UNIVERSIDADE DE SÃO PAULO

Escola de Artes, Ciências e Humanidades

Matheus Mendes de Sant'Ana

Experimentos de classificação de documentos para caracterização autoral

São Paulo

Novembro de 2017

Universidade de São Paulo

Escola de Artes, Ciências e Humanidades

Matheus Mendes de Sant'Ana

Experimentos de classificação de documentos para caracterização autoral

Monografia apresentada à Escola de Artes, Ciências e Humanidades, da Universidade de São Paulo, como parte dos requisitos exigidos na disciplina ACH 2017 – Projeto Supervisionado ou de Graduação II, para obtenção do título de Bacharel em Sistemas de Informação.

Área de Concentração: Processamento de língua natural.

Orientador: Prof. Dr. Ivandré Paraboni

Modalidade: TCC Curto (1 semestre) – individual

São Paulo

Novembro de 2017

Agradecimentos

A Deus em primeiro lugar, pela coragem, ânimo e determinação que tem me dado ao longo de toda essa jornada.

Ao meu orientador Ivandré pela amizade, por toda a ajuda e paciência que teve comigo durante o período que estive neste curso, servindo como fonte de inspiração e exemplo para mim.

Ao aluno Georges pela amizade, pelo apoio técnico e pelos bons conselhos durante todo o tempo que realizei este trabalho.

A esta universidade, principalmente ao seu corpo docente, por viabilizar esta oportunidade de crescer em conhecimento e poder traçar esse caminho tão grandioso que me acompanhará durante toda minha vida.

A todos aqueles que direta ou indiretamente fizeram parte da minha formação, sem os quais eu não teria chegado até aqui, muito obrigado!

Dedicatória

A Deus, minha esperança e meu alicerce, sou grato de todo meu coração pelo conhecimento que me concedeu, através do qual pude alcançar este tão sonhado momento em minha vida.

À memória de meu querido pai Ricardo, por todos os momentos que passamos juntos, pelas sábias palavras que me sustentaram durante tempos tão difíceis e por todos os conselhos que me seguirão até o fim.

À minha querida mãe Regiane, minha fonte de alegria, sem a qual jamais poderia pensar em chegar tão longe, agradeço a você por acreditar em mim!

À minha irmã Isabelle, não tenho palavras para te agradecer por toda a bondade, sinceridade e paciência que teve comigo no decorrer de todo esse tempo, um abraço do seu querido irmão!

Glossário

AM: Aprendizado de máquina – subárea da Inteligência Artificial baseada em algoritmos que utilizam conhecimento pré-concedido (supervisionados) ou conhecimento experimental proveniente de tentativas (por reforço) para prever dados futuros e/ou desconhecidos.

PLN: Processamento de Língua Natural – subárea da Inteligência Artificial que estuda padrões humanos de linguagem e comportamento, de modo que esses padrões possam ser expressos e/ou interpretados através de uma máquina.

Stopwords: Palavras de pouca relevância para a interpretação do texto, por exemplo preposições e artigos.

Token: Divisão do texto em uma sequência de caracteres, simbolizando uma palavra no mundo real.

Outlier: Observação pontual e discrepante em relação às demais observações de uma amostra.

Resumo

A classificação de um documento é a tarefa de determinar o grupo ao qual o documento pertence, através de características comuns entre esse documento e um determinado grupo. Um tipo de classificação de documentos de especial interesse para este trabalho é o caso da caracterização autoral, que consiste em classificar documentos com base em características de seus autores, como gênero, idade, entre outras.

Este trabalho trata da tarefa de caracterização autoral baseada em textos em português, inglês e espanhol por meio de técnicas de classificação de documentos. Para cada tarefa de caracterização, será feita a análise comparativa de desempenho entre os modelos de classificação escolhidos.

Os resultados poderão auxiliar na escolha de um melhor modelo de classificação para a execução de tarefas similares, e servirão de referência para futuras pesquisas na área (*e.g.*, baseadas em técnicas mais sofisticadas).

Palavras chaves: Caracterização autoral, Classificação de documentos, Processamento de Língua Natural.

Abstract

Document classification is a task of determining the group to which a document belongs, through common features between that document and a particular group. A kind of document classification of special interest for this work is the case of author profiling, which consists of classifying documents based on characteristics of their authors, such as gender, age, and others.

This work addresses the task of author profiling based on texts in Portuguese, English and Spanish through document classification techniques. For each profiling task, a comparative performance analysis will be carried out between selected classification models.

The results may help determining a suitable classification model for performing similar tasks, and will serve as a reference for future research in the field (*e.g.*, based on more sophisticated techniques).

Keywords: Author profiling, Document classification, Natural Language Processing.

Lista de Figuras

Figura 1. Parte da Figura 2 do artigo de Kusner et al. (2015, p. 3). Compatibilidade entre palavras de documentos de comprimento igual.....11

Figura 2. Parte da Figura 2 do artigo de Kusner et al. (2015, p. 3). Compatibilidade entre palavras de documentos de comprimento diferente.....11

Lista de Tabelas

Tabela 1. Informações complementares da base de dados em português.....17

Tabela 2. Informações complementares da base de dados em inglês.....17

Tabela 3. Informações complementares da base de dados em espanhol.....17

Tabela 4. Precisão, revocação e medida F relacionadas à tarefa *age* da base de dados em português.....19

Tabela 5. Precisão, revocação e medida F relacionadas à tarefa *course* da base de dados em português.....20

Tabela 6. Precisão, revocação e medida F relacionadas à tarefa *gender* da base de dados em português.....20

Tabela 7. Precisão, revocação e medida F relacionadas à tarefa *relig* da base de dados em português.....21

Tabela 8. Precisão, revocação e medida F relacionadas à tarefa *ti* da base de dados em português.....21

Tabela 9. Precisão, revocação e medida F relacionadas à tarefa *ext* da base de dados em português.....21

Tabela 10. Precisão, revocação e medida F relacionadas à tarefa *cons* da base de dados em português.....21

Tabela 11. Precisão, revocação e medida F relacionadas à tarefa *agr* da base de dados em português.....22

Tabela 12. Precisão, revocação e medida F relacionadas à tarefa *neu* da base de dados em português.....22

Tabela 13. Precisão, revocação e medida F relacionadas à tarefa *ope* da base de dados em português.....22

Tabela 14. Precisão, revocação e medida F relacionadas à tarefa *age* da base de dados em inglês.....23

Tabela 15. Precisão, revocação e medida F relacionadas à tarefa *gender* da base de dados em inglês.....23

Tabela 16. Precisão, revocação e medida F relacionadas à tarefa *age* da base de dados em espanhol.....23

Tabela 17. Precisão, revocação e medida F relacionadas à tarefa *gender* da base de dados em espanhol.....24

Sumário

1 Introdução....1

2 Objetivos....2

2.1 Objetivo Geral....2

2.2 Objetivos Específicos....2

3 Estudo Teórico....3

3.1 Visão Geral....3

3.2 Descrição dos Métodos de Classificação....3

4 Metodologia....12

4.1 Tarefas de Caracterização Autoral....12

4.2 Modelos de Classificação....13

5 Avaliação....16

5.1 Conjuntos de Dados....16

5.2 Procedimento....18

5.3 Resultados....18

5.4 Discussão....24

5.4.1 Análise dos resultados por idioma....24

5.4.2 Análise dos resultados por tarefa....25

5.4.3 Análise geral dos resultados....26

6 Conclusão....27

Referências Bibliográficas....28

APÊNDICE A – Repositório do código em *Python*....32

APÊNDICE B – Repositórios dos dados de saída....33

ANEXO A – Códigos em *Python* utilizados como base....34

1 Introdução

A classificação de documentos pode ser definida como a rotulação de cada documento de teste em um grupo com características semelhantes a ele, levando em consideração atributos que permitam distingui-lo de forma eficiente entre os demais grupos (*e.g.*, distinção por sintaxe ou por semântica). Um tipo de classificação de documentos de especial interesse para este trabalho é o caso da caracterização autoral, que pode ser definida como uma tarefa mais específica de classificação. A caracterização autoral consiste em classificar documentos com base em características de seus autores, como gênero, idade, entre outras.

O foco do presente trabalho é o problema computacional da caracterização autoral a partir de documentos textuais. Compararemos diversos modelos de classificação, entre eles *Bag of Words* (MCCALLUM et al., 1998) e *Doc2Vec* (LE e MIKOLOV, 2014), através da execução de uma série de tarefas de caracterização autoral.

Para as tarefas de caracterização, utilizaremos três conjuntos de dados. O primeiro conjunto (RAMOS et al., 2017; SILVA e PARABONI, 2017) é constituído de dados em português: dados extraídos de usuários do *Facebook* (voluntários) – encontra-se codificado em formato numérico, no qual as palavras foram substituídas por índices devido à necessidade de manter a privacidade dos usuários que cederam os dados. Já o segundo conjunto de dados (RANGEL et al., 2013) está em inglês e o terceiro conjunto de dados (RANGEL et al., 2013) está em espanhol.

Os três conjuntos de dados textuais são rotulados com informações autorais diversas. Em relação ao primeiro conjunto de dados, as tarefas de caracterização autoral são referentes às dez seguintes propriedades (classes): idade, curso, gênero, religião, atuação na área de TI, extroversão, conscienciosidade, agradabilidade, neuroticismo e abertura à experiência. Sobre o segundo e o terceiro conjuntos de dados, as tarefas são referentes à idade e ao gênero.

2 Objetivos

2.1 Objetivo Geral

O objetivo deste trabalho foi desenvolver e avaliar modelos de classificação de documentos para tarefas de caracterização autoral, utilizando técnicas tradicionais de aprendizagem de máquina, de modo a produzir resultados de referência para futuros estudos nesta área.

2.2 Objetivos Específicos

Os objetivos específicos deste trabalho foram:

- Treinar modelos de classificação de documentos, baseados em uma série de algoritmos de AM (aprendizado de máquina) de interesse para a área.

- Testar os modelos produzidos, com base em córpus em português, inglês e espanhol – conforme exigido para cada tarefa de caracterização autoral.

3 Estudo Teórico

3.1 Visão Geral

Nesta seção serão descritos brevemente os 7 métodos de classificação de documentos a serem utilizados neste trabalho.

3.2 Descrição dos Métodos de Classificação

Word2Vec

Este método é considerado um grande avanço recente na área de PLN, representando *tokens* (pedaços da divisão do texto que simbolizam "palavras" no mundo real) em vetores de distâncias em relação aos *tokens* com semântica similar (MIKOLOV et al., 2013a, 2013b). Embora o método propriamente dito não tenha sido implementado nem utilizado diretamente nas tarefas de caracterização autoral deste relatório, será denotada a sua teoria devido à sua contextualização indireta na implementação de alguns dos métodos aqui citados.

Motivacionalmente, a maior parte da complexidade do modelo de redes neurais por alimentação direta (*feedforward*) é descrita por Mikolov et al. (2013b) como sendo causada pela camada escondida não linear no modelo. Diferentemente desse modelo anterior, os modelos idealizados – relacionados ao método *Word2Vec* – podem não ser capazes de representar os dados tão precisamente como redes neurais, mas podem ser treinados em muito mais dados eficientemente.

Recentemente, Mikolov et al. (2013b) introduziram o modelo *Skip-gram* ao *Word2Vec*: trata-se de um método eficiente para aprendizado, a partir de representações de alta qualidade de palavras em vetores. Diferentemente da arquitetura da maioria das redes neurais utilizadas anteriormente – baseadas em aprendizagem por meio de vetores de palavras – o treino do modelo *Skip-gram* não envolve multiplicações de matrizes densas, o que torna o treinamento muito mais eficiente.

Segundo Goldberg e Levy (2014, p. 5) o método, através do modelo *Skip-gram*, assume a hipótese de que "palavras em contextos similares possuem significados similares". Simplificadamente, um contexto é um conjunto de *n* palavras adjacentes no documento em questão, posto que *n* é um número arbitrário escolhido a princípio.

Após a divisão do documento em contextos de tamanho *n*, são gerados os *tokens* do documento: estes por sua vez podem ser tanto frases como palavras únicas. A criação de *tokens* é feita mais especificamente a partir da função *softmax*, a qual provê o aumento de similaridade semântica entre os termos que compõem o *token*. Vale frisar que a hipótese de Goldberg e Levy (2014) sustenta-se na similaridade semântica calculada pela função *softmax*.

Em se tratando das tarefas de classificação, o objetivo substancial do modelo *Skip-gram* é melhor dividir as frases de maneira que as representações das palavras sejam mais úteis para prever as palavras ao redor, seja em uma frase ou documento (MIKOLOV et al., 2013a).

Como constatam Mikolov et al. (2013a), representações por palavras são limitadas por sua incapacidade de representar frases idiomáticas que não são composições de palavras individuais. Por exemplo, "Globo de Boston" é um artigo, e não uma combinação natural dos significados de "Globo" e "Boston". Por outro lado, usando vetores para representar todas as frases torna o modelo *Skip-gram* consideravelmente mais expressivo.

Três exemplos de representações ideais do modelo baseado em frases são:

- vetor("Montreal Canadiens") - vetor("Montreal") + vetor("Toronto") = vetor("Toronto Maple Leafs")
- vetor("Rússia") + vetor("rio") é próximo a vetor("rio Volga")
- vetor("Alemanha") + vetor("capital") é próximo a vetor("Berlim")

O modelo *Skip-gram* tem como coadjuvante outro modelo chamado NCE (*Noise Contrastive Estimation*), o qual postula que um bom modelo deve ser capaz de diferenciar dados de ruído por meio de regressão logística. Enquanto o NCE pode maximizar a probabilidade de *log* do *softmax* pela remoção de ruídos, o *Skip-gram* está apenas preocupado com o aprendizado de representações vetoriais de alta qualidade. Dito isso, estamos livres para simplificar o NCE ao passo que as representações vetoriais mantenham sua qualidade: ao juntarmos o modelo *Skip-gram* ao NCE de forma coerente, podemos obter tempo de

treinamento (relacionado fortemente ao NCE) e precisão (relacionada fortemente ao *Skip-gram*) razoáveis.

Averaging Word Vectors

Neste método, primeiro carregamos um modelo *Word2Vec* (MIKOLOV et al., 2013a, 2013b) pré-treinado com base em uma grande massa de documentos, obtemos então *tokens* com semântica altamente coesa. Aplicamos ao *Word2Vec* um mecanismo diferente para separar *tokens* e preservamos as palavras do vocabulário assim como se encontram no texto, podendo estar em caixa alta ou baixa.

Bag of Words

O método *Bag of* Words se baseia em uma característica singela de um documento de texto: a contagem da ocorrência de cada palavra neste documento. O modelo é uma representação simplificadora do texto, usada no processamento de língua natural e recuperação de informação. Neste modelo, um texto (como uma frase ou um documento) é representado como o conjunto de suas palavras, desconsiderando a sintaxe, a semântica e a ordem delas (MCCALLUM et al., 1998).

Após obtidas as principais palavras distintas de um córpus, cada texto que se pretende classificar é mapeado em um vetor, o qual conterá uma posição correspondente a cada uma das principais palavras desse córpus. Cada uma dessas posições conterá o número da quantidade de vezes que a palavra aparece no documento, ou sua proporção em relação às demais palavras do documento.

N-gramas de Caracteres

O método tem como principal recurso um divisor “pobre” de *tokens*, mas que funciona bem em alguns casos: em vez de utilizar quebra de linha, tabulações, espaços ou outros separadores para dividir e extrair as palavras do texto (como ocorre no método *Bag of Words* por exemplo), esse *tokenizer* extrai sequências – chamadas propriamente de n-gramas de

caracteres ou *tokens* – através da divisão do texto em sequências de caracteres de tamanho semelhante (FUSILIER et al., 2015).

Um n-grama de caractere (em inglês "*character n-gram*") é uma sequência de caracteres de comprimento *n*, extraída de um determinado documento: se temos em um documento o texto "abcdef", exemplos de n-gramas de caractere desse texto, utilizando comprimento 3, seriam os conjuntos de caracteres "abc", "bcd", "cde" e "def" (FUSILIER et al., 2015). O parâmetro *n* pode variar dependendo da linguagem e do domínio de interesse.

Deep IR

Trata-se de uma técnica desenvolvida por Taddy (2015), quem contribuiu com o tutorial *Gensim* (REHUREK e SOJKA, 2011). Resumidamente, o método *Deep IR* assume os seguintes passos:

1. Treine *n* modelos *Word2Vec*, posto que *n* corresponde ao número de classes do documento. Cada modelo será treinado com dados de somente uma das classes possíveis do documento.

2. Verifique o modelo que classifica melhor os dados usando o Teorema de Bayes (VAPNIK, 1998).

3. Utilize o melhor modelo do passo 2 para executar as tarefas de classificação.

Segundo Taddy (2015), este método utiliza a abordagem de mapeamento das palavras em um espaço vetorial, chamada de representação de linguagem distribuída – acompanhando a mesma lógica do método *Word2Vec* (MIKOLOV et al., 2013a, 2013b). Para padronização do texto, codifica-se todas as palavras presentes no vocabulário em uma árvore binária de Huffman (KNUTH, 1985).

O algoritmo é executado como segue. Inicialmente, calcula-se a probabilidade de cada documento em cada classe, que será igual à média das probabilidades de cada frase desse documento ser encontrada na classe em questão – equação (6), proposta por Taddy (2015, p. 3). Conclusivamente, dentre todos os documentos do córpus, verifica-se qual classe está associada à maior probabilidade de documento: a classe resultante será considerada como o

melhor modelo de treino para a tarefa de classificação – equação (7), proposta por Taddy (2015, p. 3).

O uso deste método oferece muitas vantagens (TADDY, 2015):

- **Simplicidade**: A estratégia de inversão funciona para qualquer modelo de linguagem que pode (ou seu treinamento pode) ser interpretado como um modelo probabilístico. A estratégia é também interpretável: qualquer intuição que se tenha sobre o modelo de linguagem distribuída pode ser aplicada diretamente à regra de classificação baseada em inversão.

- **Escalabilidade**: Quando estamos trabalhando com córpus massivos, frequentemente é útil dividir os dados em blocos como parte de estratégias de computação distribuída. Este modelo de classificação por inversão fornece um particionamento conveniente de alto nível dos dados. Um sistema eficiente poderia ajustar representações de linguagem separadas por classe, as quais fornecerão respostas para tarefas de PLN. Quando se deseja tratar um documento como sem rótulo, as tarefas de PLN podem ser respondidas através da agregação das respostas específicas por classe.

- **Desempenho**: Foi descoberto que a inversão do *Word2Vec* apresenta maior redução de taxas de classificação errônea do que classificações baseadas tanto no modelo *Doc2Vec* (LE e MIKOLOV, 2014) como na regressão inversa multinomial (MNIR) de Taddy (2013) – considerando-se uma configuração de inicialização simples e padrão durante a classificação de documentos.

Doc2Vec

O modelo é muito similar ao *Word2Vec*, porque procura dividir o documento em *tokens* com semântica bem definida e de diferentes tamanhos, advindos de sequências de caracteres do texto de tamanho variável (como frases ou parágrafos). O método se sobrepõe ao algoritmo *Bag of Words* devido ao fato de este último desconsiderar a ordem e a semântica das palavras. Por exemplo, palavras como “poderoso”, “forte” e “Paris” poderiam ter a mesma distância entre si no modelo *Bag of Words*, o que dificilmente ocorreria em uma representação distribuída como *Doc2Vec* (LE e MIKOLOV, 2014).

Grosso modo, a única diferença entre este método (chamado também de *Paragraph Vector*) e o *Word2Vec* (MIKOLOV et al., 2013a, 2013b) é a adição do *token* "parágrafo" no modelo de predição: cada parágrafo do texto é mapeado em um único vetor. O *token* de parágrafo pode ser pensado como uma outra palavra, atuando como uma memória que lembra o que está faltando no contexto atual, ou o tópico do parágrafo. O vetor de parágrafo é compartilhado entre todos os contextos gerados a partir do mesmo parágrafo, mas não entre parágrafos. A matriz vetorial *W* de palavras, no entanto, é compartilhada entre parágrafos, isto é, o vetor gerado por uma palavra é o mesmo para todos os parágrafos (LE e MIKOLOV, 2014).

Inspirado no modelo de vetores de palavras baseados em redes neurais, o método funciona da seguinte forma: tanto os vetores de parágrafos como os vetores de palavras são treinados usando o descendente de gradiente estocástico, sendo que o gradiente é obtido via retropropagação. No espaço gerado, "poderoso" estaria mais próximo de "forte" do que de "Paris", representando o significado aproximado dos termos.

Um ponto curioso do método *Doc2Vec* é o fato de que a precisão da regressão logística pode mudar quando os vetores do conjunto de teste mudam. Esta circunstância ocorre devido à descida do gradiente para cada tarefa de classificação executada, inferindo o vetor – inicializado aleatoriamente – de um documento que se pretende classificar.

O método mapeia palavras em vetores, mantendo a ordem delas assim como aparecem no texto. A concatenação ou soma dos vetores são usadas como recursos para predizer a próxima palavra na frase.

TF-IDF

O nome deste método é derivado da expressão *Term Frequency – Inverse Document Frequency*. O método se baseia em uma maneira um pouco mais avançada de contar palavras em um documento. O método ajusta para a classificação o comprimento do documento, a frequência de palavras e a frequência de uma palavra específica em um documento específico (LESKOVEC, RAJARAMAN, ULLMAN, 2014).

O método *TF-IDF* está fundamentado em uma estatística numérica destinada a refletir o quão importante é uma palavra para um documento em uma coleção ou córpus (LESKOVEC,

RAJARAMAN, ULLMAN, 2014). O valor *TF-IDF* aumenta proporcionalmente em relação ao número de vezes que uma palavra aparece no documento, mas é compensado pela frequência da palavra no córpus, assimilando o fato de que algumas palavras aparecem com mais frequência no geral (BEEL et al., 2016).

Suponha que temos um conjunto de documentos de texto em português e desejamos determinar quais documentos são mais relevantes para a consulta "a vaca preta". Uma maneira simples de começar é eliminarmos documentos que não contém as palavras "a", "vaca" e "preta", mas isso ainda pode nos deixar com muitos documentos. Para distingui-los de forma mais precisa, devemos contar o número de vezes que cada termo ocorre em cada documento e somá-los: o número de vezes (frequência) que um termo ocorre no documento é chamado de *term frequency*. A primeira forma de atribuir peso aos termos é através do método de Luhn (1957), postulando que o peso de um termo que ocorre em um documento é proporcional à sua frequência de termo.

Utilizando ainda o exemplo do parágrafo anterior, devido ao termo "a" ser muito comum, a frequência desse termo tende a enfatizar incorretamente documentos que contêm a palavra "a" mais frequentemente, sem dar o peso devido aos termos mais significativos como "vaca" e "preta". Dada essa questão, é incorporado ao algoritmo o fator *inverse document frequency* (frequência inversa do documento), o qual diminui o peso dos termos que ocorrem muito frequentemente no documento e aumenta o peso dos termos que ocorrem mais raramente.

Em Sparck Jones (1972) concebeu-se uma interpretação estatística da especificidade do termo, chamada *Inverse Document Frequency* (IDF), a qual se tornou um pilar da ponderação dos termos. A especificidade de um termo pode ser quantificada como uma função inversa do número de documentos nos quais ele ocorre.

Tanto a estatística *term frequency* como a estatística *inverse document frequency* podem ser calculadas de maneiras diferentes, a depender da função criada para cada uma. Exemplo do uso de *term frequency* seria utilizar a frequência absoluta ou relativa dos termos. A estatística *TF-IDF* surge como a multiplicação entre as estatísticas *TF* e *IDF*. Finalmente, supõe-se que quanto maior o valor de *TF-IDF* para um determinado termo, maior é a sua relevância no córpus (BEEL et al., 2016).

Atualmente, *TF-IDF* é um dos esquemas mais populares de ponderação de termos para classificação e recuperação de informações. Por exemplo, 83% dos sistemas recomendadores

baseados em texto no domínio das bibliotecas digitais utilizam este método (BEEL et al., 2016).

Word Mover's Distance

Da mesma forma que o método *Averaging Word Vectors*, este método também utiliza o modelo *Word2Vec* pré-treinado com base em uma grande massa de textos, tirando proveito da técnica *Skip-gram* mencionada na descrição do modelo *Word2Vec* (MIKOLOV et al., 2013a, 2013b) para obter distâncias entre *tokens*. O *Word Mover's Distance* (*WMD*) é um algoritmo recente, desenvolvido por Kusner et al. (2015), cujo maior diferencial em relação aos outros métodos aqui citados é a sua capacidade de alcançar bons resultados em análises de sentimentos.

Segundo Kusner et al. (2015), a distância *WMD* mede a dissimilaridade entre dois documentos de texto como a distância mínima que as palavras incorporadas de um documento precisam "viajar" para alcançar as palavras incorporadas de outro documento. Vale ressaltar a existência de dois tipos de distância nesse estudo: as distâncias entre palavras e entre documentos. A distância *WMD* procura utilizar a similaridade (distância) entre palavras, calculadas pelo *Word2Vec*, para então calcular a similaridade (distância) entre documentos.

Kusner et al. (2015) afirmam que uma forma de medida de dissimilaridade entre palavras é proporcionada naturalmente por sua distância euclidiana no espaço do *Word2Vec*. Já o cálculo da distância entre documentos – por meio da distância *WMD* – é efetuado seguindo 3 passos essenciais:

1. Remova as *stopwords*;
2. Se os documentos possuem a mesma quantidade de palavras: encontre o fluxo entre cada par de palavra correspondente (Fig.1);
3. Se os documentos não possuem a mesma quantidade de palavras: encontre um ou mais fluxos entre pares de palavras semelhantemente correspondentes (Fig.2).

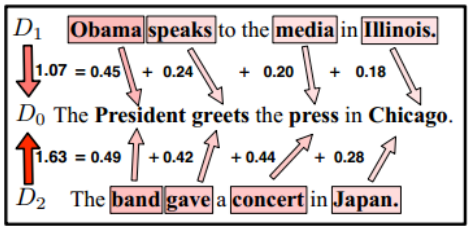

Figura 1. Parte da Figura 2 do artigo de Kusner et al. (2015, p. 3). Compatibilidade entre palavras de documentos de comprimento igual (note que as *stopwords* são desconsideradas). Cada seta, rotulada com seu respectivo peso, representa um fluxo entre duas palavras. A soma dos pesos de cada fluxo resulta na distância entre cada par de documentos.

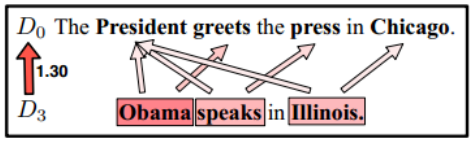

Figura 2. Parte da Figura 2 do artigo de Kusner et al. (2015, p. 3). Compatibilidade entre palavras de documentos de comprimento diferente. Cada seta, rotulada com seu respectivo peso, representa um fluxo entre duas palavras. A incompatibilidade de comprimento faz com que o *WMD* mova palavras de um documento para uma ou mais palavras semelhantes do outro documento, resultando em mais de um fluxo saindo de algumas palavras.

O fluxo entre palavras é criado a partir da maior similaridade entre elas, que por sua vez é extraída do espaço vetorial *Word2Vec* (MIKOLOV et al., 2013a, 2013b). Cada fluxo tem um peso, assim como pode-se verificar na Fig.1, e cada peso de um fluxo é a distância euclidiana (DANIELSSON, 1980) entre o par de palavras no espaço vetorial do *Word2Vec*. Após seguir as três etapas, o cálculo da distância entre um par de documentos é a soma de todos os pesos dos fluxos gerados entre eles (KUSNER et al., 2015).

A distância entre documentos servirá para predizer a que grupo um documento pertence: embasado no modelo concedido, quanto menor a distância entre um par de documentos, mais similar o documento é do outro (KUSNER et al., 2015). Dado um documento de teste, podemos conceder-lhe o rótulo do documento de treino que detém a menor distância *WMD* relacionada a ele.

4 Metodologia

Neste capítulo descrevemos as tarefas de caracterização autoral propostas, os métodos e os conjuntos de dados utilizados nestes experimentos.

4.1 Tarefas de Caracterização Autoral

Os problemas de caracterização autoral a serem considerados em cada base de dados estão dispostos a seguir. A base de dados em português está dividida em 10 classes, e cada classe vinculada à sua tarefa de caracterização, cujas informações são integrantes do córpus *b5-post* (RAMOS et al., 2017; SILVA e PARABONI, 2017):

- ***age*** – Representa a faixa etária dos autores dos documentos, dividida em 3 subclasses: a1820 (de 18 a 20 anos), a2325 e a2861. As faixas intermediárias são omitidas pois a informação de idade disponível é aproximada (não sabemos exatamente qual a idade da pessoa no momento em que publicou o texto no *Facebook*).
- ***course*** – O curso de graduação (da EACH – USP) dos voluntários, com código de 1 a 11 para os cursos mais frequentes e 0 para os demais cursos.
- ***gender*** – Representa o gênero de cada indivíduo, dividido nas subclasses *male* (masculino) e *female* (feminino).
- ***relig*** – O grau de religiosidade de cada indivíduo, em uma escala de 1 (nada religioso) a 5 (muito religioso). A informação é agrupada em três subclasses: r12 para respostas entre 1 e 2, r3 e r45.
- ***ti*** – Indica se o indivíduo possui formação ou atuação em Tecnologia de Informação. Dividido nas duas subclasses *no* (não possui formação ou atuação na área) e *yes* (possui formação ou atuação na área).
- ***ext***, ***cons***, ***agr***, ***neu***, ***ope***: As cinco dimensões de personalidade do modelo *Big Five* (JOHN, DONAHUE, KENTLE, 1991; JOHN, NAUMANN, SOTO, 2008; RAMOS et al., 2017), na forma de subclasses binárias – *no* (não pertencente à classe) e *yes*

(pertencente à classe) – representando Extroversão, Conscienciosidade, Agradabilidade, Neuroticismo e Abertura à experiência. Estas informações foram computadas a partir de inventários de personalidade (ANDRADE, 2008) como parte do córpus *b5-post*.

Tanto a base de dados em inglês como a base de dados em espanhol estão divididas em 2 classes, e cada classe vinculada à sua tarefa de caracterização:

- ***age*** – Representa a faixa etária dos autores dos documentos, dividida em 3 subclasses: 10s (entre 13 e 17 anos), 20s (entre 23 e 27 anos) e 30s (entre 33 e 47 anos) (RANGEL et al., 2013).
- ***gender*** – Representa o gênero de cada indivíduo, dividido nas subclasses *male* (masculino) e *female* (feminino) (RANGEL et al., 2013).

4.2 Modelos de Classificação

Os algoritmos de AM escolhidos para a comparação de desempenho foram: *Bag of Words* (MCCALLUM et al., 1998), N-gramas de caracteres (FUSILIER et al., 2015), *Term Frequency – Inverse Document Frequency* (BEEL et al., 2016; LESKOVEC, RAJARAMAN, ULLMAN, 2014; LUHN, 1957; SPARCK JONES, 1972), *Deep IR* (TADDY, 2015), *Doc2Vec* (LE e MIKOLOV, 2014), *Averaging Word Vectors* (MIKOLOV et al., 2013a, 2013b) e *Word Mover's Distance* (KUSNER et al., 2015). Esses métodos são baseados majoritariamente nas bibliotecas *Gensim* (REHUREK e SOJKA, 2011) e *Sklearn* (PEDREGOSA et al., 2011).

A gama de algoritmos de AM escolhidos varia desde mais antigos – *e.g.*, *Bag of Words* (MCCALLUM et al., 1998) – até mais recentes – *e.g.*, *Word Mover's Distance* (KUSNER et al., 2015). Cada tarefa de caracterização citada nesta seção foi executada por todos os algoritmos de AM.

Seguem as siglas que denotam cada algoritmo, as quais serão utilizadas especificamente na seção 5 (a ausência do algoritmo *Word Mover's Distance* será discutida na seção 5.3 também):

- **BoW**: *Bag of Words*. Usa um extrator de *tokens* da biblioteca *NLTK* (BIRD, 2006), que divide o texto contido entre espaços, tabulações e quebra de linha. O vocabulário é limitado nas 3000 palavras mais frequentes de todas as analisadas no córpus de treino, e as utiliza para atribuir os pesos à regressão logística. No nosso experimento, não são removidas *stopwords* (palavras de relevância desprezível, por exemplo preposições e artigos) nos dados em português e espanhol, mas somente nos dados em inglês, devido ao fato de a biblioteca *Sklearn* (PEDREGOSA et al., 2011) oferecer suporte apenas ao idioma inglês.
- **Char**: N-gramas de caracteres. Escolhemos o comprimento de cada n-grama de caractere entre 3 e 6 caracteres e utilizamos somente os 3000 n-gramas de caracteres mais frequentes.
- **Tf_Idf**: *Term Frequency – Inverse Document Frequency*. O método extrai os principais *tokens* com os melhores valores de *TF-IDF* e então treina um modelo de regressão logística.
- **Deep_IR**: *Deep IR*. A extração de *tokens* é diferente de outros métodos: o texto é posto em caixa-baixa e as palavras são separadas por pontuação e espaços em branco, conforme discutido em Taddy (2015). Como propõe Taddy (2015), é feita a divisão do documento em frases. Além disso, são eliminados os caracteres especiais do texto.
- **Doc2VecV1**: 1ª implementação do *Doc2Vec*. Utiliza os *tokens* criados pelo modelo *Doc2Vec* para executar o algoritmo *KNN* (ZHANG e ZHOU, 2007) que realiza a tarefa de classificação propriamente dita, com *k* igual a 1, usando a métrica *Cosine* (LIAO e XU, 2015) para calcular a distância entre vizinhos.
- **Doc2VecV2**: 2ª implementação do *Doc2Vec*. Utiliza os *tokens* criados pelo modelo *Doc2Vec* para treinar uma regressão logística. A regressão conta com uma inicialização padrão pseudoaleatória dos pesos.
- **Doc2VecV3**: 3ª implementação do *Doc2Vec*. Utiliza os *tokens* criados pelo modelo *Doc2Vec* para treinar uma regressão logística. A regressão conta com uma inicialização pseudoaleatória gerada a partir de uma semente arbitrária.
- **AveragV1**: 1ª implementação do *Averaging Word Vectors*. Após os *tokens* serem extraídos pelo *Word2Vec*, executamos o algoritmo de classificação *KNN* (ZHANG e ZHOU, 2007) com valor de *k* igual a 3, utilizando a métrica *Cosine* (LIAO e XU, 2015) para calcular a distância entre vizinhos.

- **AveragV2**: 2ª implementação do *Averaging Word Vectors*. Baseada em regressão logística, utiliza as representações vetoriais dos *tokens* (extraídos dos documentos de treino pelo *Word2Vec*) para realizar a tarefa de classificação.

5 Avaliação

Este capítulo apresenta os resultados das tarefas de caracterização discutidas no capítulo anterior. Para compararmos o desempenho dos algoritmos mencionados, utilizaremos os valores de precisão, revocação e medida F.

A medida de precisão é definida como a porcentagem de instâncias recuperadas que são relevantes, cujo resultado é a proporção de verdadeiros positivos sobre a soma entre verdadeiros positivos e falsos positivos. A fórmula da precisão é definida a seguir (GOUTTE, GAUSSIER, 2005, p. 347):

$$p = \frac{Verdadeiros\,Positivos}{Verdadeiros\,Positivos + Falsos\,Positivos}$$

A revocação é a porcentagem de instâncias relevantes que são recuperadas, cujo resultado é a proporção de verdadeiros positivos sobre a soma entre verdadeiros positivos e falsos negativos. A fórmula da revocação é definida a seguir (GOUTTE, GAUSSIER, 2005, p. 347):

$$r = \frac{Verdadeiros\,Positivos}{Verdadeiros\,Positivos + Falsos\,Negativos}$$

Já a medida F é igual à média harmônica entre precisão e revocação. A fórmula da medida F é definida a seguir (GOUTTE, GAUSSIER, 2005, p. 89):

$$F = \frac{2}{\frac{1}{precisão} + \frac{1}{revocação}}$$

5.1 Conjuntos de Dados

Fizemos uso de três conjuntos de dados para comparar o desempenho entre os métodos de classificação. O primeiro conjunto de dados está em português, chamado de córpus *b5-post* (RAMOS et al., 2017; SILVA e PARABONI, 2017), que não possui os dados delimitados em

subconjuntos de treino e de teste. A tabela 1 mostra um resumo das instâncias dessa base de dados em português, bem como informações adicionais de cada classe.

Português (Pt)	Classe	Tipo	Instâncias	Total de Palavras
Treino/Teste	Idade	Nominal	[a1820] = 185, [a2325] = 190, [a2861] = 145	1438405
Treino/Teste	Curso	Nominal	[1]=195, [2]=46, [3]=39, [4]=36, [5]=27, [6]=27, [7]=28, [8]=20, [9]=14, [10]=9, [11]=26	1256454
Treino/Teste	Gênero	Binária	[masculino] = 444, [feminino] = 583	2883443
Treino/Teste	Religião	Nominal	[r12] = 218, [r3] = 97, [r45] = 127	1230744
Treino/Teste	TI	Binária	[positivas] = 325, [negativas] = 496	2313674
Treino/Teste	Agradab.	Binária	[positivas] = 542, [negativas] = 486	2884808
Treino/Teste	Cons.	Binária	[positivas] = 510, [negativas] = 518	2884808
Treino/Teste	Extrov.	Binária	[positivas] = 508, [negativas] = 520	2884808
Treino/Teste	Neurot.	Binária	[positivas] = 498, [negativas] = 530	2884808
Treino/Teste	Abert.	Binária	[positivas] = 536, [negativas] = 492	2884808

Tabela 1. Informações complementares da base de dados em português.

Os outros dois conjuntos de dados possuem uma configuração similar entre si: tanto o conjunto de dados em inglês (RANGEL et al., 2013) como o conjunto de dados em espanhol (RANGEL et al., 2013) têm os dados delimitados em 2 subconjuntos de teste e 1 subconjunto de treino cada. A tabela 2 mostra um resumo das instâncias do conjunto de dados em inglês, enquanto a tabela 3 mostra um resumo dos dados em espanhol, além de informações adicionais de cada classe no seu contexto.

Inglês (En)	Classe	Tipo	Instâncias	Total de Palavras
Treino	Idade	Nominal	[10s] = 17200, [20s] = 85800, [30s] = 133600	174235131
Treino	Gênero	Binária	[masculino] = 118300, [feminino] = 118300	174235131
Teste (1)	Idade	Nominal	[10s] = 12040, [20s] = 1480, [30s] = 7680	9983469
Teste (1)	Gênero	Binária	[masculino] = 10600, [feminino] = 10600	9983469
Teste (2)	Idade	Nominal	[10s] = 14448, [20s] = 1776, [30s] = 9034	13273486
Teste (2)	Gênero	Binária	[masculino] = 12538, [feminino] = 12720	13273486

Tabela 2. Informações complementares da base de dados em inglês.

Espanhol (Es)	Classe	Tipo	Instâncias	Total de Palavras
Treino	Idade	Nominal	[10s] = 2500, [20s] = 42600, [30s] = 30800	19897730
Treino	Gênero	Binária	[masculino] = 37950, [feminino] = 37950	19897730
Teste (1)	Idade	Nominal	[10s] = 2720, [20s] = 3840, [30s] = 240	1587446
Teste (1)	Gênero	Binária	[masculino] = 3400, [feminino] = 3400	1587446
Teste (2)	Idade	Nominal	[10s] = 3264, [20s] = 4608, [30s] = 288	1889618
Teste (2)	Gênero	Binária	[masculino] = 4080, [feminino] = 4080	1889618

Tabela 3. Informações complementares da base de dados em espanhol.

5.2 Procedimento

Visto que a base de dados em português não possui subconjuntos específicos de treino e de teste, além de que a base é pequena, foi efetuada uma validação cruzada para cada tarefa de caracterização utilizando-se o método *k-folds* com *k* igual a 10 (valor arbitrário). Contrariamente, devido às bases de dados em inglês e em espanhol possuírem subconjuntos previamente delimitados em treino e teste, não foi executada validação cruzada, para que os resultados deste experimento possam ser comparados com possíveis implementações futuras que executem estas tarefas com base nestes mesmos dados.

Geramos uma tabela de precisão, revocação e medida F para cada tarefa de caracterização mencionada na seção 4.1, com todos os valores arredondados em 2 casas decimais. Posto que os dados em inglês e espanhol possuem 2 subconjuntos de teste, especialmente para eles as tabelas de resultados apresentam a média dos valores obtidos pelas tarefas de caracterização de ambos subconjuntos de teste. As tabelas contêm uma coluna “Média F1” correspondente à média das medidas F, ponderada pelas instâncias de cada subclasse da tabela em questão.

Procuramos seguir também a abordagem padrão de cada algoritmo em cada tarefa de caracterização, com implementações mais usuais ou sem alterar parâmetros pré-definidos pelas bibliotecas empregadas. Fica sugerido como trabalho futuro a execução desses métodos utilizando parâmetros otimizados.

5.3 Resultados

A única inviabilidade deste trabalho foi concluir as tarefas de caracterização usando o algoritmo *Word Mover's Distance* (KUSNER et al., 2015), devido à sua alta complexidade e escassez de tempo e recursos computacionais. A título de exemplificação, a execução da primeira tarefa de caracterização relacionada a este método perdurou por aproximadamente 6 dias sem sua conclusão efetiva. Sugerimos como trabalho futuro a execução e uso comparativo de desempenho desse método de classificação com recursos de *hardware* mais potentes.

As tarefas de caracterização executadas pelo método *Averaging Word Vectors* (MIKOLOV et al., 2013a, 2013b) foram concluídas empregando-se apenas a base de dados em inglês, devido ao modelo pré-treinado utilizar somente dados em inglês.

Os resultados para cada idioma são apresentados individualmente a seguir. Foram assinalados em azul os maiores valores de média de medida F e em laranja os menores valores, também foram destacados os maiores valores de medida F para cada subclasse dentro de cada tabela.

1. **Português**

Faixa Etária	**a1820**			**a2325**			**a2861**			
	P	R	**F1**	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,56	0,59	0,57	0,48	0,50	0,49	0,66	0,55	0,59	0,55
Char	0,56	0,58	0,57	0,47	0,45	0,45	0,57	0,53	0,54	0,52
Tf_Idf	0,63	0,62	**0,62**	0,56	0,57	**0,56**	0,67	0,65	**0,65**	**0,61**
Deep_IR	0,60	0,48	0,52	0,46	0,66	0,54	0,63	0,41	0,48	0,52
Doc2VecV1	0,47	0,42	0,44	0,41	0,38	0,39	0,34	0,43	0,37	**0,40**
Doc2VecV2	0,46	0,58	0,51	0,44	0,35	0,38	0,46	0,40	0,43	0,44
Doc2VecV3	0,45	0,58	0,51	0,41	0,36	0,38	0,47	0,37	0,41	0,43

Tabela 4. Precisão, revocação e medida F relacionadas à tarefa *age* da base de dados em português.

Curso		BoW	Char	Tf_Idf	Deep_IR	Doc2VecV1	Doc2VecV2	Doc2VecV3
1	P	0,61	0,57	0,61	0,62	0,70	0,50	0,50
	R	0,67	0,66	0,84	0,07	0,30	0,95	0,95
	F1	0,63	0,60	**0,70**	0,12	0,41	0,65	0,65
2	P	0,28	0,20	0,25	0,10	0,00	0,26	0,17
	R	0,25	0,17	0,26	0,03	0,00	0,11	0,09
	F1	0,23	0,18	**0,25**	0,04	0,00	0,14	0,10
3	P	0,05	0,01	0,19	0,00	0,07	0,10	0,15
	R	0,08	0,01	0,21	0,00	0,04	0,03	0,07
	F1	0,06	0,01	**0,18**	0,00	0,05	0,05	0,09
4	P	0,08	0,12	0,11	0,00	0,05	0,00	0,00
	R	0,10	0,14	0,09	0,00	0,07	0,00	0,00
	F1	0,09	**0,13**	0,10	0,00	0,06	0,00	0,00
5	P	0,19	0,17	0,24	0,00	0,09	0,00	0,00
	R	0,22	0,11	0,22	0,00	0,16	0,00	0,00
	F1	0,20	0,13	**0,21**	0,00	0,11	0,00	0,00
6	P	0,10	0,05	0,27	0,00	0,07	0,00	0,00
	R	0,16	0,06	0,27	0,00	0,21	0,00	0,00
	F1	0,12	0,05	**0,27**	0,00	0,10	0,00	0,00
7	P	0,23	0,18	0,13	0,00	0,11	0,03	0,03
	R	0,17	0,18	0,12	0,00	0,33	0,05	0,05
	F1	**0,18**	0,17	0,11	0,00	0,17	0,04	0,03
8	P	0,28	0,10	0,52	0,10	0,05	0,15	0,20
	R	0,20	0,10	0,35	0,05	0,05	0,10	0,10
	F1	0,22	0,10	**0,40**	0,07	0,05	0,12	0,13
9	P	0,00	0,00	0,00	0,00	0,01	0,00	0,00
	R	0,00	0,00	0,00	0,00	0,02	0,00	0,00
	F1	0,00	0,00	0,00	0,00	**0,01**	0,00	0,00
10	P	0,00	0,00	0,00	0,00	0,00	0,00	0,00
	R	0,00	0,00	0,00	0,00	0,00	0,00	0,00
	F1	**0,00**	**0,00**	**0,00**	**0,00**	**0,00**	**0,00**	**0,00**
11	P	0,00	0,13	0,03	0,07	0,01	0,00	0,00
	R	0,00	0,15	0,05	0,79	0,03	0,00	0,00
	F1	0,00	**0,14**	0,03	0,12	0,01	0,00	0,00
Média F1		0,34	0,31	**0,39**	**0,06**	0,21	0,30	0,30

Tabela 5. Precisão, revocação e medida F relacionadas à tarefa *course* da base de dados em português.

Gênero	Feminino			Masculino			
	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,82	0,81	0,82	0,76	0,76	0,76	0,79
Char	0,81	0,81	0,81	0,76	0,75	0,75	0,78
Tf_Idf	0,83	0,86	**0,85**	0,81	0,77	**0,79**	**0,82**
Deep_IR	0,58	0,99	0,73	0,93	0,05	0,10	**0,46**
Doc2VecV1	0,76	0,70	0,73	0,65	0,70	0,67	0,70
Doc2VecV2	0,70	0,80	0,74	0,68	0,55	0,60	0,68
Doc2VecV3	0,70	0,79	0,74	0,68	0,56	0,61	0,68

Tabela 6. Precisão, revocação e medida F relacionadas à tarefa *gender* da base de dados em português.

Religião	**r12**			**r3**			**r45**			
	P	R	**F1**	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,57	0,61	0,58	0,22	0,19	0,20	0,43	0,43	0,42	0,45
Char	0,57	0,59	0,57	0,27	0,27	0,26	0,37	0,35	0,35	**0,44**
Tf_Idf	0,60	0,71	0,64	0,33	0,19	0,24	0,47	0,45	0,45	0,50
Deep_IR	0,61	0,85	0,71	0,20	0,02	0,03	0,55	0,52	**0,53**	**0,51**
Doc2VecV1	0,65	0,48	0,55	0,27	0,45	**0,34**	0,47	0,44	0,45	0,48
Doc2VecV2	0,60	0,90	**0,72**	0,00	0,00	0,00	0,54	0,49	0,51	0,50
Doc2VecV3	0,61	0,90	**0,72**	0,00	0,00	0,00	0,53	0,48	0,50	0,50

Tabela 7. Precisão, revocação e medida F relacionadas à tarefa *relig* da base de dados em português.

TI	**Não**			**Sim**			
	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,73	0,75	0,74	0,60	0,57	0,58	0,68
Char	0,70	0,71	0,70	0,55	0,52	0,54	0,64
Tf_Idf	0,75	0,80	**0,78**	0,66	0,59	**0,62**	**0,72**
Deep_IR	0,67	0,92	**0,78**	0,74	0,31	0,42	0,64
Doc2VecV1	0,73	0,63	0,68	0,54	0,65	0,58	0,64
Doc2VecV2	0,68	0,87	0,76	0,66	0,36	0,46	0,64
Doc2VecV3	0,67	0,87	0,76	0,65	0,34	0,44	**0,63**

Tabela 8. Precisão, revocação e medida F relacionadas à tarefa *ti* da base de dados em português.

Extroversão	**Não**			**Sim**			
	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,57	0,58	0,57	0,57	0,56	0,56	0,57
Char	0,52	0,48	0,49	0,51	0,55	0,53	**0,51**
Tf_Idf	0,59	0,52	0,55	0,56	0,62	0,59	0,57
Deep_IR	0,61	0,38	0,46	0,55	0,76	**0,64**	0,55
Doc2VecV1	0,64	0,46	0,53	0,57	0,73	**0,64**	0,58
Doc2VecV2	0,60	0,71	**0,65**	0,63	0,52	0,57	0,61
Doc2VecV3	0,61	0,69	**0,65**	0,63	0,55	0,58	**0,62**

Tabela 9. Precisão, revocação e medida F relacionadas à tarefa *ext* da base de dados em português.

Conscienciosidade	**Não**			**Sim**			
	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,53	0,55	0,54	0,52	0,50	0,51	**0,53**
Char	0,55	0,55	0,55	0,54	0,54	0,54	0,55
Tf_Idf	0,57	0,59	0,58	0,57	0,55	0,55	**0,57**
Deep_IR	0,58	0,57	0,56	0,59	0,58	0,57	0,56
Doc2VecV1	0,59	0,44	0,50	0,55	0,69	**0,61**	0,55
Doc2VecV2	0,57	0,64	**0,60**	0,58	0,50	0,53	**0,57**
Doc2VecV3	0,56	0,64	**0,60**	0,57	0,49	0,53	**0,57**

Tabela 10. Precisão, revocação e medida F relacionadas à tarefa *cons* da base de dados em português.

Agradabilidade	**Não**			**Sim**			
	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,48	0,48	0,48	0,53	0,52	0,53	**0,51**
Char	0,48	0,49	0,48	0,55	0,54	0,54	**0,51**
Tf_Idf	0,52	0,52	**0,52**	0,58	0,57	0,57	**0,55**
Deep_IR	0,54	0,43	0,48	0,57	0,68	0,62	**0,55**
Doc2VecV1	0,57	0,32	0,40	0,56	0,78	**0,65**	0,53
Doc2VecV2	0,54	0,40	0,45	0,56	0,69	0,62	0,54
Doc2VecV3	0,55	0,41	0,46	0,56	0,69	0,62	0,54

Tabela 11. Precisão, revocação e medida F relacionadas à tarefa *agr* da base de dados em português.

Neuroticismo	**Não**			**Sim**			
	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,57	0,54	0,55	0,54	0,57	0,55	0,55
Char	0,55	0,53	0,54	0,52	0,54	0,53	0,54
Tf_Idf	0,58	0,54	0,56	0,54	0,57	**0,56**	**0,56**
Deep_IR	0,53	0,53	0,53	0,51	0,52	0,51	0,52
Doc2VecV1	0,53	0,45	0,48	0,49	0,57	0,52	0,50
Doc2VecV2	0,52	0,72	0,60	0,52	0,31	0,38	**0,49**
Doc2VecV3	0,53	0,73	**0,61**	0,53	0,30	0,38	0,50

Tabela 12. Precisão, revocação e medida F relacionadas à tarefa *neu* da base de dados em português.

Abertura à experiência	**Não**			**Sim**			
	P	R	**F1**	P	R	**F1**	**Média F1**
BoW	0,49	0,52	0,50	0,53	0,50	0,52	**0,51**
Char	0,48	0,50	0,49	0,52	0,51	0,52	**0,51**
Tf_Idf	0,51	0,50	0,51	0,55	0,56	0,56	0,54
Deep_IR	0,59	0,29	0,39	0,55	0,80	**0,65**	0,53
Doc2VecV1	0,52	0,70	**0,59**	0,59	0,40	0,47	0,53
Doc2VecV2	0,53	0,49	0,51	0,56	0,60	0,58	**0,55**
Doc2VecV3	0,53	0,48	0,50	0,55	0,59	0,57	0,54

Tabela 13. Precisão, revocação e medida F relacionadas à tarefa *ope* da base de dados em português.

2. Inglês

Faixa Etária	10s			20s			30s			
	P	R	F1	P	R	F1	P	R	F1	Média F1
BoW	0,77	0,60	0,67	0,13	0,00	0,00	0,53	0,82	**0,65**	0,62
Char	0,73	0,66	0,69	0,52	0,01	0,02	0,54	0,72	0,62	0,62
Tf_Idf	0,72	0,63	0,68	0,18	0,09	0,12	0,53	0,68	0,60	0,61
Deep_IR	0,77	0,55	0,64	0,14	0,13	0,13	0,54	0,77	0,63	0,60
Doc2VecV1	0,68	0,21	0,33	0,07	0,32	0,12	0,46	0,65	0,54	**0,39**
Doc2VecV2	0,61	0,86	0,71	0,00	0,00	0,00	0,52	0,29	0,37	0,54
Doc2VecV3	0,61	0,86	0,71	0,00	0,00	0,00	0,52	0,29	0,37	0,54
AveragV1	0,70	0,62	0,66	0,13	0,15	**0,14**	0,52	0,60	0,56	0,58
AveragV2	0,71	0,74	**0,72**	0,00	0,00	0,00	0,57	0,65	0,61	**0,63**

Tabela 14. Precisão, revocação e medida F relacionadas à tarefa *age* da base de dados em inglês.

Gênero	Masculino			Feminino			
	P	R	F1	P	R	F1	Média F1
BoW	0,65	0,32	0,43	0,55	0,83	**0,66**	0,55
Char	0,61	0,49	0,54	0,57	0,69	0,62	0,58
Tf_Idf	0,58	0,50	0,54	0,56	0,63	0,60	0,57
Deep_IR	0,57	0,64	**0,60**	0,59	0,51	0,55	0,57
Doc2VecV1	0,51	0,40	0,44	0,51	0,63	0,56	**0,51**
Doc2VecV2	0,57	0,32	0,41	0,53	0,76	0,62	0,52
Doc2VecV3	0,57	0,32	0,41	0,53	0,76	0,63	0,52
AveragV1	0,57	0,56	0,56	0,57	0,59	0,58	0,57
AveragV2	0,63	0,54	0,58	0,60	0,69	0,64	**0,61**

Tabela 15. Precisão, revocação e medida F relacionadas à tarefa *gender* da base de dados em inglês.

3. Espanhol

Faixa Etária	10s			20s			30s			
	P	R	F1	P	R	F1	P	R	F1	Média F1
BoW	0,67	0,33	0,44	0,62	0,89	**0,73**	0,00	0,00	0,00	0,59
Char	0,65	0,31	0,42	0,62	0,89	**0,73**	0,10	0,00	0,01	0,58
Tf_Idf	0,58	0,56	**0,57**	0,67	0,68	0,67	0,12	0,15	**0,13**	**0,61**
Deep_IR	0,60	0,51	0,55	0,66	0,74	0,70	0,05	0,05	0,05	**0,61**
Doc2VecV1	0,51	0,34	0,41	0,62	0,46	0,53	0,05	0,44	0,09	**0,46**
Doc2VecV2	0,62	0,11	0,19	0,58	0,95	0,72	0,00	0,00	0,00	0,48
Doc2VecV3	0,62	0,11	0,19	0,58	0,95	0,72	0,00	0,00	0,00	0,48

Tabela 16. Precisão, revocação e medida F relacionadas à tarefa *age* da base de dados em espanhol.

Gênero	Masculino			Feminino			
	P	R	F1	P	R	F1	Média F1
BoW	0,63	0,64	**0,64**	0,64	0,62	0,63	**0,63**
Char	0,61	0,62	0,62	0,62	0,60	0,61	0,62
Tf_Idf	0,58	0,59	0,59	0,59	0,58	0,58	0,58
Deep_IR	0,64	0,28	0,39	0,54	0,85	**0,66**	**0,53**
Doc2VecV1	0,56	0,40	0,47	0,53	0,68	0,60	0,54
Doc2VecV2	0,55	0,55	0,55	0,55	0,55	0,55	0,55
Doc2VecV3	0,54	0,55	0,55	0,55	0,54	0,54	0,55

Tabela 17. Precisão, revocação e medida F relacionadas à tarefa *gender* da base de dados em espanhol.

5.4 Discussão

Nesta seção, discutiremos os melhores métodos de classificação a partir de diferentes pontos de vista. Utilizaremos a média das medidas F de cada tabela da seção 5.3 como principal critério de comparação entre os métodos.

Como parâmetro de análise, é perceptível que as versões *Doc2VecV2* e *Doc2VecV3* do modelo *Doc2Vec* apresentam implementações e resultados muito semelhantes. Por esse fato, denotaremos ambas as versões como "modelo *Doc2Vec* baseado em regressão logística", enquanto a implementação *Doc2VecV1* será denotada por "modelo *Doc2Vec* baseado no algoritmo *KNN*".

Outro fator a ser levado em conta é a tarefa *course* da base de dados em português, que pode ser considerada como um *outlier* na interpretação dos dados. Isso se deve à sua pequena quantidade de instâncias por subclasse e ao baixo desempenho relacionado a todos os métodos de classificação considerados.

5.4.1 Análise dos resultados por idioma

Sobre a base de dados em português, consideramos as implementações *Doc2VecV2* e *Doc2VecV3* como uma única implementação, então se ambas as implementações possuem as maiores médias de medida F em uma tabela, será contabilizada a pontuação do *Doc2Vec*

baseado em regressão logística apenas uma vez. Os métodos que apresentaram as melhores médias das medidas F – considerando os empates – foram o *TF-IDF* (melhor valor em 7 das 10 tarefas de caracterização), depois o *Doc2Vec* baseado em regressão logística (melhor valor em 3 tarefas) e por último o método *Deep IR* (melhor valor em 2 tarefas).

A base de dados em inglês contou com unanimidade no resultado. O método que alcançou melhor desempenho foi o *Averaging Word Vectors* baseado em regressão logística, obtendo os maiores valores das médias de medida F nas duas tarefas de caracterização (tabelas 14 e 15).

Em relação à base de dados em espanhol, houve um empate entre os métodos *TF-IDF* e *Deep IR* na tarefa relacionada à faixa etária (tabela 16). Já na tarefa relacionada ao gênero, o método *Bag of Words* apresentou o melhor resultado (tabela 17).

De forma geral, em relação às tarefas dos dados em português, o método que apresentou os melhores resultados foi o *TF-IDF*. Sobre os dados em inglês, a implementação do método *Averaging Word Vectors* baseada em regressão logística apresentou os melhores resultados, enquanto houve um empate entre os métodos *TF-IDF*, *Deep IR* e *Bag of Words* no que diz respeito aos dados em espanhol.

5.4.2 Análise dos resultados por tarefa

Tendo em vista as tarefas *age* e *gender*, consideraremos os três conjuntos de dados simultaneamente para as análises dos métodos, devido ao fato destas informações serem as únicas disponíveis nos três córpus. As tarefas *course*, *relig* e *ti* serão consideradas separadamente, enquanto as tarefas *ext*, *cons*, *agr*, *neu* e *ope* serão interpretadas como uma única tarefa, representando o fator de personalidade *Big Five* (JOHN, DONAHUE, KENTLE, 1991; JOHN, NAUMANN, SOTO, 2008; RAMOS et al., 2017).

Utilizaremos uma pontuação baseada na quantidade de vezes que o método aparece com o maior valor da coluna “Média F1” para determinar o melhor método. Como critério de desempate, utilizaremos o maior valor do atributo “Média F1” relacionado ao método no escopo da tarefa de caracterização em questão.

A tarefa de caracterização autoral referente à faixa etária obteve os melhores resultados utilizando o método *TF-IDF* (tabelas 4 e 16), depois o *Averaging Word Vectors* baseado em regressão logística (tabela 14) e por último o *Deep IR* (tabela 16). Por outro lado, a tarefa de caracterização referente ao gênero obteve os melhores resultados utilizando o método *TF-IDF* (tabela 6), depois o método *Bag of Words* (tabela 17) e por último o método *Averaging Word Vectors* baseado em regressão logística (tabela 15).

Os métodos que obtiveram os melhores resultados executando as tarefas *course*, *relig* e *ti* foram respectivamente *TF-IDF* (tabela 5), *Deep IR* (tabela 7) e *TF-IDF* (tabela 8). Já as tarefas do *Big Five* (JOHN, DONAHUE, KENTLE, 1991; JOHN, NAUMANN, SOTO, 2008; RAMOS et al., 2017) foram executadas mais eficientemente através dos métodos *TF-IDF* e *Doc2Vec* baseado em regressão logística – ambos empataram –, permanecendo também com certa relevância o método *Deep IR*.

5.4.3 Análise geral dos resultados

Podemos destacar o método *TF-IDF* por obter o maior valor de média das medidas F dentre todas as tarefas de caracterização, igual a 0.82 na tarefa *gender* do conjunto de dados em português (tabela 6). Por outro lado, desconsiderando a tarefa *course* mencionada, o método *Doc2Vec* baseado no algoritmo *KNN* obteve o menor valor desse atributo, equivalente a 0.39 na tarefa *age* do conjunto de dados em inglês (tabela 14).

Estabelecemos o seguinte critério referente aos melhores métodos: para cada tabela da seção 5.3, o método que obtiver o melhor valor na coluna “Média F1” recebe um ponto. No final, o método que obtiver mais pontos será considerado o melhor.

Os métodos que obtiveram melhor desempenho nas tarefas de caracterização autoral são, por ordem de desempenho, *TF-IDF*, *Averaging Word Vectors* baseado em regressão logística – decidimos colocá-lo na segunda posição devido à unanimidade na base de dados em inglês –, as implementações do *Doc2Vec* baseadas em regressão logística e o *Deep IR* (estes dois últimos métodos empataram).

6 Conclusão

Este trabalho apresentou uma série de experimentos de classificação de documentos para caracterização autoral, na forma de textos em português, inglês e espanhol. Foram avaliados os métodos *Bag of Words* (MCCALLUM et al., 1998), N-gramas de caracteres (FUSILIER et al., 2015), *TF-IDF* (BEEL et al., 2016; LESKOVEC, RAJARAMAN, ULLMAN, 2014; LUHN, 1957; SPARCK JONES, 1972), *Deep IR* (TADDY, 2015), *Doc2Vec* (LE e MIKOLOV, 2014) e *Averaging Word Vectors* (MIKOLOV et al., 2013a, 2013b) através da execução das tarefas de caracterização referentes às bases de dados mencionadas.

Como trabalho futuro, sugere-se que sejam feitas implementações de métodos que incorporem as ideias dos métodos citados que obtiveram os melhores resultados, sendo eles *TF-IDF* (BEEL et al., 2016; LESKOVEC, RAJARAMAN, ULLMAN, 2014; LUHN, 1957; SPARCK JONES, 1972), *Doc2Vec* (LE e MIKOLOV, 2014), *Averaging Word Vectors* (MIKOLOV et al., 2013a, 2013b) e *Deep IR* (TADDY, 2015). Como exemplo, poderíamos associar as ideias do *TF-IDF* ao *Doc2Vec*, classificando os documentos através da divisão do texto em *tokens* com semântica bem definida, além de selecionar os *tokens* que apresentam maior relevância no texto com base em sua frequência relativa.

Recomendamos que seja executado o método *Word Mover's Distance* (KUSNER et al., 2015) com recursos computacionais mais potentes e maior disponibilidade de tempo, e/ou então que possa ser efetuada uma implementação mais eficiente deste método de classificação, a fim de comparar com os outros métodos descritos neste trabalho. Por último, propomos que os métodos utilizados neste trabalho sejam executados com parâmetros otimizados para demais comparações.

Referências Bibliográficas

ANDRADE, Josemberg Moura de. Evidências de validade do inventário dos cinco grandes fatores de personalidade para o Brasil. **Doctoral dissertation, Programa de Pós-Graduação em Psicologia social, do trabalho e das organizações**, Instituto de Psicologia, Universidade de Brasília, Brasília, DF, 2008.

BEEL, Joeran et al. paper recommender systems: a literature survey. **International Journal on Digital Libraries**, v. 17, n. 4, 2016. p. 305-338.

BIRD, Steven. NLTK: the natural language toolkit. In: **Proceedings of the COLING/ACL on Interactive presentation sessions**. Association for Computational Linguistics, 2006. p. 69-72.

DANIELSSON, Per-Erik. Euclidean distance mapping. **Computer Graphics and image processing**, v. 14, n. 3, p. 227-248, 1980.

FUSILIER, Donato Hernández et al. Detection of opinion spam with character n-grams. In: **International Conference on Intelligent Text Processing and Computational Linguistics**, 2015. Springer, Cham. p. 285-294.

GOLDBERG, Yoav; LEVY, Omer. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. **arXiv preprint arXiv:1402.3722**, 2014.

GOUTTE, Cyril; GAUSSIER, Eric. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In: **ECIR**. 2005. p. 345-359.

JOHN, Oliver P.; DONAHUE, Eileen M.; KENTLE, Robert L. The big five inventory – versions 4a and 54. **Institute of Personality and Social Research**, University of California, Berkeley, 1991.

JOHN, Oliver P.; NAUMANN, Laura P.; SOTO, Christopher J. Paradigm shift to the integrative big five trait taxonomy. **Handbook of personality: Theory and research**, v. 3, 2008. p. 114-158.

KNUTH, Donald E. Dynamic huffman coding. **Journal of algorithms**, v. 6, n. 2, p. 163-180, 1985.

KUSNER, Matt et al. From word embeddings to document distances. In: **International Conference on Machine Learning**, 2015. p. 957-966.

LE, Quoc; MIKOLOV, Tomas. Distributed representations of sentences and documents. In: **Proceedings of the 31st International Conference on Machine Learning (ICML-14)**, 2014. p. 1188-1196.

LESKOVEC, Jure; RAJARAMAN, Anand; ULLMAN, Jeffrey David. **Mining of massive datasets**. Cambridge university press, 2014. p. 1-17.

LIAO, Huchang; XU, Zeshui. Approaches to manage hesitant fuzzy linguistic information based on the cosine distance and similarity measures for HFLTSs and their application in qualitative decision making. **Expert Systems with Applications**, v. 42, n. 12, p. 5328-5336, 2015.

LUHN, Hans Peter. A statistical approach to mechanized encoding and searching of literary information. **IBM Journal of research and development**, v. 1, n. 4, 1957. p. 309-317.

MCCALLUM, Andrew et al. A comparison of event models for naive bayes text classification. In: **AAAI-98 workshop on learning for text categorization**, 1998. p. 41-48.

MIKOLOV, Tomas et al. Distributed representations of words and phrases and their compositionality. In: **Advances in neural information processing systems**, 2013. p. 3111-3119.

MIKOLOV, Tomas et al. Efficient estimation of word representations in vector space. **arXiv preprint arXiv:1301.3781**, 2013.

PEDREGOSA, Fabian et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, n. Oct, 2011. p. 2825-2830.

RAMOS, Ricelli MS et al. Relatório Técnico PPgSI-001/2017 O córpus b5 de textos e inventários de personalidade (v. 1.0.). São Paulo, SP, 2017.

RANGEL, Francisco et al. Overview of the author profiling task at PAN 2013. **Notebook Papers of CLEF**, 2013. p. 23-26.

REHUREK, R.; SOJKA, P. Gensim–python framework for vector space modelling. **NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic**, 2011.

SILVA, B. B. C.; PARABONI, I. Learning personality traits from Facebook text. **IEEE Latin America** (to appear), 2017.

SPARCK JONES, Karen. A statistical interpretation of term specificity and its application in retrieval. **Journal of documentation**, v. 28, n. 1, 1972. p. 11-21.

TADDY, Matt. Document Classification by Inversion of Distributed Language Representations. In: **53th ACL conference**, 2015. p. 45-49.

TADDY, Matt. Multinomial inverse regression for text analysis. **Journal of the American Statistical Association**, v. 108, n. 503, p. 755-770, 2013.

VAPNIK, Vladimir Naumovich. **Statistical learning theory**. New York: Wiley, 1998.

ZHANG, Min-Ling; ZHOU, Zhi-Hua. ML-KNN: A lazy learning approach to multi-label learning. **Pattern recognition**, v. 40, n. 7, p. 2038-2048, 2007.

APÊNDICE A – Repositório do código em *Python*

Tanto o tutorial como o código utilizado nas tarefas de caracterização estão disponíveis no *link* que segue. Recomendamos demasiadamente o uso deste código para tarefas similares de classificação, estaremos disponíveis também para tirar dúvidas ou corrigir eventuais *bugs*.

https://github.com/matszrmn/TCC

APÊNDICE B – Repositórios dos dados de saída

Nesta seção estão disponibilizadas as matrizes de confusão provenientes de cada tarefa de caracterização, e cada matriz presente em 3 formatos: imagem, documento de texto e planilha *CSV*. Estão disponibilizadas também as tabelas PRF no formato *ODS* e no formato *CSV*.

1. Matrizes de confusão

1. Matrizes de confusão relacionadas aos dados em **português**:

https://drive.google.com/file/d/0B_x9Pne58-VTRFpXcGxvaHdJUlE/view?usp=sharing

2. Matrizes de confusão relacionadas aos dados em **inglês**:

https://drive.google.com/file/d/0B_x9Pne58-VTZ2NEV29ZUHdfeEU/view?usp=sharing

3. Matrizes de confusão relacionadas aos dados em **espanhol**:

https://drive.google.com/file/d/0B_x9Pne58-VTMjBvR0g4WTZ4aDA/view?usp=sharing

2. Tabelas PRF

https://drive.google.com/file/d/0B_x9Pne58-VTdUdUb1FNd2diQWc/view?usp=sharing

ANEXO A – Códigos em *Python* utilizados como base

1. "*Hello World*" de todos os métodos utilizados neste relatório, sendo que alguns deles utilizam a biblioteca *Gensim*:

https://github.com/RaRe-Technologies/movie-plots-by-genre

2. Algoritmo *Word Mover's Distance* em *Python*:

http://vene.ro/blog/word-movers-distance-in-python.html