundations and Trends® in Machine Learning*, Now Publishers, Inc., v. 4, n. 4, p. 267–373, 2012.

TRABOULSI, H. Arabic Named Entity Extraction: A Local Grammar-Based Approach. In: *Proceedings of the International Multiconference on Computer Science and Information Technology, IMCSIT'09*. Mragowo, Poland: IEEE, 2009. v. 4, p. 139–143.

TRIOLA, M. F. *Introdução à Estatística: Atualização da Tecnologia*. Rio de Janeiro: LTC, 2014.

Unitex. 2018. Acesso em: 02/03/2018. Disponível em: <http://unitexgramlab.org/>.

WEI, C.-H. et al. tmVar: a Text Mining Approach for Extracting Sequence Variants in Biomedical Literature. *Bioinformatics*, Oxford Univ Press, v. 29, n. 11, p. 1433–1439, 2013.

WILLIAMS, L. et al. The Role of Idioms in Sentiment Analysis. *Expert Systems with Applications*, Elsevier, v. 42, n. 21, p. 7375–7385, 2015.

XU, W. et al. *Workshop on Noisy User-generated Text (W-NUT)*. 2018. Acesso em: 03/10/2018. Disponível em: <http://noisy-text.github.io/2018/>.

YANG, J.; ZHANG, Y.; DONG, F. Neural Reranking for Named Entity Recognition. *arXiv preprint arXiv:1707.05127*, 2017.

ZHANG, B. et al. RPI BLENDER TAC-KBP2017 13 Languages EDL System. In: *Proceedings of the Tenth Text Analysis Conference (TAC2017)*. Maryland, USA: NIST, 2017.

ZHOU, G.; SU, J. Named Entity Recognition using an HMM-based Chunk Tagger. In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Philadelphia, PA: Association for Computational Linguistics, 2002. p. 473–480.

# Apêndices

# APÊNDICE A – Teste de Wilcoxon

| $\alpha$ (*Uma cauda*) | **0.005** | **0.010** | **0.025** | **0.05** |
|---|---|---|---|---|
| $\alpha$ (*Duas caudas*) | **0.01** | **0.02** | **0.050** | **0.1** |
| n=5 | - | - | - | 1 |
| 6 | - | - | 1 | 2 |
| 7 | - | 0 | 2 | 4 |
| 8 | 0 | 2 | 4 | 6 |
| 9 | 2 | 3 | 6 | 8 |
| 10 | 3 | 5 | 8 | 11 |
| 11 | 5 | 7 | 11 | 14 |
| 12 | 7 | 10 | 14 | 17 |
| 13 | 10 | 13 | 17 | 21 |
| 14 | 13 | 16 | 21 | 26 |
| 15 | 16 | 20 | 25 | 30 |
| 16 | 19 | 24 | 30 | 36 |
| 17 | 23 | 28 | 35 | 41 |
| 18 | 28 | 33 | 40 | 47 |
| 19 | 32 | 38 | 46 | 54 |
| 20 | 37 | 43 | 52 | 60 |
| 21 | 43 | 49 | 59 | 68 |
| 22 | 49 | 56 | 66 | 75 |
| 23 | 55 | 62 | 73 | 83 |
| 24 | 61 | 69 | 81 | 92 |
| 25 | 68 | 77 | 90 | 101 |

# APÊNDICE B – Como executar as ferramentas

Este apêndice apresenta algumas diretrizes para executar as ferramentas usadas neste trabalho em linha de comando.

## B.1 Segmentação usando o Unitex

Um grafo transdutor (Sentence.fst2) é usado para realizar a segmentação em sentenças no Unitex. Esse grafo descreve os diferentes contextos para o fim de uma sentença e insere a saída {S} ao texto de entrada sempre que um desses contextos é reconhecido. Sendo assim, a segmentação no Unitex é realizada através do programa *Fst2Txt* que aplica o transdutor Sentence.fst2 ao texto.

O comando utilizado para segmentar o arquivo entrada.txt no Unitex (versão 3.1) foi

```
Unitex/App/UnitexToolLogger Fst2Txt -t entrada.txt "<dir-trabalho>/
Portuguese (Brazil)/Graphs/Preprocessing/Sentence/Sentence.fst2" -M
```

onde <dir-trabalho> é o caminho completo do diretório de trabalho pessoal definido no primeiro uso do Unitex. O parâmetro -M indica que o grafo será aplicado no modo MERGE para que a saída seja anexada ao texto.

Dado o texto de entrada abaixo, retirado da CD do Segundo HAREM,

> O software pode ser personalizado, e é oferecido gratuitamente na Web. Cerca de 400 projetos em 30 países estão usando o sistema, afirma Liebenberg.

a saída após a execução da segmentação será

> O software pode ser personalizado, e é oferecido gratuitamente na Web.{S} Cerca de 400 projetos em 30 países estão usando o sistema, afirma Liebenberg.{S}

## B.2 *Tokenization* e *POS-tagging* usando OpenNLP

A tokenization e o *POS-tagging* foram realizados pelo OpenNLP usando seus modelos pré-treinados para o Português.

O comando utilizado para obter os *tokens* com o OpenNLP (versão 1.6.0) foi

```
bin/opennlp TokenizerME pt-token.bin < entrada.txt > saida.txt
```

Usando esse comando, o texto entrada.txt abaixo

> O software pode ser personalizado, e é oferecido gratuitamente na Web. Cerca de 400 projetos em 30 países estão usando o sistema, afirma Liebenberg.

seria transformado no seguinte arquivo saida.txt. Observe que os *tokens* obtidos são separados por espaços.

> O software pode ser personalizado , e é oferecido gratuitamente na Web . Cerca de 400 projetos em 30 países estão usando o sistema , afirma Liebenberg .

De forma semelhante, o comando utilizado para realizar o *POS-tagging* foi

```
bin/opennlp POSTagger pt-pos-perceptron.bin < entrada.txt > saida.txt
```

A saída obtida para os *tokens* do texto de entrada apresentado acima foi

> O_art software_n pode_v-fin ser_v-inf personalizado_v-pcp ,_-punc e_conj-c é_v-fin oferecido_v-pcp gratuitamente_adv na_v-fin Web_prop ._punc Cerca_prop de_prp 400_num projetos_n em_prp 30_num países_n estão_v-fin usando_v-ger o_art sistema_n ,_punc afirma_v-fin Liebenberg_prop ._punc

As etiquetas atribuídas[1] pelo OpenNLP aparecem após o símbolo _.

## B.3 Aplicação de LG no Unitex

Uma LG construída no Unitex possui a extensão .grf. Após compilar a LG, um arquivo .fst2 é obtido. Esse último é utilizado no programa *Locate* que aplica a gramática ao texto e constrói um índice de ocorrências (arquivo *concord.ind*).

O arquivo de entrada para o programa *Locate* deve possuir a extensão .snt que corresponde a um arquivo pré-processado pelo Unitex. O pré-processamento consiste em normalizar os separadores, realizar a segmentação em sentenças, obter os *tokens* e aplicar os

[1] https://visl.sdu.dk/visl/pt/symbolset-floresta.html

dicionários do Unitex. Essas operações são realizadas pelos programas *Normalize*, *Fst2Txt*, *Tfst2Grf* e *Dico*, respectivamente.

Um exemplo de uso do programa *Locate* é apresentado a seguir.

```
Unitex/App/UnitexToolLogger Locate -t entrada.snt «dir-grafos>/
LG.fst2-A -L -M -qutf8-no-bom
```

<dir-grafos> é o caminho completo do diretório onde está o grafo compilado LG.fst2. Os parâmetros usados são:

-A para indexar todas as ocorrências identificadas.

-L para reconhecer as sequências mais longas.

-M para anexar as saídas aos textos de entrada.

Após a chamada do programa *Locate*, o programa *Concord* foi usado para produzir uma concordância a partir do índice de ocorrências obtido pelo *Locate*. Uma versão modificada do texto (com as saídas anexadas) foi produzida usando o parâmetro -m.

Segue um exemplo de uso do programa *Concord*.

```
Unitex/App/UnitexToolLogger Concord entrada_snt/concord.ind -m
saida.txt-qutf8-no-bom
```

Para o texto de entrada

> O software pode ser personalizado, e é oferecido gratuitamente na Web. Cerca de 400 projetos em 30 países estão usando o sistema, afirma Liebenberg.

e considerando a aplicação da LG construída neste trabalho, o arquivo saida.txt produzido após a execução de Locate e Concord foi

> O software pode ser personalizado, e é oferecido gratuitamente na<LOCAL> Web</LOCAL>. <VALOR>Cerca de 400</VALOR> projetos em 30 países estão usando o sistema, afirma<PESSOA> Liebenberg</PESSOA>.

## B.4 CRF na biblioteca MALLET

A interface *SimpleTagger* da biblioteca MALLET (versão 2.0.8) foi utilizada para inferir e aplicar o modelo CRF.

O comando utilizado para inferir o modelo CRF foi

```
java -cp "class:lib/mallet-deps.jar"cc.mallet.fst.SimpleTagger -train
true -model-file modeloCrf treino.txt
```

onde `-train true` especifica que será realizado o treinamento e –model-file especifica o nome do arquivo onde o modelo será salvo (modeloCrf nesse caso). Esse comando realiza o treinamento a partir do arquivo treino.txt (último parâmetro do comando).

Um exemplo de arquivo de treino é apresentado abaixo.

**Quem** pron-indp ini=cap cap=maxmin simb=alpha prevW=null prevT=null prevCap=null nextW=é nextT=v-fin nextCap=min prev2W=null prev2T=null prev2Cap=null next2W=o next2T=art next2Cap=min palpite=O O
**é** v-fin ini=ncap cap=min simb=alpha prevW=Quem prevT=pron-indp prevCap=maxmin nextW=o nextT=art nextCap=min prev2W=null prev2T=null prev2Cap=null next2W=secretário-geral next2T=n next2Cap=null palpite=O O
**o** art ini=ncap cap=min simb=alpha prevW=é prevT=v-fin prevCap=min nextW=secretário-geral nextT=n nextCap=null prev2W=Quem prev2T=pron-indp prev2Cap=maxmin next2W=da next2T=v-pcp next2Cap=min palpite=O O
**secretário-geral** n ini=ncap cap=null simb=null prevW=o prevT=art prevCap=min nextW=da nextT=v-pcp nextCap=min prev2W=é prev2T=v-fin prev2Cap=min next2W=UGT next2T=prop next2Cap=max palpite=O O
**da** v-pcp ini=ncap cap=min simb=alpha prevW=secretário-geral prevT=n prevCap=null nextW=UGT nextT=prop nextCap=max prev2W=o prev2T=art prev2Cap=min next2W=? next2T=punc next2Cap=null palpite=O O
**UGT** prop ini=cap cap=max simb=alpha prevW=da prevT=v-pcp prevCap=min nextW=? nextT=punc nextCap=null prev2W=secretário-geral prev2T=n prev2Cap=null next2W=null next2T=null next2Cap=null palpite=I_ORGANIZACAO I_ORGANIZACAO
**?** punc ini=simb cap=null simb=null prevW=UGT prevT=prop prevCap=max nextW=null nextT=null nextCap=null prev2W=da prev2T=v-pcp prev2Cap=min next2W=null next2T=null next2Cap=null palpite=O O

Nesse arquivo, cada linha representa um *token* que aparece em negrito na primeira coluna, seguido de suas características e o seu rótulo correto (*tag*) na última coluna. Todos eles separados por um espaço. Uma linha em branco deve existir entre cada sentença.

O comando utilizado para aplicar o modelo CRF (modeloCrf) ao arquivo teste.txt foi

```
java -cp "class:lib/mallet-deps.jar"cc.mallet.fst.SimpleTagger -model-file
modeloCrf teste.txt > saida.txt
```

Um exemplo de arquivo enviado para rotulação é apresentado a seguir. Observe que esse arquivo é semelhante ao enviado para treino, mas não possui a última coluna de

rótulos pois esses devem ser atribuídos pelo modelo.

**Quem** pron-indp ini=cap cap=maxmin simb=alpha prevW=null prevT=null prevCap=null nextW=é nextT=v-fin nextCap=min prev2W=null prev2T=null prev2Cap=null next2W=o next2T=art next2Cap=min palpite=O
**é** v-fin ini=ncap cap=min simb=alpha prevW=Quem prevT=pron-indp prevCap=maxmin nextW=o nextT=art nextCap=min prev2W=null prev2T=null prev2Cap=null next2W=secretário-geral next2T=n next2Cap=null palpite=O
**o** art ini=ncap cap=min simb=alpha prevW=é prevT=v-fin prevCap=min nextW=secretário-geral nextT=n nextCap=null prev2W=Quem prev2T=pron-indp prev2Cap=maxmin next2W=da next2T=v-pcp next2Cap=min palpite=O
**secretário-geral** n ini=ncap cap=null simb=null prevW=o prevT=art prevCap=min nextW=da nextT=v-pcp nextCap=min prev2W=é prev2T=v-fin prev2Cap=min next2W=UGT next2T=prop next2Cap=max palpite=O
**da** v-pcp ini=ncap cap=min simb=alpha prevW=secretário-geral prevT=n prevCap=null nextW=UGT nextT=prop nextCap=max prev2W=o prev2T=art prev2Cap=min next2W=? next2T=punc next2Cap=null palpite=O
**UGT** prop ini=cap cap=max simb=alpha prevW=da prevT=v-pcp prevCap=min nextW=? nextT=punc nextCap=null prev2W=secretário-geral prev2T=n prev2Cap=null next2W=null next2T=null next2Cap=null palpite=I_ORGANIZACAO
**?** punc ini=simb cap=null simb=null prevW=UGT prevT=prop prevCap=max nextW=null nextT=null nextCap=null prev2W=da prev2T=v-pcp prev2Cap=min next2W=null next2T=null next2Cap=null palpite=O

# APÊNDICE C – Exemplos de LGGs

Este apêndice apresenta alguns exemplos de LGGs construídos neste trabalho. A LG completa está disponível em:

<https://inf.ufes.br/~elias/dataSets/ner/recursosTese-julianaPirovani.zip>

## C.1 Categoria Pessoa

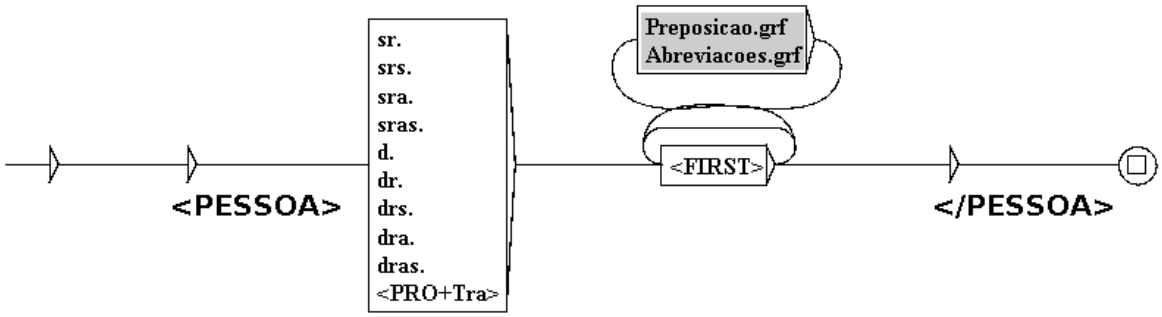


Figura 33 – Regra no grafo que reconhece a categoria Pessoa

LGG que reconhece nomes antecedidos por formas de tratamento. São identificadas formas de tratamento como *Sr.*, *Dra.* e outras através do código <PRO+Tra> do Unitex como *senhor* e *excelência* seguidas por palavras iniciadas com letra maiúscula identificadas pelo código <FIRST>. Preposições e abreviações podem aparecer entre as palavras iniciadas com letra maiúscula. Note que as formas de tratamento fazem parte do nome identificado.

Ocorrências identificadas por esse LGG:
*...a morte de <PESSOA>D. Afonso Henriques</PESSOA>?*
*...de que o <PESSOA>senhor Javier Solanas</PESSOA> é o exemplo...*

## C.2 Categoria Local

Esse LGG reconhece nomes de lugares antecedidos pela preposição *em*. Tempo.grf entre ![ e ] indica um contexto negativo. Ele é usado para identificar como Local, as palavras iniciadas com letra maiúscula (<FIRST>) que não tenham sido reconhecidas pelo LGG Tempo.grf. Isso evita o reconhecimento de NEs da categoria Tempo após a preposição *em* como na sentença *batalha dos Atoleiros em Abril*. Observe que, nesse grafo, a preposição *em* é apenas uma evidência externa e não faz parte da NE reconhecida.

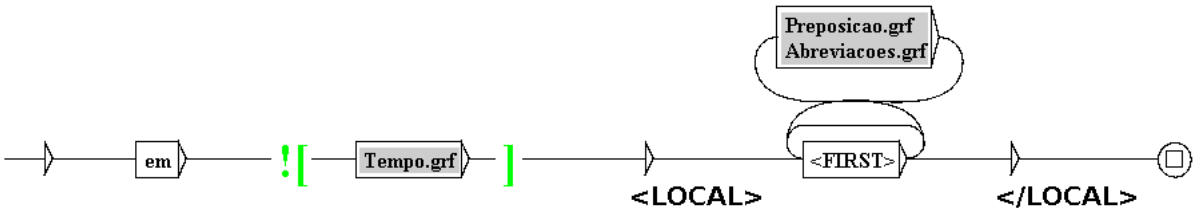


Figura 34 – Regra no grafo que reconhece a categoria Local

Exemplos de ocorrências identificadas por esse grafo foram:
*...colonização portuguesa em <LOCAL>Angola</LOCAL>...*
*...que apareceu em <LOCAL>Porto Alegre</LOCAL> nos anos 90...*

## C.3 Categoria Organização


Figura 35 – Regra no grafo que reconhece a categoria Organização

O LGG acima possui dois caminhos: no primeiro, as palavras *Universidade*, *Organização* e *Secretaria* são reconhecidas como parte da NE da categoria Organização; no segundo, essas palavras são utilizadas apenas como evidência externa e não fazem parte da NE reconhecida. <LOWER> é o código para reconhecer letras minúsculas no Unitex e foi usado em um contexto (entre [ e ]) para garantir que as palavras *universidade*, *organização* e *secretaria* reconhecidas iniciem com letra minúscula no segundo caminho. No grafo que compõe a LG desse trabalho, além dessas três palavras, outras são usadas como evidências.

Ocorrências identificadas pelo LGG foram:
*...com estudantes da <ORGANIZACAO>Universidade de Copenhague</ORGANIZACAO>*
*Quem é o líder da organização <ORGANIZACAO>Pro Familia</ORGANIZACAO>?*

## C.4 Categoria Tempo

Esse LGG reconhece NEs da categoria Tempo que contém uma sequência de dígitos, identificados pelo código <NB> do Unitex, seguidos de meses cujo reconhecimento foi

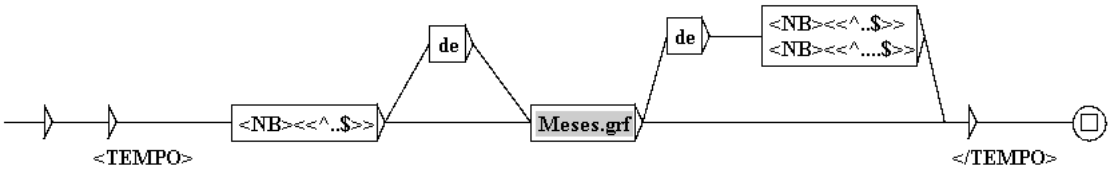


Figura 36 – Regra no grafo que reconhece a categoria Tempo

previamente detalhado no grafo Meses.grf. Entre o número representando o dia e o mês pode aparecer a preposição *de*. Essa preposição também pode aparecer após o mês seguida por um outro número representando o ano. <<^..$ >> e <<^....$ >> após <NB> denota a aplicação de um filtro morfológico ao número identificado indicando que ele deve ter dois dígitos ou quatro dígitos respectivamente.

Ocorrências identificadas pelo LGG foram:
*...<TEMPO>27 Março de 2005</TEMPO>. OBS: no início de um documento.*
*...foi conduzido em <TEMPO>19 de maio de 1566</TEMPO> com pompa.*

## C.5 Categoria Valor

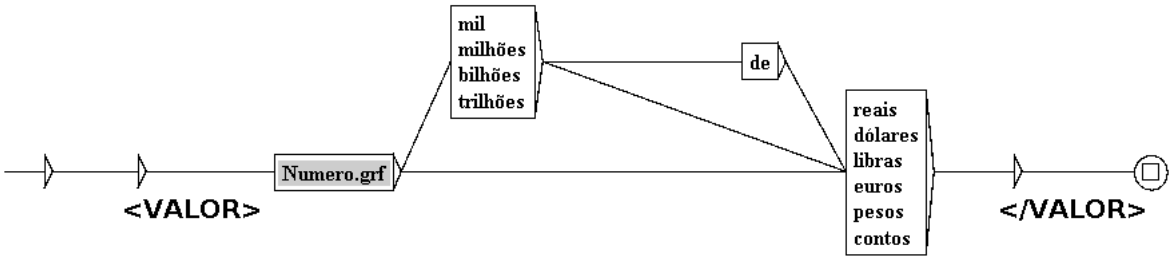


Figura 37 – Regra no grafo que reconhece a categoria Valor

A regra apresentada nesse LGG reconhece números utilizando o LGG Numero.grf (que inclui o reconhecimento de números com pontos decimais) seguidos de palavras como *reais*, *dólares* e *pesos*. Entre o número e uma dessas palavras podem aparecer expressões como *mil de* e *bilhões de*.

Seguem alguns exemplos de ocorrências identificadas:
*...por <VALOR>50 milhões de libras</VALOR> em 1998.*
*...se fixou em <VALOR>1,2556 dólares</VALOR>, segundo dados...*

## C.6 Categoria Abstração

Esse grafo reconhece palavras como *denominada* e *chamava* seguidas por palavras iniciadas com letra maiúscula que representam NEs da categoria Abstração. Preposições e

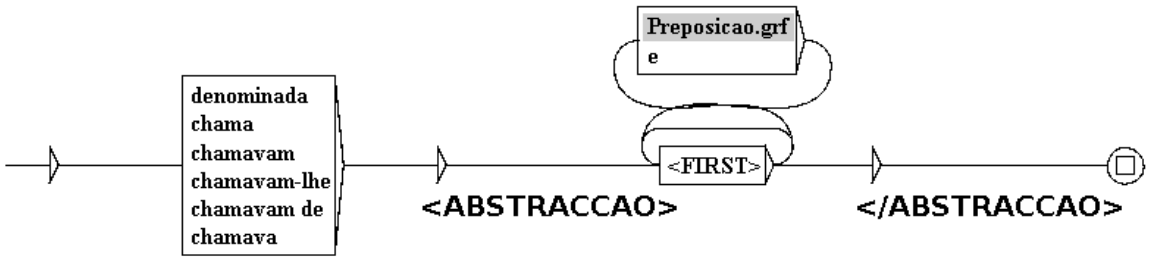


Figura 38 – Regra no grafo que reconhece a categoria Abstração

a conjunção *e* podem fazer parte dessas NEs.

Exemplos de ocorrências identificadas:
*Como se chama <ABSTRACCAO>Cayenne</ABSTRACCAO> em português?*
*A revista foi denominada <ABSTRACCAO>Medicina e Cultura</ABSTRACCAO>*

## C.7 Categoria Acontecimento

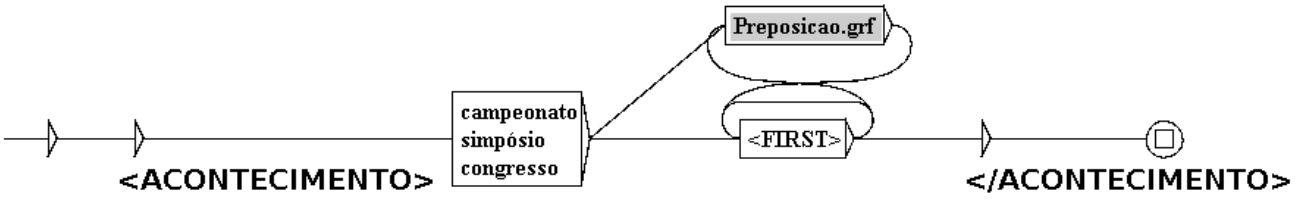


Figura 39 – Regra no grafo que reconhece a categoria Acontecimento

Esse LGG reconhece palavras iniciadas com letra maiúscula antecedidas por palavras como *campeonato*, *simpósio* e *congresso*. Outras palavras são usadas como evidências na LG construída neste trabalho.

Ocorrências identificadas pelo LGG foram:
*...Itália-Nigéria no <ACONTECIMENTO>Campeonato do Mundo</ACONTECIMENTO> de 1994 ?*
*...<ACONTECIMENTO>Congresso Brasileiro de Advocacia Pública</ACONTECIMENTO> foi promovido?*

## C.8 Categoria Obra

Esse LGG reconhece palavras iniciadas com letra maiúscula antecedidas por palavras como *livro*, *canção* e *filme*. Essas palavras são evidências externas que indicam a presença de NEs da categoria Obra.

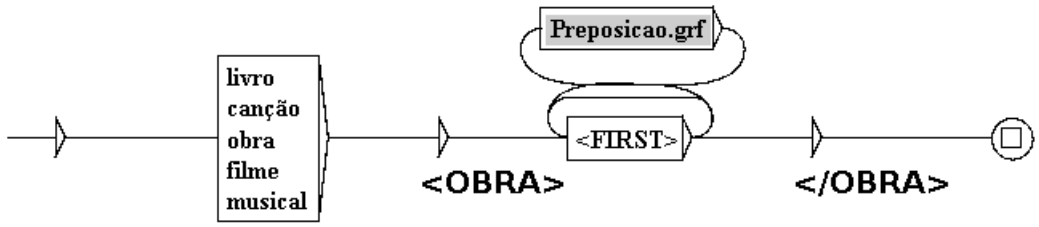


Figura 40 – Regra no grafo que reconhece a categoria Obra

Ocorrências identificadas pelo LGG foram:
*Qual a duração do filme <OBRA>A Nona Porta</OBRA>?*
*...do seu livro <OBRA>Seis Propostas</OBRA> para o...*

## C.9 Categoria Coisa

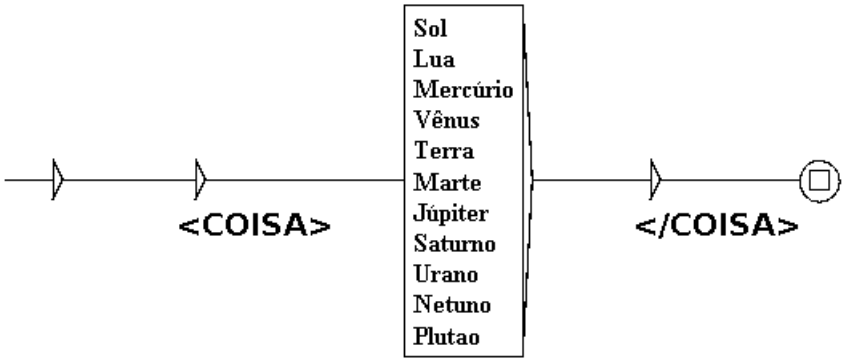


Figura 41 – Regra no grafo que reconhece a categoria Coisa

Como explicado na Seção 5.1, o LGG criado para categoria Coisa reconhece apenas algumas palavras como *Sol*, *Terra* e *Plutão*.

Exemplos de ocorrências identificadas por esse grafo foram:
*...e dos anéis de <COISA>Saturno</COISA>, diretamente...*
*Como a volta à <COISA>Terra</COISA> não foi ...*

## C.10 Categoria Outro

Essa regra reconhece NEs da categoria Outro iniciadas pelas palavras *Prémio*, *Prêmio* ou *Premio*. Preposições e a conjunção *e* podem compor a NE.

Ocorrências identificadas pelo LGG foram:
*Quem recebeu o <OUTRO>Prémio Nobel da Literatura</OUTRO> nesse ano?*
*Com que <OUTRO>Prémio Nobel</OUTRO> é que Seamus Heaney...*

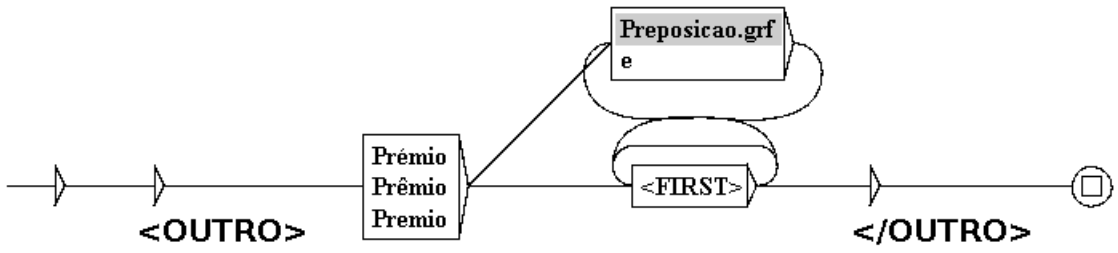


Figura 42 – Regra no grafo que reconhece a categoria Outro