CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA CELSO SUCKOW DA FONSECA

DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO

DEPARTAMENTO DE PESQUISA

COORDENADORIA DE PESQUISA E ESTUDOS TECNOLÓGICOS

RELATÓRIO FINAL DE INICIAÇÃO CIENTÍFICA

EXPLORANDO AS FUNÇÕES DE ROTULAGEM DO SNORKEL PARA TEXTOS DA ÁREA DE FARMACOVIGILÂNCIA

**Edital PIBIC 2019**

**Aluno:**

Leandro Zirondi de Sousa

Bacharelado em Ciência da Computação / 4° Período

Aluno: Bolsista CEFET/RJ

**Professora Orientadora:**

Kele Teixeira Belloze, D. Sc.

Rio de Janeiro, RJ - Brasil

09 / 2020

**RESUMO**

Nos últimos anos o avanço em aprendizado de máquina tem se dado em parte a *datasets* rotulados. Tais *datasets*, porém ainda são extremamente custosos de se produzir. Este trabalho apresenta um estudo sobre a ferramenta Snorkel, uma ferramenta capaz de automatizar este processo de rotulamento e procura analisar sua viabilidade para o uso na língua portuguesa visando apoiar o serviço de farmacovigilância do país. A metodologia deste trabalho envolveu o entendimento e execução da ferramenta Snorkel, o uso de *datasets* da área de farmacovigilância e a busca em *tweets* de relatos de eventos adversos causados por medicações. Verificou-se que por mais que haja problemas com a ferramenta devido às suas diferentes versões, sua instalação e execução e ainda são possíveis. Viu-se também que para o uso da língua portuguesa é necessária a criação ou adaptação de *datasets* para que os mesmos estejam somente em português, assim como o desenvolvimento de funções de rotulagem que se adaptem aos dados do domínio. A partir desses resultados, pode-se considerar que o uso da ferramenta é viável para ser utilizada na língua portuguesa e assim desenvolver tarefas para apoiar a área de farmacovigilância do país.

# 1 INTRODUÇÃO

Um dos motivos do grande avanço na área de aprendizado de máquina que temos visto nos últimos anos é dado ao uso de grandes *datasets* rotulados (SEN, SHRIVASTAVA, *et al.*, 2017). Porém, ao mesmo tempo, tais *datasets* são extremamente custosos de se desenvolver. Além de necessitarem um grande volume de dados, a sua rotulação também é trabalhosa, muitas vezes necessitando a disponibilidade de especialistas de domínio e demandando um tempo extenso para a sua rotulação devido ao seu volume (RATNER, BACH, *et al.*, 2017).

O Snorkel é uma ferramenta desenvolvida pelo *snorkel-team*, com o intuito de resolver o problema da rotulação de grandes *datasets* para o uso em sistemas de aprendizado de máquina. Sua ideia principal é ser um sistema de rotulação de “*weak supervision”*, um sistema onde a rotulação de dados é automatizada extraindo tais informações de textos em linguagem natural. (RATNER, BACH, *et al.*, 2018)

Uma área que já se beneficia com esse extração de informações da linguagem natural é a farmacovigilância (CHEN, FAVIEZ, *et al.*) (REDE PAN-AMERICANA DE HARMONIZAÇÃO DA REGULAMENTAÇÃO FARMACÊUTICA, 2011). A farmacovigilância é a ciência e atividades relativas à detecção, compreensão e prevenção de eventos adversos relacionados a medicamentos, como define a Organização Mundial de Saúde (OMS). Infelizmente no Brasil o sistema de farmacovigilância ainda é fraco, havendo falta de incentivo tanto por parte dos profissionais da saúde quanto por parte da infraestrutura do sistema de saúde. (REDE PAN-AMERICANA DE HARMONIZAÇÃO DA REGULAMENTAÇÃO FARMACÊUTICA, 2011)

Nessa perspectiva, diante da deficiência do sistema de farmacovigilância brasileiro, o uso destes sistemas de extração de informação, dentro do domínio da língua portuguesa, pode ajudar a fortalecer a área de farmacovigilância no país.

Portando, se questiona: o uso da ferramenta Snorkel para rotulação de *datasets* é possível no contexto da língua portuguesa e desta forma ser aplicada na área de farmacovigilância?

Então, o objetivo geral da presente pesquisa é utilizar a ferramenta Snorkel, contextualizada na língua portuguesa, para auxiliar a área de farmacovigilância do país.

Para tanto, foram então definidos os seguintes objetivos específicos: entender o funcionamento da ferramenta; verificar a viabilidade da ferramenta no domínio da Língua Portuguesa; aplicar as funcionalidades da ferramenta em textos no contexto de farmacovigilância.

# 2. METODOLOGIA

## 2.1 Snorkel

A ferramenta Snorkel se estrutura em quatro etapas principais: leitura do *corpus* e geração de sentenças, definição das relações e os candidatos, criação das *Labeling Functions* e *Gold Labels* e, por fim, o treinamento do modelo.

## 2.2 O Corpus e as Sentenças

A primeira parte da execução vem com a leitura do corpus e sua divisão em sentenças. Para o processo de leitura, a tarefa é simplesmente ler o arquivo do disco para a memória. Na questão da divisão de sentenças, a ferramenta usa o *Spacy*, uma biblioteca para a linguagem Python para o processamento de linguagem natural, para gerar essas sentenças. Esse processo é feito numa função *CorpusParser* na qual o *corpus* será dividido em documentos, cada entrada no conjunto do *corpus*, e depois em sentenças. Nesse processamento, é criado então objetos na memória do tipo *Document* que são tuplas de *Sentences*. São com esses objetos que o processamento do *Snorkel* irá ocorrer.

Essas sentenças são então divididas aleatoriamente em três subconjuntos: um conjunto de treino, um conjunto de desenvolvimento e um conjunto de testes. O conjunto de treino é o conjunto que será usado para treinar o modelo ao final da execução das etapas da ferramenta, enquanto os outros dois são conjuntos de suporte para testes. Todos os conjuntos também são salvos em um banco de dados para evitar a reexecução desnecessária dessa tarefa.

## 2.3 A Relação e os Candidatos

Tendo as sentenças prontas, a próxima tarefa é a definição da relação e dos candidatos. A relação é o esquema do relacionamento que se quer extrair do *corpus*. Uma relação, para o Snorkel, é um esquema de relacionamento entre n-partes de uma sentença onde pode-se conter a informação procurada no texto. Essas n-partes do texto são os candidatos, são os elementos que se relacionam e inferem a informação extraída.

Por exemplo, nos tutoriais fornecidos pelos desenvolvedores, a relação procurada era a de casamento. Então o esquema dessa relação é uma relação binária entre dois candidatos, candidatos esses sendo pessoas que em texto são diferenciadas por nomes. Neste caso, como os candidatos são pessoas, a procura por eles em sentenças acaba se tornando uma procura por nomes, porém, cada esquema de relação acaba sendo diferente para cada necessidade de uso da ferramenta. O Snorkel então fornece um conjunto de funções *matchers* que sãofunções auxiliares que permitem o usuário definir seu conjunto de candidatos, seja ele pessoas, *strings* de texto específicas, datas, locais, expressões regulares etc.

Para poder procurar por esses candidatos nas sentenças, um extrator de candidatos então é definido. Tal extrator é formado pela relação entre as funções *matchers* dos candidatos. Esse extrator então percorre pelas sentenças dos documentos do *corpus* procurando por esses candidatos para achar possíveis menções da relação procurada.

## 2.4 Labeling Functions e Gold Labels

Tendo essas possíveis relações, a próxima etapa do processamento é avaliar se essas relações achadas no *corpus* são relações verdadeiras. Para isso o *Snorkel* utiliza dois métodos: as *Labeling Functions* e as *Gold Labels*.

### 2.4.1 *Labeling Functions*

As *Labeling Functions* são divididas em duas categorias, padrão e *distant supervision*. A categoria padrão representa funções, definidas pelo usuário, que o candidato é avaliado por meio da estrutura da sentença. Já a *distant supervision* é um conjunto de exemplos de elementos de relações válidas que é alimentado na ferramenta. Essas funções já tentam validar os candidatos a partir de um conhecimento prévio expresso nesse conjunto de exemplos. Ambos os tipos de *Labeling Functions* podem avaliar os candidatos em três maneiras: positivo, há uma relação válida aqui; negativo, não há uma relação válida aqui e abstinência, onde a função não consegue indicar nada.

Para a verificação da efetividade dessas funções, o *Snorkel* fornece um conjunto de funções auxiliares que permitem o usuário testar a acurácia dessas funções. Nesse processo, as *Labeling Functions* são treinadas e aplicadas num modelo de classificação para terem sua acurácia avaliada entre si para eventualmente virarem apenas um modelo classificador.

### 2.4.2 Gold Labels

As *Gold Labels* são sentenças extraídas do *corpus* que são rotuladas a mão pelo desenvolvedor para exemplificar sentenças exemplos de relação positiva e/ou negativa para o modelo final. Existem também funções auxiliares fornecidas pela ferramenta para ajudar o usuário a fazer essa rotulação manual.

## 2.5 Treinamento do Modelo

Por fim, tendo os candidatos separados, a *Labeling Functions* num estado satisfatório e as *Gold Labels* separadas, o usuário consegue treinar o modelo final para a extração dos candidatos em que a relação se mostrou positiva.

# 3. RESULTADOS

Os primeiros desafios com a ferramenta surgiram logo com sua instalação. A versão utilizada neste projeto, 0.7 ou *Snorkel Extraction*, foi a última versão antes de uma mudança consideravelmente grande no *release* seguinte da ferramenta, causando problemas tanto em quesito de documentação quanto de suporte.

Por tentativa e erro de várias instalações, logo descobriu-se certas necessidades para que a instalação e execução da ferramenta ocorresse adequadamente: o requerimento do sistema operacional Linux, a utilização de um ambiente de desenvolvimento Python isolado, e a instalação de bibliotecas da linguagem Python que não estavam sendo instaladas durante o processo de instalação do Snorkel.

Como mencionado antes, um dos problemas que surgiram foi a falta de documentação disponível para a versão utilizada. A documentação disponibilizada estava desatualizada, com informações referentes à versão 0.6, tornando a consulta para a procura de funções e métodos confusa, mostrando resultados que continham pedaços de códigos alterados ou removidos da versão mais atual. Felizmente, boa parte da documentação ainda pôde ser utilizada e ajudou no entendimento da ferramenta.

Com a instalação concluída, a próxima etapa foi estudar os tutoriais fornecidos para entender a execução da ferramenta e a sua aplicação considerando o nosso domínio da língua portuguesa e da farmacovigilância. Esse estudo dos tutoriais ajudou a entender os passos necessários para uma execução completa da ferramenta, trazendo o entendimento descrito na sessão de metodologia.

Com esse entendimento em mãos, começou a tarefa de entender quais informações do domínio seriam necessárias para conseguirmos executar o *Snorkel*. Pesquisando pela documentação, foi encontrado uma função do tipo *matcher* que consegue identificar relações procuradas com dicionários (um conjunto de palavras). Como essa função é alimentada com dicionários criados pelo usuário, a utilização pela língua portuguesa é garantida, já que basta o usuário criar esses dicionários e utilizar um *corpus* em português. Já o conteúdo desses dicionários depende do que está sendo procurado: no caso desta pesquisa, uma menção de que um medicamento causou um evento adverso. Sendo assim, viu-se necessário o uso de dois dicionários, um populado com nomes de medicamentos e outro populado com eventos adversos, os quais foram extraídos do trabalho de (DA CUNHA, 2019).

O corpus também foi extraído de (DA CUNHA, 2019), tendo sido um conjunto de *tweets*, postagens de até 140 caracteres, geralmente em texto informal, extraídas da plataforma Twitter entre os anos de 2016 e 2017. Essa extração procurou por tweets que mencionavam substâncias, extraindo então apenas o texto do tweet e salvando-o em um arquivo de texto.

O *corpus* é alimentado ao programa em formato .csv (*comma separated value*, valor separado por vírgula), porém como o conteúdo dos *tweets* pode conter vírgulas, optou-se pelo uso do formato de arquivo .tsv (*tab separeted value*, valor separado por tabulação), já que ele pode ser intercalado com arquivos .csv sem problemas. A fonte desse conteúdo veio de um arquivo .txt, para o qual foi necessário um processamento de texto para garantir o formato adequado transformando as entradas Capitulares e MAÍUSCULAS para minúsculas. O código respectivo encontra-se no Anexo I.

O primeiro dicionário, o de medicamentos, foi gerado a partir da junção de quatro arquivos disponibilizados por (DA CUNHA, 2019), dois contendo o nome químico dos medicamentos, como Paracetamol, e outros dois contendo um equivalente comercial, como Tylenol. Após essa junção, foram filtradas as entradas Capitulares e MAÍUSCULAS. O código respectivo encontra-se no Anexo II.

Já o dicionário de eventos foi apenas filtrado as entradas Capitulares e MAÍUSCULAS. O código respectivo encontra-se no Anexo III.

Ambos os dicionários são carregados como *list*, uma estrutura de dados da linguagem Python. Até é possível salvar uma *list* como um arquivo para uso posterior, porém para facilidade de leitura humana, decidiu-se usar o formato .txt para salvar esses dicionários e usar um script para popular as *lists* em memória com o conteúdo desse arquivo. O código respectivo encontra-se no Anexo IV.

Tendo o corpus e ambos os dicionários prontos, já é possível executar a primeira parte do Snorkel e extrair candidatos. Para isso, é definida uma função *matcher* para cada dicionário e logo depois é definida a relação entre eles. Essa relação é definida em um extrator de candidatos. Como na língua portuguesa pode-se dizer tanto “O medicamento X me causou o evento Y” quanto “O evento Y foi causado pelo medicamento X”, caracteriza-se uma relação é simétrica. Felizmente, o Snorkel está preparado para lidar com relações simétricas, caso o contrário, seria necessária uma execução para “O medicamento X me causou o evento Y” e outra para “O evento Y foi causado pelo medicamento X”.

Com os candidatos em mãos, o foco passa agora para as *Labeling Functions.* Mas antes do desenvolvimento delas, foi visto a necessidade de se entender melhor os textos que estão sendo avaliados nos candidatos. Como o *corpus* é composto de *tweets*, os candidatos também serão *tweets*, e a linguagem informal é majoritariamente usada no *Twitter*, então um script para analisar as palavras que aparecem com mais frequência foi desenvolvido para buscar um entendimento melhor da linguagem.

# 4. DISCUSSÃO

Com esse problema da linguagem informal nos candidatos, é interessante trazer ferramentas fora do contexto do Snorkel para um auxílio nos próximos passos. O LIWC (*Linguistic Inquiry and Word Count*) em português pode ser utilizado para verificar quais palavras e ideias estão sendo descritas entre os candidatos de uma sentença, podendo trazer então um filtro para palavras e ideias que confirmam ou negam a relação na sentença. O LIWC poderia também ser utilizado numa *Labeling Function*, cruzando referência com verbos e palavras que trazem a ideia de causalidade.

A estrutura da língua portuguesa também precisa ser analisada para verificar as possibilidades de funções mais baseadas em estrutura da língua. Uma ideia já existente refere-se ao contexto de alergias, pois, no português, se você tem alergia, tem alergia a alguma coisa. Essa estrutura “alergia a” se mostrou útil em testes preliminares com candidatos nos quais a palavra alergia aparecia. A análise da estrutura da língua pode levantar alguns problemas, porque o *corpus* é totalmente formado por *tweets*, que por sua vez estão numa linguagem informal.

Há uma questão dos *datasets* utilizados também, o de substâncias e o de eventos. A falta desses conteúdos em português pode ser um sinal de problema, pois será necessário comparar os resultados do modelo com as bulas dessas substâncias para saber se o evento é documentado ou não. Essa comparação pede um *dataset* robusto de substâncias e eventos causados por elas. Existem sites como o bulario.com.br onde é fornecida informações diversas sobre várias substâncias em português, mas *datasets* já processados e disponibilizados sobre o assunto não foram encontrados. Essa questão poderia ser resolvida com um *web* *scraping* para gerar um *dataset* de substância-efeito colateral, e que seria bastante relevante para apoiar diversas pesquisas na área de farmacovigilância no Brasil.

# 5. CONCLUSÃO

Com o objetivo de analisar a viabilidade do uso da língua portuguesa com a ferramenta Snorkel, esse trabalho buscou utilizar o domínio de farmacovigilância para investigar este uso.

Mesmo com a confirmação desta viabilidade, ainda se vê uma limitação com a falta de *datasets* na língua portuguesa do domínio, como verificada com o trabalho de (DA CUNHA, 2019). A versão utilizada da ferramenta, mesmo havendo mais suporte e desenvolvimento, ainda se mostra extremamente útil para resolver o problema da rotulação de grandes volumes de dados.

Apesar de ter sido verificada a possibilidade do uso do Snorkel, ainda se faz necessária a continuidade da pesquisa sobre os *datasets* reais obtidos, o estudo e desenvolvimento de *Labeling Functions* e *Golden Labels*, assim como a execução dos modelos de aprendizado de máquina para extração dos candidatos.

# 6. REFERÊNCIAS BIBLIOGRÁFICAS

CHEN, X. et al. **Mining Patients' Narratives in Social Media for Pharmacovigilance: Adverse Effects and Misuse of Methylphenidate**. [S.l.].

DA CUNHA, A. M. **DETECÇÃO DE SINAIS DE EVENTOS ADVERSOS DE MEDICAMENTOS EM TEXTOS INFORMAIS**. Centro Federal de Educação Tecnológica Celso Suckow da Fonseca. Rio de Janeiro, p. 134. 2019.

RATNER, A. et al. **Snorkel: Rapid Training Data Creation**. [S.l.]. 2017.

RATNER, A. et al. **Snorkel: Rapid Training Data Creation**. Stanford University. Stanford, p. 14. 2018.

REDE PAN-AMERICANA DE HARMONIZAÇÃO DA REGULAMENTAÇÃO FARMACÊUTICA. **Boas práticas de farmacovigilância para as Américas**. [S.l.]. 2011.

SEN, C. et al. **Revisiting Unreasonable Effectiveness of Data in Deep Learning Era**. [S.l.]. 2017.

ZHANG, C. et al. **DeepDive: Declarative Knowledge Base Construction**. [S.l.].

# 7. AGRADECIMENTOS

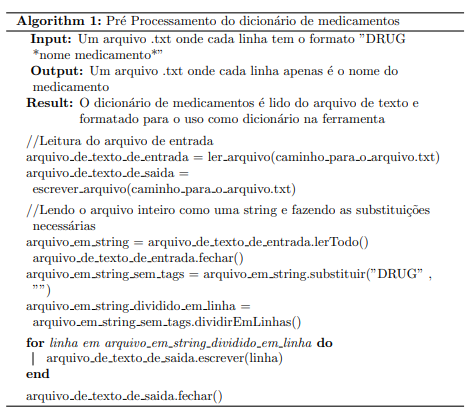
O autor agradece ao CEFET-RJ pelo apoio no desenvolvimento desta pesquisa. Ao professor Gustavo Paiva Guedes e Silva, D. SC pelo apoio e colaboração na presente pesquisa. A professora Kele pela paciência e orientação. A Stephanie, Leonardo, Marcela e Myllena por todo apoio do mundo.

# 8. Anexos

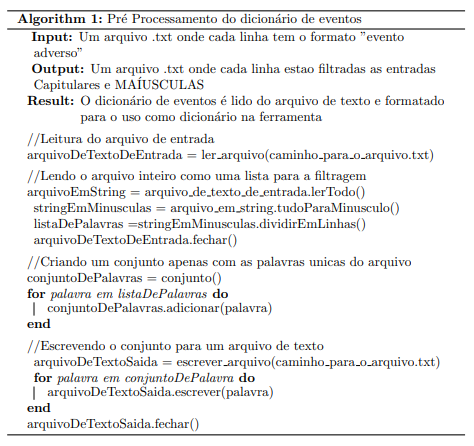
## ANEXO I

## 

## ANEXO II



## ANEXO III



## ANEXO IV

## 