

ANATEL  
AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES

**PROPOSTA DE MELHORIAS PARA A  
IMPLEMENTAÇÃO, ORGANIZAÇÃO E  
IMPLANTAÇÃO DA SOLUÇÃO – RESOLUÇÕES**

MARCELLO SANDI PINHEIRO

PO: Thiago Pereira de Brito Vieira

Brasília, outubro de 2020

## SUMÁRIO

<b>INTRODUÇÃO.....</b>	<b>3</b>
<b>CAPÍTULO 2 – RESOLUÇÕES E SIMILARIDADE.....</b>	<b>8</b>
2.1 – ENTENDIMENTO SOBRE AS RESOLUÇÕES.....	8
2.2 – AVALIAÇÃO DA IMPLEMENTAÇÃO DO RELATÓRIO DE RESOLUÇÕES.....	12
2.3 – TRANSFER LEARNING COM AJUSTE FINO NO BERT.....	13
2.4 – SENTENCE-BERT – SENTENCE EMBEDDINGS USING SIAMESE BERT-NETWORKS.....	16
<b>CAPÍTULO 3 - ABORDAGEM PROPOSTA.....</b>	<b>17</b>
3.1- PROPOSTA.....	17
3.1.1 – Descoberta de tópicos.....	17
3.1.2 – Busca por similaridade semântica.....	18
<b>CONCLUSÃO.....</b>	<b>19</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>20</b>

## INTRODUÇÃO

*“A missão da Anatel é promover o desenvolvimento das telecomunicações do País de modo a dotá-la de uma moderna e eficiente infraestrutura de telecomunicações, capaz de oferecer à sociedade serviços adequados, diversificados e a preços justos, em todo o território nacional”* em [PORTAL, 2020].

Corroborando com a missão, o Relatório Técnico ANATEL [ANATEL, 2019] analisou quantitativamente 687 Resoluções e nele é explicado que a Anatel tem por diretriz a **simplificação** regulatória e por objetivo a **otimização** e a **transparência** da Agência e sua relação com a **sociedade**, regulando e avaliando a complexidade do conjunto normativo expedido pela Agência.

O tripé simplificação, otimização e transparência são diretrizes da Anatel, por outro lado, a [MINUTA, 2020] dispõe sobre desburocratização de procedimentos administrativos e mostra que Governo Federal tem como meta gerar economia e aproveitar melhor os

recursos financeiros e pessoais, e ambos possuem o mesmo objetivo em comum que é atender melhor a sociedade.

Disposto disso, em [ANATEL, 2019], se observa que do total de 687 Resoluções 329 (48%) estão vigentes, sendo que 123 Resoluções foram responsáveis pela revogação de outras 358 (52%). A figura 1 mostra a Rede de Revogações onde se observa que duas Resoluções foram responsáveis por revogar a maioria.

Somente no atual Conselho da Anatel, tendo como signatário Leonardo Euler de Moraes, figura 2, foi responsável pela revogação de 173 Resoluções. No Conselho anterior foram revogadas 54 e antes desse, foram 36 Resoluções revogadas, assinalando a tendência em simplificar, mas, também, de conhecer eventuais interdependências entre as Resoluções [ANATEL, 2019].

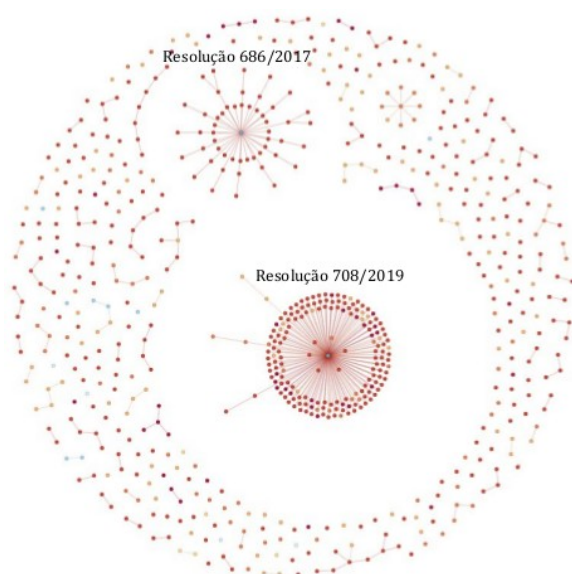


Figura 1 – Rede de Revogações

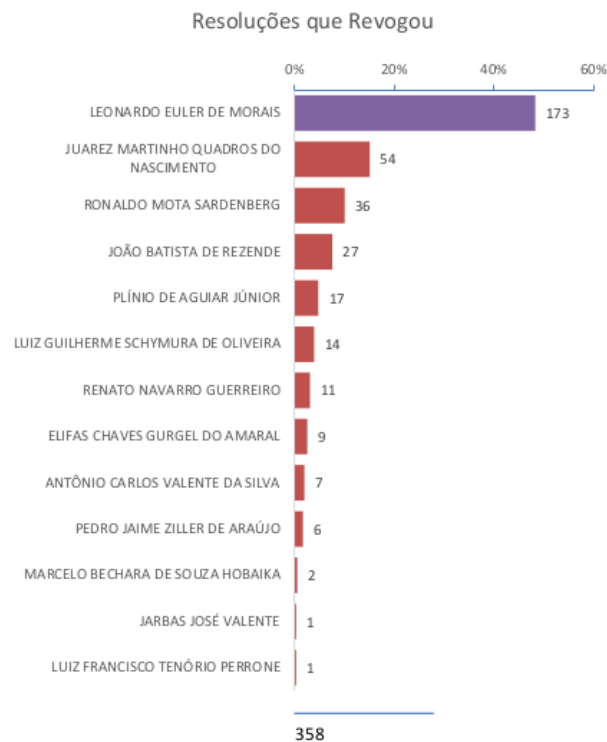


Figura 2 – Quantidade de Resoluções Revogadas por Signatário

Considerando a existência de interdependências entre Resoluções, pergunta-se: **é possível haver alguma similaridade entre as sentenças, parágrafos ou itens das Resoluções, sendo, portanto, candidatos a simplificação?** O gráfico 3 abaixo mostra a média de Itens por Resolução.

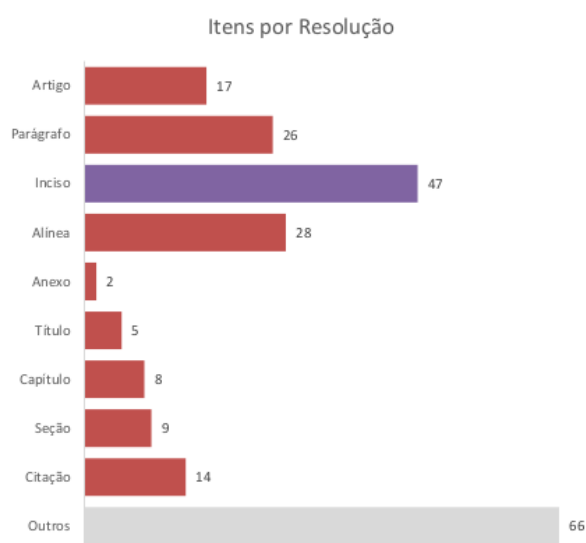


Figura 3 – Quantidade média de Itens por Resolução

Pela quantidade média de informações por Itens de Resolução fica evidente que a diminuição das sentenças de cada Item, por graus de similaridade pode ser uma maneira de eliminar e simplificar os conteúdos redundantes nos documentos

A partir desta análise, se define como situação-problema encontrar uma solução efetiva para tratar as similaridades existentes nas sentenças das Resoluções da Anatel.

A necessidade de encontrar e tratar a similaridade entre as sentenças dos Itens de uma Resolução, além das sentenças dos Itens entre as Resoluções, visando a redução e a simplificação dos textos torna-se, desse modo, objeto de estudo:

- Propor, na primeira etapa, um método que avalie o grau de similaridade entre as sentenças das Resoluções.

Desse modo, o **objetivo geral** desta proposta é definido como:

- A partir de tecnologias de Processamento da Linguagem Natural, fornecer informações que possam suportar a simplificação das Resoluções da Anatel, apresentando sentenças com graus de similaridade como suporte às análises dos gestores da Anatel.

Para atender as necessidades que o objetivo geral pressupõe, os seguintes **objetivos específicos** são definidos:

- Definir estruturas (asserções) que possibilitem recuperar sentenças regulatórias de interesse;
- Identificar o método para grau de similaridade;
- Recuperar sentenças de interesse por meio de aprendizado não-supervisionado;
- Validar o resultado junto aos gestores.

A seguinte **hipótese** de pesquisa é considerada:

- É possível que a análise de similaridade revele sentenças, parágrafos ou itens correspondentes em um regulamento ou entre regulamentos, sendo, portanto, candidatos a simplificação?

Para se atingir os objetivos traçados e provar a hipótese, foram definidas as seguintes **tarefas**:

- Revisar a literatura relacionada a NLP<sup>1</sup> e NLU<sup>2</sup> no intuito de apresentar, explicar e mostrar como tais tecnologias serão aplicadas;
- Analisar os artigos selecionados a fim de estruturar uma abordagem, arquitetura e solução aderente aos pressupostos da presente proposta;

---

1 NLP – Natural Language Process.

2 NLU - Natural Language Understanding

- Propor uma abordagem, arquitetura e solução com base nos conteúdos selecionados visando um Sistema de Recomendação<sup>3</sup>.

A **relevância** e a **justificativa** dessa proposta leva em consideração a premissa de que por meio da similaridade nas sentenças entre e intra Resoluções, é possível subsidiar a análise de redundâncias pela área competente nos discursos regulatórios, diminuindo a complexidade para melhor atender a sociedade tornando-se, assim, alinhada a normativa sobre desburocratização contida em [MINUTA, 2020].

No capítulo 2 é feita a análise dos artigos de interesse selecionados, tendo em vista a elaboração da presente proposta.

O capítulo 3 se refere as etapas de desenvolvimento da abordagem proposta.

Na conclusão é realizada uma análise geral onde serão relatados as dificuldades e obstáculos enfrentados, bem como a explicação de como os desafios serão solucionados.

---

3 Sistema de Recomendação – a) [https://pt.wikipedia.org/wiki/Aprendizado\\_de\\_máquina](https://pt.wikipedia.org/wiki/Aprendizado_de_máquina)

b) [https://pt.wikipedia.org/wiki/Aprendizado\\_de\\_máquina](https://pt.wikipedia.org/wiki/Aprendizado_de_máquina)

c) <https://medium.com/data-hackers/deep-learning-para-sistemas-de-recomendação-parte-1-introdução-b19a896c471e> Acessos em: 16/11/2020.

## CAPÍTULO 2 – RESOLUÇÕES E SIMILARIDADE

### 2.1 – Entendimento sobre as Resoluções

Uma Resolução<sup>4</sup> significa: “*Ato normativo com efeitos internos ao órgão que a criou ou a outros órgãos subordinados, podendo em raras hipóteses ter efeitos externos. É ato sempre subordinado à Lei e à Constituição Federal*”.

Para a Anatel<sup>5</sup> as Resoluções “*...regulam o setor de telecomunicações brasileiro, conforme competência estabelecida no art. 22 da LGT e no art. 35 do Regulamento da Agência. No Regimento Interno da Agência, foi estabelecida no art. 40, inciso I, e tem o procedimento de edição detalhado em seu art. 62 e seguintes*”.

As Resoluções da Anatel possuem em seus conteúdos no mínimo dois dos seguintes itens: artigo, parágrafo, inciso, alínea, anexo, título, capítulo, seção, citação e outros. Em cada item existem sentenças que dentro de um determinado contexto fazem sentido no discurso normativo.

A identificação de divergências entre as sentenças dos itens de uma Resolução, bem como nas sentenças entre os itens das Resoluções requer levar em consideração as discrepâncias a um patamar muito além da sinonímia para similaridade e antonímia para contradição [LI e QIN, 2020].

Em outras palavras, os discursos nas resoluções atravessam os saberes entre si enquanto que similaridade e contradição resultam, respectivamente, tanto da similitude como do contraste de homogeneidades antagônicas nos discursos.

Essa fronteira pode ser entendida como um lugar de conflito, onde direitos e deveres, normas e leis resultam em movências. Nesse sentido, **entende-se por movências como sendo aspectos de conflito entre “forças” em um discurso** [MITTMANN, 2010].

4 <https://www.dicionarioinformal.com.br/resolucao/> Acesso em: 13 out. 2020.

5 Resoluções da Anatel. <https://www.anatel.gov.br/legislacao/resolucoes> Acesso em: 13 out. 2020.



Em situações desse tipo [MITTMANN, 2010] explica que é possível pela sintaxe organizar a movência para um efeito mais estável, onde haja equilíbrio e mais homogeneidade de análise nos discursos e sentenças a fim de levar a um desfecho.

As estruturas sintáticas denominadas Sintagmas Nominais (SN) e Sintagmas Verbais (SV) permitem uma análise entre as sentenças dos discursos, pois são estruturas gramaticais que se relacionam através de uma ação entre sujeito e predicado [PINHEIRO, 2017].

Por meio da análise sintática os SN e SV nas sentenças podem ser usados, também, em um nível hierárquico de correspondências. Essas correspondências são baseadas na hiperonímia<sup>6</sup>, que significa que uma palavra é hierarquicamente superior a outra, enquanto o inverso é hiponímia. Exemplos de hiperônimo e hipônimo são:

Hiperônimo de	Hipônimo de
País	Brasil
Mamífero	Cão
Líquido	Água
Bebida	Café
Brasil	Brasília

Tabela 1 – Exemplos de hiperônimo e hipônimo

Agora, então, cabe a pergunta: como a análise sintática entre essas estruturas poderia ocorrer tendo em vista a averiguação de similaridade semântica?

Outra pergunta: como a análise sintática se compara em termos de esforço e resultados de análise de similaridade à algoritmos que podem abstrair entendimento sintático e inferir um modelo de linguagem a partir de aprendizado profundo, tais como BERT<sup>7</sup> e ULMFit<sup>8</sup>?

Com base em [MAYNARD-REID, 2020], [JOSHI, 2020] e [TANDOORI, 2020], é explicado que “*pre-training*” em modelos de linguagem são treinamentos feitos por meio de várias técnicas para fins de propósitos gerais em modelos de representação da

6 <https://www.normaculta.com.br/hiponimia-e-hiperonimia/> Acesso em: 14 out. 2020.

7 <https://neuralmind.ai/2020/01/26/neuralmind-disponibiliza-modelo-bert-inteligencia-artificial-do-google-em-portugues/>

8 <https://towardsdatascience.com/understanding-ulmfit-and-elmo-the-shift-towards-transfer-learning-in-nlp-b5d8e2e3f664>

linguagem natural, onde quantidades enormes de textos não etiquetados/categorizados são usadas neste intuito.

Quando surge a necessidade de ajustar esses modelos de linguagem de propósitos gerais a um conjunto de textos **menor** e/ou de uma **determinada área** ou **funcionalidade** específicas, técnicas de treinamento em modo ajuste fino, do inglês “*fine-tuning techniques*”, são usadas com objetivo de atender a especificidades pertinentes à área ou de uma funcionalidade, por exemplo, busca por similaridade semântica e Sistemas de Recomendação [MAYNARD-REID, 2020], [JOSHI, 2020] e [TANDOORI, 2020].

Com este objetivo em mente, os SN e SV são as estruturas sintáticas contidas na gramática da linguagem em diversos idiomas. O potencial de análise e conhecimento que essas estruturas possuem amplia a abrangência contextual e o sentido no qual se refere o discurso.

Por exemplo, a frase “*Aprova a destinação das faixas de rádio frequências*” possui os seguintes sintagmas:

- Sintagma verbal: Aprova (núcleo do predicado verbal)
- Sintagma nominal:
  - o a destinação (sintagma nominal simples nível 1, núcleo do sujeito, especificador nominal ou determinante, objeto direto)
  - o das faixas (sintagma nominal simples nível 1, especificador nominal ou determinante, objeto preposicional)
  - o de **rádio frequência** (no caso *frequência* qualifica *rádio*, portanto é um adjetivo que compõe o sintagma nominal e nesse caso é um sintagma nominal nível 2, objeto direto)
  - o a **destinação das faixas** (sintagma nominal composto nível 2)
  - o a destinação das faixas de rádio frequência (sintagma nominal composto nível 3)

Uma palavra sozinha não mostra o sentido ou contexto no qual está inserida. Diferentemente, nos exemplos acima os micros contextos existentes dão sentido a frase. Os SN e/ou SV no contexto desta proposta formará o dicionário de termos, também conhecido

como léxico<sup>9</sup>, e conforme [MAYNARD-REID, 2020], [JOSHI, 2020] e [TANDOORI, 2020], o treinamento em modo ajuste fino, sendo necessário, utilizará o dicionário de termos citado como *input* objetivando, por exemplo, aprimorar a busca por similaridade semântica e aumentar o *score* desta funcionalidade para um Sistema de Recomendação.

Cabe ressaltar que não existe a melhor abordagem semântica e sintática, haja vista que todas elas foram desenvolvidas para resolver determinadas situações-problema, seja qual for a área, e foram a solução para diversas aplicações, e algumas são evolução das antecessoras.

O fundamental é encontrar a abordagem que seja pertinente e o mais próximo possível das relações custo vs benefício vs tempo vs resultado vs qualidade.

---

9 [https://pt.wikipedia.org/wiki/Léxico\\_da\\_Língua\\_portuguesa](https://pt.wikipedia.org/wiki/Léxico_da_Língua_portuguesa) Acesso em: 26/10/2020

## 2.2 – Avaliação da implementação do relatório de Resoluções

Ao analisar o código `Resolucoes.py` foi observado a falta de comentários explicando o objetivo de cada bloco de código. Por exemplo em:

```
while True:
    # decrease the maxInt value by factor 10
    # as long as the OverflowError occurs.
    try:
        csv.field_size_limit(maxInt)
        break
    except OverflowError:
        maxInt = int(maxInt/10)
```

Há uma iteração e pelo trecho comentado é possível entender que ocorre um decremento, mas qual o objetivo? É de algum arquivo do tipo `csv`<sup>10</sup>? Qual(is)? Onde é usado?

Existem 32 variáveis que armazenam strings com padrões de expressões regulares. Pelo nome das variáveis é possível inferir o objetivo do regex. Sendo essas expressões utilizadas, exclusivamente, na extração de informação das Resoluções pode acontecer de surgir um ou mais tipos de estruturas textuais que não se encaixam nos padrões dos regex. Ainda, um regex tem uma sintaxe hostil para quem não o domina, tornando o trabalho de manutenção bem difícil para quem ainda não domina as expressões regulares.

O código foi dividido em blocos de funções, mas tem três funções que chamam atenção pelo tamanho e pela quantidade de estruturas de condição, a saber: `read_resolucao(this_id_resolucao)`, `get_resolucoes()` e `get_signatario_e_cargo(tag)`.

Compreender a funcionalidade dessas funções, somado a falta de comentários explicativos torna complexo e difícil o trabalho de entendimento e manutenção desse programa. Inclusive, ainda que existam funções menores elas, também carecem de comentários, o que torna a tarefa de compreensão ainda mais difícil.

A função `get_citacoes( )` é extremamente longa e nada fácil de entender. Possui estruturas de repetições e de controle aninhadas, mais uma série de chamadas de objetos com um conjunto de parâmetros enormes que se perdem do campo visual.

---

<sup>10</sup> <https://en.wikipedia.org/wiki/CSV>

Em síntese, é possível intuir que o programa Resolucoes.py atende a uma necessidade bem específica, a sua manutenção e entendimento são bem complexos e os padrões regex limitam a capacidade de identificar qualquer outro tipo de estrutura textual, ou seja, em se tratando de textos existe uma grande possibilidade de algum trecho pertinente a necessidade não ser capturado.

### **2.3 – Transfer Learning com Ajuste Fino no BERT**

O Aprendizado por Transferência, do inglês Transfer Learning - TL [RUDER et al, 2019], é uma técnica em que um modelo de aprendizado profundo (Deep Learning - DL) é treinado por meio de um grande conjunto de dados. A transferência de aprendizado acontece quando o modelo é usado para realizar tarefas semelhantes em outro conjunto de dados, cuja denominação conhecida é modelo pré-treinado [DEVLIN et al, 2019]. A concepção do modelo pré-treinado está relacionada ao ponto de partida que, nesse caso, começa a partir de um treinamento pré-existente, a exemplo o ImageNet , cujo modelo voltado para a visão computacional foi treinado utilizando imagens da web.

Em [SARKAR et al, 2018] é explicado que o aprendizado por transferência se tornou uma opção viável em NLP. A maioria das tarefas em NLP, tais como classificação de texto, modelagem de linguagem, tradução automática dentre outros, possuem por característica em comum a modelagem de sequência. Grande parte dos modelos de aprendizado de máquina e redes neurais possuem um viés quando é necessário analisar informações sequenciais em textos.

Nesse sentido é comum a utilização das Redes Neurais Recorrentes (RNN – Recurrent Neural Networks) como as LSTM (Long-Short Term Memory) porque suas arquiteturas permitem modelar informações sequenciais [HOCHREITER, 1997]. Por outro lado, as RNN possuíam o problema que impossibilitava a paralelização e isso gerava um grande impacto no tempo de treinamento requerido, acabando por inviabilizar o treinamento em um grande conjunto de dados [SARKAR et al, 2018] e [BAKKER, 2018].

Em 2017, foi publicado o artigo “*Attention is All You Need*” onde a Google introduziu o conceito Transformer que marcou o avanço em análise de sequências em NLP [VASWANI et al, 2017], onde a limitação no treinamento das RNN foi elegantemente

resolvida viabilizando o treinamento, por exemplo de uma LSTM [HOCHREITER, 1997], pegando toda a sequência de dados como entrada e possibilitando, inclusive, que o processamento fosse feito em uma GPU<sup>11</sup>.

Outro avanço muito substancial vem do fato de pré-treinar o modelo sem a necessidade de rotular os dados, ou seja, é necessário fornecer apenas uma grande quantidade de dados textuais não rotulados para treinar um modelo baseado em Transformer [VASWANI et al, 2017], possibilitando a utilização de um modelo treinado em tarefas, por exemplo, de classificação de texto, reconhecimento de entidade nomeada (NER<sup>12</sup>), geração de texto e etc.

No contexto da presente proposta, a similaridade será viabilizada por intermédio dessa tecnologia. Nesse intuito, o emprego do modelo de linguagem BERT - Bidirectional Encoder Representation from Transformers [DEVLIN et al, 2019] será aplicado, haja vista, como ponto de partida, o grande número de parâmetros, que variam entre 100 a 300 milhões, herdados a partir desse modelo pré-treinado.

Jacob Devlin et al [DEVLIN et al, 2019] projetaram essa arquitetura no intuito de possibilitar o retreinamento bidirecional de maneira profunda a partir de textos não rotulados.

BERT possui vários codificadores Transformer empilhados (apenas a parte do codificador) e a bidirecionalidade no treinamento é o diferencial porque em sua camada Self-Attention [VASWANI et al, 2017] é executada da esquerda para direita e vice-versa. Dessa maneira a arquitetura BERT faz com que, por exemplo, uma palavra tenha relação com todas as demais da frase/sentença e, por meio de *Attention*, faz com que a sentença obtenha relação com outras sentenças [CONNEAU, 2020].

O ajuste fino no BERT-multilingue seria utilizar o modelo pré-treinado como ponto de partida e, sendo o caso, incorporar novas sentenças por meio de um retreinamento [RODRIGUES et al, 2020] e [DEVLIN et al, 2019]. Por exemplo, em [SOUZA, 2020] o corpus do português brasileiro denominado BrWaC - Brazilian Web as Corpus [FILHO, et al, 2018] foi utilizado no retreinamento do BERT-multilingue em modo fine-tuning e

---

11 <https://pt.wikipedia.org/wiki/CUDA>

12 [https://en.wikipedia.org/wiki/Named-entity\\_recognition](https://en.wikipedia.org/wiki/Named-entity_recognition)

avaliado nas tarefas STS – Semantic Textual Similarity usando correlação de Pearson, RTE – Recognizing Textual Entailment e o NER – Named Entity Recognition usando F1-score. Esse trabalho deu origem ao BERTimbau<sup>13</sup>.

O exemplo acima demonstra a flexibilidade da arquitetura BERT de ser ajustado visando tarefas ou áreas específicas. Sendo o caso, aplicar o ajuste fino na presente proposta aumentará a acurácia na busca por similaridade semântica.

---

<sup>13</sup> <https://github.com/neuralmind-ai/portuguese-bert>

## 2.4 – Sentence-BERT – Sentence Embeddings using Siamese BERT-Networks

Sentence-BERT é uma variante do framework BERT que implementa em sua estrutura as Redes Neurais Siamesas<sup>14</sup>.

Em relação ao Sentence-BERT (SBERT) os autores Nils Reimers e Iryna Gurevych tiveram duas grandes ideias: primeiro, desenvolveram uma abordagem que diminuiu absurdamente o tempo e o esforço quando, por exemplo, em busca de um par que se assemelhe semanticamente a sentença de entrada. Segundo os autores [REIMERS e GUREVYCH, 2019], o desempenho da SBERT ademais da rapidez a precisão se equipara ao do BERT de [DEVLIN et al, 2019].

Os autores fizeram experimentos comparando o SBERT e o SROBERTa com vários métodos nas tarefas de Similaridade Semântica Textual (STS – Semantic Textual Similarity) e na de Aprendizado por Transferência. Os resultados comprovam que os demais métodos foram superados na tarefa STS, o mesmo ocorrendo em tarefas de aprendizado não supervisionado onde o BERT e o RoBERTa<sup>15</sup> também foram superados.

A utilização do SBERT no contexto desta proposta será a considerada posto que esse modelo de linguagem, como consta em [REIMERS e GUREVYCH, 2019], apresentou melhores resultados em um tempo de retreinamento muito menor. Este modelo de linguagem contribui para um possível ajuste fino, em caso de considerar esta possibilidade o recurso computacional pode ser o de uma GPU padrão NVIDIA GTX ou superior, ademais da CPU<sup>16</sup>.

O SBERT possui, dentre outras, uma função de busca<sup>17</sup> semântica cujo objetivo é melhorar a precisão de uma pesquisa e aumentar a compreensão sobre um determinado conteúdo. Nesse caso é calculado o grau de semelhança semântica de um texto de entrada em relação a outros textos tendo como resultado um *score* de semelhança para cada texto retornado.

---

14 [https://en.wikipedia.org/wiki/Siamese\\_neural\\_network](https://en.wikipedia.org/wiki/Siamese_neural_network)

15 <https://arxiv.org/pdf/1907.11692.pdf>

16 [https://en.wikipedia.org/wiki/Central\\_processing\\_unit](https://en.wikipedia.org/wiki/Central_processing_unit)

17 [https://www.sbert.net/docs/usage/semantic\\_search.html](https://www.sbert.net/docs/usage/semantic_search.html)



## CAPÍTULO 3 - ABORDAGEM PROPOSTA

### 3.1- Proposta

Desenvolver um Sistema de Recomendação que, dado uma Resolução informada, retorna  $n$  Resoluções ordenadas da maior para a de menor grau de semelhança utilizando, no caso, a função de busca semântica do modelo de linguagem SBERT.

A tarefa #10 corresponde a subseção 3.1.1 e a tarefa #9 a subseção 3.1.2, as quais serão descritas logo abaixo. Em ambas tarefas se estima uma Sprint para cada.

#### 3.1.1 – Descoberta de tópicos

Tem por objetivo a recuperação de informação das estruturas sintáticas Sintagmas Nominais (SN) e Sintagmas Verbais (SV), cujo resultado será o dicionário de termos (léxico). O KeyBERT<sup>18</sup> possui em sua API duas funções de semelhança por cosseno: a) Max Sum Similarity (MSS) com uma diversificação de termos regular e b) Maximal Marginal Relevance (MMR) que permite alta diversificação de termos nos resultados.

Por se tratar de uma tarefa de suma importância, como explicado nos capítulos anteriores, é recomendada para o seu desenvolvimento uma sprint. Várias iterações serão feitas no intuito de selecionar um conjunto ótimo de termos para o dicionário.

Nesta fase o KeyBERT terá como entrada o corpus de Resoluções e como saída o dicionário de termos (léxico) que, por sua vez, será entrada no modelo de linguagem SBERT, o qual será descrito a seguir.

---

<sup>18</sup> <https://towardsdatascience.com/keyword-extraction-with-bert-724efca412ea> e <https://github.com/MaartenGr/KeyBERT> Acesso em: 17/11/2020.

[2] [https://en.wikipedia.org/wiki/Word\\_embedding](https://en.wikipedia.org/wiki/Word_embedding)

### 3.1.2 – Busca por similaridade semântica

Corresponde a Tarefa #9 Implementar busca por similaridade com SBERT. Nessa etapa serão usados como subsídios o dicionário de termos criado na etapa 3.1.1 e todo o conhecimento aprendido a partir destas tarefas para desenvolver o Sistema de Recomendação. O SBERT possui *score* e essa métrica informa o grau de similaridade entre os textos de interesse.

Tendo recebido como entrada o dicionário de termos da fase anterior, o SBERT estará preparado para melhor avaliar o grau de similaridade semântica, por exemplo entre uma nova Resolução e as demais presentes no corpus simulando, neste sentido, um Sistema de Recomendação.

Funcionalmente, seria receber a nova Resolução como entrada e ter como resultado um conjunto *top n* de Resoluções símile por ordem decrescente.

## CONCLUSÃO

A proposta aqui desenvolvida é o início de várias etapas de análises a serem feitas sobre as Resoluções a fim de atingir o objetivo de ser um instrumento que auxilia os gestores da SPR em suas atividades, dentre elas a eliminação de redundâncias e a simplificação.

As abordagens e métodos explicados e selecionados neste trabalho formam o conjunto de tecnologias visando o desenvolvimento de um Sistema de Recomendação, onde diversas situações-problema que se apresentam em um trabalho de análise do discurso [MANNING, 2009] e [JURAFSKY e MARTIN, 2019] são tratados.

O protótipo desenvolvido mostrou a viabilidade de usar o SBERT na busca por similaridade semântica. As tarefas previstas para o desenvolvimento desta proposta estão embasadas em técnicas de Linguística Computacional, Processamento de Linguagem Natural, Compreensão da Linguagem Natural, Estatística, Machine Learning e Deep Learning, dando base e garantindo o desenvolvimento do Sistema de Recomendação de Similaridade sobre as Resoluções para a Anatel.

## REFERÊNCIAS BIBLIOGRÁFICAS

- [ANATEL, 2019] ANATEL. Relatório Técnico – Análise sobre as Resoluções da Anatel. Brasília, junho de 2019.
- [PORTAL, 2020] PORTAL, da Anatel.  
<https://www.anatel.gov.br/Portal/exibirPortalPaginaEspecial.do?acao=&codItemCanal=801&nomeVisao=Conhe%20a%20Anatel&nomeCanal=Sobre%20a%20Anatel&nomeItemCanal=Miss%20o%20atribui%20E7%F5es%20e%20caracter%EDsticas#:~:text=A%20miss%C3%A3o%20da%20Anatel%20%C3%A9,em%20todo%20o%20territ%C3%B3rio%20nacional>. Acesso em: 06 out. 2020.
- [MINUTA, 2020] MINUTA. Proposta normativa sobre desburocratização.  
<https://www.novo.justica.gov.br/seus-direitos-2/elaboracao-legislativa/debate-sobre-desburocratizacao/desburocratizacao>  
 Acesso em: 06 out. 2020.
- [MITTMANN 2010] MITTMANN, Solange. Heterogeneidade constitutiva, contradição histórica e sintaxe. Revista do Programa de Pós-Graduação em Letras da Universidade de Passo Fundo, v.6 – n.1 – p. 85-101, jun. 2010.
- [JURAFSKY e MARTIN, 2019] JURAFSKY, Daniel, MARTIN, James H. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Third Edition draft, Stanford University, oct. 2019
- [CONNEAU, 2020] CONNEAU, Alexis, et al. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. p. 670 – 680, September, 2017. Association for Computational Linguistics.  
<https://www.aclweb.org/anthology/D17-1070> Acesso em: 08 out. 2020.
- [LI e QIN, 2020] LI, Luyang, QIN, Bing, LIU, Ting. Contradiction Detection with Contradiction-Specific Word Embedding. May 24, 2017. Algorithms — Open Access Journal.  
<https://www.mdpi.com/journal/algorithms> Acesso em: 07 out. 2020.
- [PINHEIRO, 2017] PINHEIRO, Marcello S., PRADO, Hércules A. do, FERNEDA, Edilson, LADEIRA, Marcelo. An Approach for Text Mining Based on Noun Phrases. International Conference on Intelligent Decision Technologies – IDT 2015. pp 525-535. Smart Innovation, Systems and Technologies, Spring Book Series, 2017.
- [SILVA, 2010] SILVA, Bento Carlos Dias da. Brazilian Portuguese WordNet: A computational linguistic exercise of encoding bilingual relational lexicons. International Journal of Computational Linguistics and Applications, v. 1, n. 1, p. 137-150, 2010. Disponível em: <http://hdl.handle.net/11449/124798>
- [REIMERS, 2019] REIMERS, Nils, GUREVYCH, Iryna. Sentence-BERT:

- Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 3982–3992, Hong Kong, China, Association for Computational Linguistics, nov. 2019.
- [CHAMI, 2020] CHAMI, Ines, GU, Albert, CHATZIAFRATIS, Vaggos, RÉ, Christopher. From Trees to Continuous Embeddings and Back: Hyperbolic Hierarchical Clustering. arXiv:2010.00402. 01 out. 2020.
- [SOUZA, 2020] SOUZA, Fábio, NOGUEIRA, Rodrigo, LOTUFO, Roberto. Portuguese Named Entity Recognition using BERT-CRF. <https://arxiv.org/abs/1909.10649> Acesso em: 15/10/2020.
- [RODRIGUES, 2020] RODRIGUES, Ruan C, SILVA, Jéssica R., CASTRO, Pedro, FELIX, Nadia. Multilingual Transformer Ensembles for Portuguese Natural Language Tasks. Conference: ASSIN 2 Shared Task: Evaluating Semantic Textual Similarity and Textual Entailment in Portuguese. At: <http://ceur-ws.org/Vol-2583/Volume: Vol-2583>, March 2020
- [FILHO, 2018] FILHO, Jorge A. W., WILKENS, Rodrigo, IDIART, Marco, VILLAVICENCIO, Aline. The brWaC Corpus: A New Open Resource for Brazilian Portuguese. In 11th Language Resources and Evaluation Conference (LREC), May 2018.
- [HOCHREITER, 2018] HOCHREITER, Sepp, SCHIMIDHUBER, Jürgen. Long Short-Term Memory. *Neural Computation* 9(8):1735-1780, 1997  
[https://www.researchgate.net/publication/13853244\\_Long\\_Short-term\\_Memory](https://www.researchgate.net/publication/13853244_Long_Short-term_Memory) Acesso em: 14/10/2020.
- [RUDER et al, 2019] RUDER, Sebastian, PETERS, Matthew E., SWAYAMDIPTA, Swabha, WOLF, Thomas. Transfer Learning in Natural Language Processing. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics. NAACL, pg 15 – 18. Association for Computational Linguistics, jun. 2019.
- [VASWANI, 2017] VASWANI, Ashish, et al. Attention Is All You Need. 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, Dec. 2017.
- [DEVLIN et al, 2019] DEVLIN, Jacob, CHANG, Ming-Wei, LEE, Kenton, TOUTANOVA, Kristina. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics. NAACL, pg. 4171–4186, vol. 1, May 2019. <https://arxiv.org/pdf/1810.04805.pdf> Acesso em: 14/10/2020.
- [JANSEN, 2018] JANSEN, Stefan. Mastering Unsupervised Learning with Python. Packt Publishing, Aug. 2018.
- [BONACCORSO, 2019] BONACCORSO, Giuseppe. Hands-On Unsupervised Learning with Python. Packt Publishing, Feb. 2019.
- [JOHNSTON, 2019] JOHNSTON, Benjamin. JONES, Aaron, KRUGER, Christopher. Applied Unsupervised Learning with Python. Packt Publishing, May 2019.
- [MANNING, 2008] MANNING, Christopher D., RAGHAVAN, Prabhakar,

- SCHÜTZE, Hinrich, Introduction to Information Retrieval, Cambridge University Press, 2008.
- [MAYNARD-REID, 2020] MAYNARD-REID, Margaret. Hands-On Transfer Learning with TensorFlow. Packt, May 2020.
- [HOWARD e RUDER, 2018] HOWARD, Jeremy, RUDER, Sebastian. Universal Language Model Fine-tuning for Text Classification. <https://arxiv.org/pdf/1801.06146.pdf> Acesso em: 23/10/2020.
- [WU, Yonghui et al, 2016] WU, Yonghui et al. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. <https://arxiv.org/pdf/1609.08144v2.pdf> Acesso em: 28/10/2020.
- [GROOTENDORST VANSCHOREN, 2020] GROOTENDORST, Maarten. VANSCHOREN, Joaquin. Beyond Bag-of-Concepts: Vectors of Locally Aggregated Concepts. Joint European Conference on Machine Learning and Knowledge Discovery in Databases – ECML PKDD 2019. Pp 681-696, apr., 2020.
- [JOSHI, 2020] JOSHI, Prateek. Transfer Learning for NLP: Fine-Tuning BERT for Text Classification. <https://www.analyticsvidhya.com/blog/2020/07/transfer-learning-for-nlp-fine-tuning-bert-for-text-classification/#:~:text=for%20Spam%20Classification-,Transfer%20Learning%20in%20NLP,model%20a%20pre%20trained%20model>. Acesso em: 16/11/2020.
- [TANDOORI, 2020] TANDOORI, Chandrashekhar. Simple Text Multi Classification Task Using Keras BERT. [https://www.analyticsvidhya.com/blog/2020/10/simple-text-multi-classification-task-using-keras-bert/?utm\\_source=linkedin&utm\\_medium=KJ|link|high-performance-blog|blogs|44146|0.375](https://www.analyticsvidhya.com/blog/2020/10/simple-text-multi-classification-task-using-keras-bert/?utm_source=linkedin&utm_medium=KJ|link|high-performance-blog|blogs|44146|0.375) Acesso em: 16/11/2020.