Análise do histórico de revisões dos artigos do *Wikipedia* usando Análise de Grupos

Trabalho de Conclusão de Curso

Engenharia da Computação

Alexandre Gustavo Valença de Azevedo Filho
Orientador: Prof. Dr. Mêuser Jorge Silva Valença
Co-Orientador: Prof. Dr. Ethan V. Munson

Universidade de Pernambuco
Escola Politécnica de Pernambuco
Graduação em Engenharia de Computação

Alexandre Gustavo Valença de Azevedo Filho

Análise do histórico de revisões dos artigos do *Wikipedia* usando Análise de Grupos

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia de Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

Recife, 27 de novembro de 2014.

Escola Politécnica de Pernambuco
Graduação em Engenharia de Computação
Projeto de Final de Curso

POLI
ESCOLA POLITÉCNICA
DE PERNAMBUCO

MONOGRAFIA DE FINAL DE CURSO

Avaliação Final (para o presidente da banca)*

No dia 9 de 12 de 2014, às 8:00 horas, reuniu-se para deliberar a defesa da monografia de conclusão de curso do discente ALEXANDRE GUSTAVO VALENCA DE AZEVEDO FILHO, orientado pelo professor Mêuser Jorge Silva Valença, sob título Análise do histórico de revisões dos artigos do Wikipédia usando Análise de Grupos, a banca composta pelos professores:

Sérgio Campello Oliveira

Mêuser Jorge Silva Valença

Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

☒ Aprovada ☐ Aprovada com Restrições* ☐ Reprovada

e foi-lhe atribuída nota: 10,0 (Dez)

*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O discente terá 07 dias para entrega da versão final da monografia a contar da data deste documento.

SÉRGIO CAMPELLO OLIVEIRA

MÊUSER JORGE SILVA VALENÇA

* Este documento deverá ser encadernado juntamente com a monografia em versão final.

Este trabalho é dedicado à mémoria do meu querido avô, Antônio Carlos de Azevedo. Ele que sempre apoiou todas as minhas decisões e colaborou no meu desenvolvimento, com seu caráter, simplicidade e alegria.

Agradecimentos

Agradeço, primeiramente, a todos os que colaboraram para minha formação acadêmica. Os que contribuíram com meu desenvolvimento como estudante, como profissional e como pessoa.

Agradeço à minha família por darem todo suporte necessário: meus pais, Alexandre Gustavo Valença de Azevedo e Ana Paula Dias Machado de Azevedo, e meus irmãos Henrique Dias Machado de Azevedo, Arthur Dias Machado de Azevedo e Luana Dias Machado de Azevedo.

Agradeço ao meu orientador Prof. Dr. Mêuser Jorge Silva Valença e co-orientador Prof. Dr. Ethan V. Munson. Eles que contribuíram auxiliando, corrigindo e aconselhando para o desenvolvimento deste trabalho. Agradeço também pela disponibilidade oferecida para iniciarmos e concluirmos este trabalho.

Agradeço, por fim, mas não menos importante, à minha namorada, Letícia Costa, pelo seu apoio e que me acompanhou durante todo o desenvolvimento deste trabalho.

Resumo

O aumento no número de citações das páginas do *Wikipedia* e de sua confiabilidade incentiva uma maior investigação das informações presentes neste site. Os dados do histórico de revisões do *Wikipedia* representam as mudanças feitas em cada artigo, e inclui informações dos autores que fizeram a mudança, o tamanho do arquivo, dentre outras. Através do uso da Análise de Grupos, uma sub-área da Mineração de Dados, este trabalho visa analisar os dados do histórico de revisões do *Wikipedia* com foco na tentativa de classificação das páginas e identificação de padrões interessantes. Foram utilizados os algoritmos K-modes, PAM e CLARA aliados a métodos de pré-processamento. Os resultados foram modestos, *F1-score* de 0,5738 foi o melhor resultado para classificação e não foram encontrados padrões satisfatórios. Entretanto, a análise identifica algumas características que podem ser aperfeiçoadas em trabalhos futuros.

Palavras-chave: Análise do *Wikipedia*, Análise de Grupos, Mineração de dados.

Abstract

The increase in the number of citations of *Wikipedia* pages and in their reliability encourages further investigation of the information contained in this website. The *Wikipedia*'s revision history data represent the changes made to the document of each article, and includes information of the authors who made the change, the file size, among others. Through the use of Group Analysis, a sub-field of Data Mining, this work analyzes the historical data of *Wikipedia* revisions. Attempts to the pages classification and identification of interesting patterns tasks. K-modes, PAM and CLARA algorithms were used together with some preprocessing methods. The results were modest, F1-score of 0.5738 was the best result for the classification task and were not found satisfactory patterns. However, the analysis identifies some characteristics that can be improved in future works.

Keywords: *Wikipedia*'s analysis, Group analysis, Data Mining.

Sumário

Índice de Figuras

Índice de Tabelas

Índice de Algoritmos

Lista de Abreviaturas e Siglas

BMJ	British Medical Journal
CC-by-SA	Creative Commons Attribution-ShareAlike
CLARA	Clustering Large Applications
FLOSS	Free/Libre/open Source Software
GFDL	GNU Free Documentation License
KDD	Knowledge Discovery in Databases
LOF	Local Outlier Factor
OOB	Out-of-bag
PAM	Particioning Around Medoids
RF	Random Forest
SGBD	Sistema Gerenciador de Bancos de Dados
SSE	Sum of Squared Errors
URL	Uniform Resource Locator
XML	Extensible Markup Language

1 Introdução

1.1 Motivação

A *Wikipedia* multilíngue como plataforma de produção de conhecimento digital, colaborativa e flexível, pode ser vista como o exemplo mais visível e bem sucedido da migração dos princípios do FLOSS (*free/Libre/open Source Software*) para a cultura predominante. Desde que foi criado em 2001, a *Wikipedia* vem crescendo rapidamente e é hoje uma das maiores fontes de conhecimento dinâmico do mundo. De acordo com o próprio *Wikipedia*, a sua enciclopédia livre está sendo construída por mais de 76 mil colaboradores ativos e possui mais de 30 milhões de artigos em 285 línguas diferentes (*Wikipedia*, 2014).

Por ser uma enciclopédia livre online, em que qualquer internauta pode editar o conteúdo de quase todos os artigos, muito ainda se questiona sobre a confiabilidade das páginas do *Wikipedia*. A verdade, entretanto, é que a *Wikipedia* conseguiu, superar seus desafios de exatidão, anonimidade, confiabilidade e vandalismo para se tornar uma plataforma que provê boa qualidade e confiabilidade aos seus leitores.

Uma pesquisa canadense recente, publicada no *British Medical Journal* (BMJ), descobriu que milhares de artigos cientifícos em revistas médicas têm citado a *Wikipedia* nos últimos anos e que os números de referências estão aumentando rapidamente. Junto com o aumento de número de citações, outro indicador de que a *Wikipedia* está ganhando respeitabilidade é a sua citação por estudiosos de renome (BOULD, 2014). O gráfico que indica o aumento no número de citações é demonstrado na Figura 1.

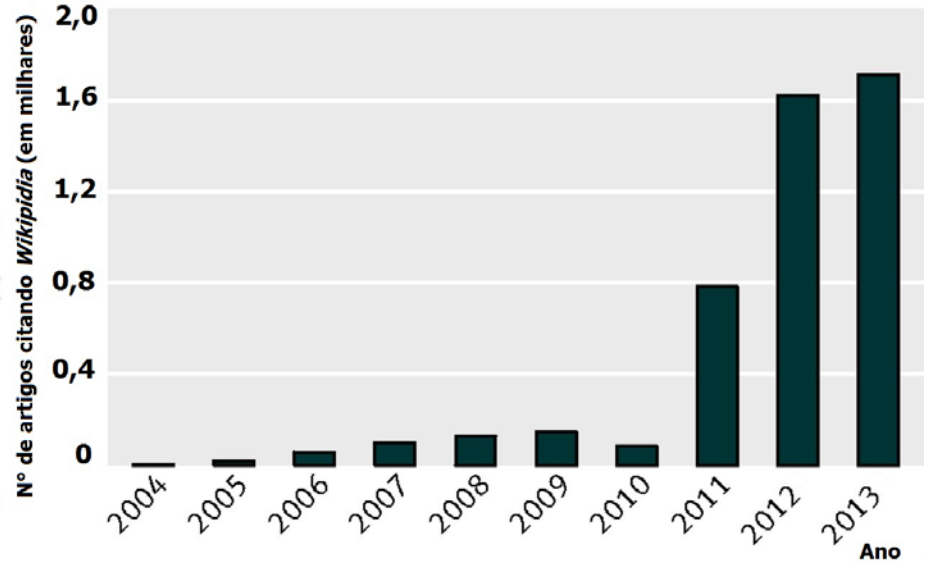

Figura 1. Número de artigos citados no *Wikipedia* por ano.

[Fonte: adaptado de (BOULD, 2014).]

Com base no crescimento tanto do número de artigos do *Wikipedia*, como também da sua confiabilidade, foi identificado um potencial de exploração dos dados do *Wikipedia*. Dustin Maass(2013), em sua tese de mestrado, identificou que o *Wikipedia* foi projetado com um sistema de categorização, porém muitas páginas ainda estão não classificadas ou mal categorizadas, e que devido ao tamanho da base de conhecimento, a categorização manual e a validação das categorizações existentes nas páginas do *Wikipedia* são inviáveis. Dustin Maass então propôs uma análise do histórico de revisões do *Wikipedia* visando o desenvolvimento futuro do sistema de categorização automática das páginas do site. Maass utilizou uma análise simples do algoritmo K-means para análise de grupos, no entanto,não obteve resultados satisfatórios quanto à classificação das páginas baseado nos grupos encontrados (MAASS, 2013).

Dado o potencial de análise dos dados e o crescimento da confiabilidade do *Wikipedia*, o trabalho exposto propõe-se a tentar melhorar a tese desenvolvida por Dustin Maass, testando, em uma nova análise, diferentes algoritmos de Mineração de Dados na busca por melhores índices de classificação. Isto será importante, pois a falta de categorização nas páginas do *Wikipedia* implica na falta de identificação e organização de tais páginas, dificultando sua busca e recuperação.

Além disso, este trabalho também visa a análise do histórico de revisões do *Wikipedia*, na tentativa de identificar padrões interessantes que possam ser suficientes ou serem combinados com as informações baseadas em conteúdo (análise de textos) para obtenção de informações úteis.

1.2 Objetivos

1.2.1 Objetivo Geral

Este trabalho visa analisar os dados do histórico de revisões do *Wikipedia* utilizando técnicas e algoritmos de Mineração de Dados, mais especificamente Análise de Grupos. Esta análise irá focar na classificação das páginas e na tentativa de identificar padrões dentro da base de dados do histórico de revisões do *Wikipedia*.

1.2.2 Objetivos Específicos

- Utilizar o *software* Metriki para extração do histórico de revisões de metadados do *Wikipedia* (frequência de mudanças, número de autores ativos, tamanho das mudanças, etc.);
- criar na linguagem R o código para análise de agrupamentos;
- implementar na linguagem Java o código para cálculo dos parâmetros de classificação (*Precision*, *Recall*, *F-score*);
- dividir as análises e resultados em duas abordagens:
    - A primeira com foco principal na classificação seguido da identificação de padrões, onde o número de grupos é exatamente igual ao número de categorias presentes na base de dados;
    - A segunda abordagem tem foco exclusivamente na identificação de padrões e não nos limita a utilizar o número de grupos igual ao número de categorias;
- comparar os resultados obtidos na primeira abordagem com o estudo realizado por Dustin Maass (MAASS, 2013).

1.3 Estrutura da Monografia

Esta monografia está organizada em 5 capítulos. O Capítulo 2 apresenta a fundamentação teórica, destacando a mineração de dados, os três diferentes algoritmos de agrupamento utilizados neste trabalho, além dos métodos de pré-processamento e avaliação da classificação. Em seguida, o Capítulo 3 apresenta as abordagens propostas. No Capítulo 4, os resultados são apresentados. Por fim, o Capítulo 5 traz as conclusões e sugere trabalhos futuros.

2 Fundamentação Teórica

O referencial teórico deste trabalho aborda informações sobre o caso de estudo em questão, que é o *Wikipedia* e do *software* Metriki e seu funcionamento, pois ele será utilizado para extração das bases de dados. Além disso, abordaremos a Mineração de Dados e sua subárea de Análise de Grupos, sem esquecer de destacar os métodos de pré-processamento que serão utilizados e as métricas de avaliação.

2.1 Wikipedia

Wikipedia é um projeto de enciclopédia multilíngue de licença livre, baseado na *web*, escrito de maneira colaborativa e que encontra-se atualmente sob administração da Fundação Wikimedia. A *Wikipedia* foi lançada em 15 de janeiro de 2001 por Jimmy Wales e Larry Sanger e tornou-se a maior e mais popular obra de referência geral na internet com cerca de 365 milhões de leitores (*Wikipedia*, 2014).

Todo o texto contido na *Wikipedia* era coberto pela *GNU Free Documentation License* (GFDL), uma licença *copyleft* que permite a redistribuição, a criação de obras derivadas e o uso comercial do conteúdo preservando os direitos autorais, até junho de 2009, quando foi adotada a licença *Creative Commons Attribution-ShareAlike* (CC-by-SA) 3.0., que é escrita de forma mais genérica e mais conhecida que a GFDL (*Wikipedia*, 2014).

Afastando-se do estilo das enciclopédias tradicionais, a *Wikipedia* emprega um sistema aberto. Exceto por algumas páginas particularmente propensas ao vandalismo, cada artigo pode ser editado de forma anônima ou com uma conta de usuário. Os conteúdos são acordados por consenso, nenhum artigo ou o seu conteúdo é propriedade do seu criador ou de qualquer outro editor, ou é avaliado por qualquer autoridade reconhecida.

Por padrão, qualquer edição em um artigo se torna disponível imediatamente, antes de qualquer revisão. Isto significa que um artigo pode conter erros, contribuições equivocadas, defesa de algo, ou mesmo absurdos evidentes, até que outro editor corrija o problema.

Contribuintes, registrados ou não, podem tirar proveito dos recursos disponíveis no *software* que opera a *Wikipedia*. A página *Ver histórico* que acompanha cada artigo registra toda e qualquer versão anterior do verbete. Apesar disso, uma revisão com conteúdo calunioso, ameaças criminosas ou violações de direitos autorais podem ser removidas mais tarde. Esta característica possibilita uma maior facilidade para comparar antigas e novas versões do artigo, desfazer alterações que um editor considera indesejáveis, ou restaurar um conteúdo perdido. Todos esses dados formam o histórico de revisões dos artigos da *Wikipedia*.

A página *Discussão*, associada a cada artigo, é utilizada para coordenar o trabalho entre vários editores. Editores regulares muitas vezes mantêm uma lista de *páginas vigiadas* de artigos de interesse para eles, de modo que eles podem facilmente manter o controle sobre todas as alterações recentes nessas páginas.

Programas de computador chamados robôs têm sido amplamente utilizados para remover vandalismos logo que eles são feitos, para corrigir erros comuns e questões estilísticas, ou para iniciar artigos, tais como entradas de geografia em um formato padrão a partir de dados estatísticos (*Wikipedia*, 2014).

2.2 Software Metriki

O *software* Metriki, originalmente desenvolvido por Peine (PEINE, 2012) em linguagem Java, foi desenvolvido na *University of Wisconsin – Milwaukee*, nos Estados Unidos e tem como objetivo a extração de dados do *Wikipedia*, mais especificamente, os metadados do histórico de revisões. Metadados podem ser definidos como "dados que descrevem os dados", ou seja, são informações úteis para identificar, localizar, compreender e gerenciar os dados.

O *software* acessa o histórico de revisões do *Wikipedia* enviando o requerimento para o servidor via URL (*Uniform Resource Locator*). O servidor então retorna um arquivo XML contendo todas as informações que foram requisitadas. Essas informações, então, são armazenadas no banco de dados. É importante destacar que as requisições são feitas através do nome do artigo o qual deseja-se obter as revisões históricas. Além disso, as requisições são feitas uma por uma, o que dependendo do tamanho da lista de artigos, torna a execução do Metriki bastante demorada.

O Metriki foi melhorado por Shah (SHAH, 2012) e esta expansão proporcionou uma estrutura melhor para a base de dados. Shah também executou algumas análises preliminares com pequenas amostras de dados e que trouxeram ainda mais interesse para análise dos metadados do *Wikipedia*.

2.3 Mineração de Dados: Análise de Grupos

A mineração de dados é o processo de descoberta automática de informações úteis em grandes depósitos de dados. As técnicas de mineração de dados são organizadas para agir sobre grandes bancos de dados com o intuito de descobrir padrões úteis e recentes que poderiam, de outra forma, permanecer ignorados (TAN; STEINBACH; KUMAR, 2009).

É importante destacar que nem todas as tarefas de descoberta de informação são consideradas mineração de dados. Por exemplo, a procura de registros individuais usando um sistema gerenciador de bancos de dados (SGBD) ou a busca de determinadas páginas da Web por meio de uma consulta em um mecanismo de busca na Internet são tarefas relacionadas à área de recuperação de dados (TAN; STEINBACH; KUMAR, 2009).

A mineração de dados é uma parte fundamental na descoberta de conhecimento em bancos de dados (KDD – *Knowledge Discovery in Databases*), que é o processo geral de conversão de dados brutos em informações úteis, conforme mostrado na Figura 2.

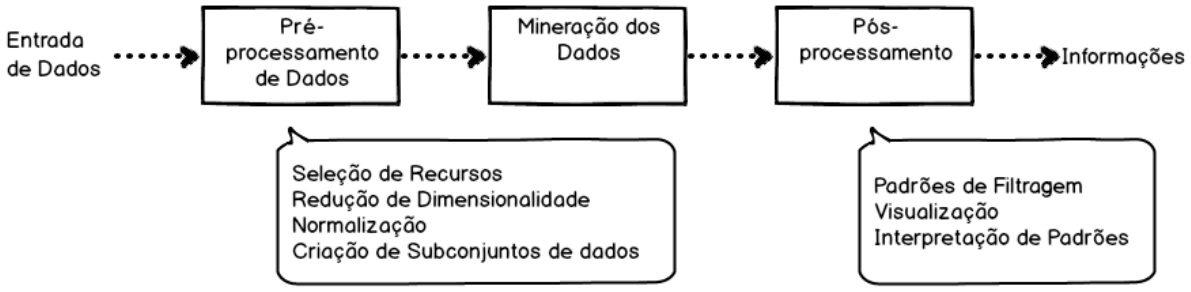

Figura 2. O processo de descoberta de conhecimentos em bancos de dados (KDD).
[Fonte: extraído de (TAN; STEINBACH; KUMAR, 2009)]

A mineração de dados apresenta quatro tarefas centrais: Modelagem Previsiva, Análise de Associação, Detecção de Anomalias (*outliers*) e Análise de Grupos. A modelagem de previsão se refere à tarefa de construir um modelo para a variável alvo como uma função das variáveis explicativas. Há dois tipos de modelagem de previsão: classificação, a qual é usada para variáveis alvo discretas, e regressão, que é usada para variáveis alvo contínuas. Por exemplo, prever se um usuário Web fará uma compra em uma livraria *online* é uma tarefa de classificação, porque a variável é uma tarefa de valor binário. Por outro lado, prever o preço futuro de uma ação é uma tarefa de regressão, porque o preço é um atributo de valor contínuo. O objetivo de ambas as tarefas é obter um modelo que minimize o erro entre os valores previsto e real da variável alvo (TAN; STEINBACH; KUMAR, 2009).

A análise de associação é usada para descobrir padrões que descrevam características altamente integradas entre os dados. Os padrões descobertos são normalmente representados na forma de regras de implicação ou subconjuntos de características. Aplicações úteis de análise de associação incluem a descoberta de genes que possuam funcionalidade associada, ou a identificação de páginas *web* que sejam acessadas juntas.

A detecção de anomalias é a tarefa de encontrar observações cujas características sejam significativamente diferentes do resto dos dados. O objetivo de um algoritmo de detecção de anomalias é descobrir as anomalias verdadeiras e evitar rotular erroneamente objetos normais como anômalos. Exemplos de aplicações seriam a detecção de fraudes, padrões incomuns de doenças e perturbações no meio ambiente.

A análise de grupos procura agrupar objetos baseada apenas em informações encontradas nos dados que descrevem os objetos e seus relacionamentos. O objetivo é que os objetos dentro de um grupo sejam semelhantes (ou relacionados) entre si e diferentes de (ou não relacionados aos) outros objetos de outros grupos. Quanto maior a semelhança (ou homegeneidade) dentro de um grupo e maior a diferença entre grupos, melhor ou mais distinto será o agrupamento (TAN; STEINBACH; KUMAR, 2009).

A análise de grupos está relacionada a outras técnicas que são usadas para dividir objetos de dados em grupos. Por exemplo, o agrupamento pode ser

considerado uma forma de classificação pelo fato de criar uma rotulagem de objetos como rótulos de classes. Entretanto, ela deriva estes rótulos apenas dos dados. Por esse motivo, a análise de grupos é às vezes chamada de classificação não supervisionada.

Existem dois tipos básicos de análise de grupos: agrupamento hierárquico e o agrupamento particional. O agrupamento particional é simplesmente uma divisão do conjunto de dados em subconjuntos (grupos) não interseccionados de modo que cada objeto esteja exatamente em um subconjunto. Nos casos em que os grupos possuem subgrupos, então obtemos um agrupamento hierárquico, que é um conjunto de grupos alinhados organizados como uma árvore. Cada nó (grupo) na árvore (exceto pelos nó folha) é a união dos seus filhos (subgrupos) e a raiz da árvore é o grupo contendo todos os objetos (TAN; STEINBACH; KUMAR, 2009).

2.4 Pré-processamento

O propósito do pré-processamento é transformar os dados de entrada brutos em um formato apropriado para análises subsequentes. Os passos envolvidos no pré-processamento de dados incluem a fusão de dados de múltiplas fontes, a limpeza dos dados para remoção de ruídos, observações duplicadas, a seleção de registros e características que sejam relevantes à tarefa de mineração dados. Devido as muitas formas através das quais os dados podem ser coletados e armazenados, o pré-processamento de dados talvez seja a etapa mais importante e trabalhosa neste processo (TAN; STEINBACH; KUMAR, 2009).

2.4.1 Seleção de atributos – *Random Forest*

O objetivo da seleção de atributos é escolher um subconjundo de atributos (ou variáveis) ou criar novos atributos que substituam um conjunto deles, reduzindo assim a dimensão do banco de dados. Assim, reduz-se a complexidade do banco de dados e consequentemente o tempo de processamento para extrair dele algum conhecimento. Além disso, utilizando a seleção de atributos é possível remover atributos desnecessários que podem causar ruído no resultado final da análise.

O *Random Forest (*RF*)* é um método que utiliza um algoritmo bastante polular e eficiente baseado na ideia de modelo de agregação para problemas de classificação

e regressão, introduzidos por Breiman (BREIMAN, 2001). O princípio do *RF* é combinar várias árvoves binárias de decisão, em que cada árvore é gerada baseada nos valores de um conjunto independente de vetores aleatórios. Os vetores aleatórios são gerados a partir de uma distribuição de probabilidades fixa.

Breiman esboçou o bom desempenho do *Random Forest* relacionando à qualidade de cada árvore juntamente com a pequena correlação entre as árvores da floresta. A correlação entre as árvores é definida como correlação ordinária de previsões, conhecida como amostras *out-of-bag* (OOB). A amostra OOB, que é um conjunto de observações que não foram utilizadas para contruir a árvore atual, é usada para estimar o erro de previsão e assim avaliar a importância das variáveis.

Robin Genuer propôs uma metodologia para seleção de variáveis usando o *Random Forest* (GENUER; POGGI; TULEAU-MALOT, 2010). Essa metodologia tem dois objetivos: encontrar variáveis importantes com grande relação a variável resposta para problemas de interpretação e encontrar o menor número de variáveis suficientes para uma boa previsão da variável resposta. Assim, foi proposto por Genuer o seguinte procedimento de duas etapas, em que a primeira é comum e a segunda depende do objetivo:

- Eliminação e *raking* preliminar:
  - computar os valores de importância do *Random Forest*, cancelar a variável de menor importância;
  - ordenar as *m* variáveis restantes em ordem decrescente de importância;
- Seleção de Variáveis:
  - para interpretação - construir a coleção mais próxima dos modelos do *Random Forest* envolvendo as *k* primeiras variáveis, para *k=1* à *m* e selecionar as variáveis envolvidas no modelo principal para o de menor erro OOB;
  - para previsão - começar pelas variáveis ordenadas guardadas para interpretação, construir uma sequência ascendente de modelos do *Random Forest*, testando as variáveis passo a passo. As variáveis do último modelo são selecionadas.

2.4.2 Detecção e Remoção de Anomalias (*Outliers*)

O objetivo na detecção de anomalias é encontrar objetos que sejam diferentes da maioria dos outros objetos. Muitas vezes, objetos anômalos são conhecidos como fatores elementos estranhos, já que em um desenho disperso dos dados, eles ficam longe dos outros pontos de dados. A detecção de anomalias também é conhecida como detecção de desvios, porque objetos anômalos têm atributos que se desviam significativamente dos valores de atributos esperados ou típicos ou, como mineração de exceções, porque as anomalias são excepcionais em algum sentido (TAN; STEINBACH; KUMAR, 2009).

Existe uma diversidade de abordagens de detecção de anomalias de diversas áreas, incluindo estatística, aprendizado de máquina e mineração de dados. Todas tentam capturar a ideia de que um objeto de dados anômalo é diferente ou de alguma forma inconsistente com outros objetos. Neste projeto, a existência de dados extremos pode inteferir diretamente nos grupos resultantes pelo algoritmo de agrupamento, como por exemplo, na média dos atributos de cada grupo.

O algoritmo LOF (*Local Outlier Factor*) é utilizado para identificar anomalias locais baseadas em densidade (BREUNIG; KRIEGEL; NG; SANDER, 2000). Utilizando LOF, a densidade local de um ponto é comparado com a densidade de seus vizinhos. Se a primeira é significativamente menor do que o segundo (com um valor de LOF maior do que um), é o ponto em uma região espaçada em relação aos seus vizinhos, o que sugere que ele seja um anômalo.

Devido à abordagem local, LOF é capaz de identificar os *outliers* em um conjunto de dados que não seriam discrepantes em outra área do conjunto de dados. Por exemplo, um ponto a uma distância pequena para um conjunto muito denso é um *outlier*, enquanto um ponto dentro de um *cluster* escasso pode apresentar distâncias semelhantes a seus vizinhos.

Enquanto a intuição geométrica de LOF é apenas aplicável a espaços de vetores de baixa dimensão, o algoritmo pode ser aplicado em qualquer contexto em que uma função de dissimilaridade pode ser definida. Foi experimentalmente demonstrado que funciona muito bem em numerosas configurações, muitas vezes superando os concorrentes, por exemplo, na detecção de intrusão (LAZAREVIC; OZGUR; ERTOZ; SRIVASTAVA; KUMAR, 2003).

2.4.3 Padronização e Normalização

A normalização ou padronização é um tipo comum de transformação de variáveis. Na comunidade de mineração de dados, os termos são muitas vezes usados intercambiavelmente. Em estatística, entretanto, o termo normalização pode ser confundido com a transformação usada para tornar uma variável normal, isto é, Gaussiana (TAN; STEINBACH; KUMAR, 2009).

A normalização ou padronização dos dados é um processo em que os atributos presentes na base de dados são organizados para aumentar a coesão dos tipos de entidade. Em outras palavras, o objetivo da normalização ou padronização dos dados é reduzir e até eliminar a redundância dos dados.

Um exemplo tradicional é o da “padronização de uma variável”. Se *m* for a média dos valores do atributo e *d* seu desvio padrão, então a transformação dada pela Equação (2.1) cria uma nova variável *n1* que tem uma média de zero e um desvio padrão de um,

$$n1 = \frac{x-m}{d}. \quad (2.1)$$

2.5 Algoritmos de agrupamento

Técnicas de agrupamento baseadas em protótipos criam um particionamento de um nível dos objetos de dados. Há um número de tais técnicas, mas duas das proeminentes são K-means e K-medoid. K-means define um protótipo em termos de um centroide, que é geralmente a média de um grupo de pontos, e uma das técnicas mais famosas por sua eficiência e simplicidade. K-medoid define um protótipo em termos de um **medóide (*medoid*)**, que é o ponto mais representativo para um grupo de pontos. Além destas duas técnicas mais conhecidas, a técnica K-modes, uma variação do K-means para tratar dados categóricos, foi testada neste trabalho.

Nesta seção serão descritas diferentes técnicas de agrupamento particionais.

2.5.1 K-means

É um dos métodos mais conhecidos e utilizados, além de ser o que possui o maior número de variações. Criado por MacQueen (MACQUEEN, 1967), objetiva-se

particionar n observações dentre k grupos em que cada observação pertence ao grupo mais próximo da média, o que resulta em uma divisão do espaço de dados.

O algoritmo básico do *K-means*, formalmente descrito pelo Algoritmo 2.1, primeiro escolhe K centroides iniciais, onde K é um parâmetro especificado pelo usuário, a saber, o número de grupos desejado. Cada ponto é atribuído a seguir ao centroide mais próximo, e cada coleção de pontos atribuídos a um centroide é um grupo. O centroide de cada grupo é então atualizado baseado nos pontos atribuídos ao grupo. Repetimos os passos de atribuição e atualização até que nenhum ponto mude de grupo ou, equivalentemente, até que os centroides permaneçam os mesmos (TAN; STEINBACH; KUMAR, 2009).

Algoritmo 1 Algoritmo *K-means* básico

1: Selecione K pontos como centroides iniciais.
2: **Repita**
3: Forme K grupos atribuindo cada ponto ao seu centroide mais próximo.
4: Recalcule o centroide de cada grupo/
5: **Até que** os centroides não mudem.

Algoritmo 1. Algoritmo *K-means* Básico

[Fonte: extraído de (TAN; STEINBACH; KUMAR, 2009).]

K-means é simples e pode ser usado para uma ampla variedade de tipos de dados. Também é bastante eficiente, embora múltiplas execuções sejam executadas com frequência. Por outro lado, ele não pode lidar com grupos não globulares ou de tamanhos e densidades diferentes. Além disso, o algoritmo é bastante sensível a ruídos e *outliers*.

2.5.2 K-modes

A grande maioria dos algoritmos de agrupamento foca em conjunto de dados numéricos. Entretanto, grande parte dos dados existentes em um conjunto de dados é categórico, em que os valores dos atributos não podem ser naturalmente ordenados como valores numéricos.

Vários algoritmos de agrupamento para agrupar dados categóricos têm sido relatados. Ralambondrainy (1995) apresentou uma abordagem que converte atributos categóricos múltiplos em atributos binários, usando zero (0) ou um (1) para

representar ou uma categoria presente ou uma categoria ausente, e trata os atributos binários como numéricos no algoritmo *K-means*. Huang (1998) propôs *K-modes clustering* que extende o *K-means* para agrupar dados categóricos usando combinações simples da medida de similaridade para os objetos categóricos. Esse método é baseado no *K-means,* mas remove a limitação de que os dados sejam numéricos. A modificação do algoritmo *K-means* para o algoritmo *K-modes* é descrita a seguir (HUANG, 1998).

- Usar uma combinação simples da medida de similaridade para objetos categóricos;
- substituir média dos grupos por *mode;*
- usar método baseado na frequência para atualizar os *modes*.

A combinação simples da medida de similaridade pode ser definida como se segue. Sejam *X* e *Y* dois objetos categóricos descritos por *m* atributos categóricos. A medida de similaridade entre X e Y pode ser definida pelo total de incompatibilidade (*mismatch*) correspondente as atributos categóricos dos dois objetos. Quanto menor o número de incompatibilidade, mais similar serão os dois objetos. Matematicamente, isso pode ser representado pelas equação (2.2) (GOWDA;DIDAY,1991):

$$d(X,Y) = \sum_{j=1}^{m} \delta(x_j, y_j) \tag{2.2}$$

onde,

$$\delta(x_j, y_j) = \begin{cases} 0 \; se \; x_j = y_j \\ 1 \; se \; x_j \neq y_{j.} \end{cases} \tag{2.3}$$

Quando a Equação (2.2) é usada como medida de similaridade para objetos categóricos, a função de custo se torna:

$$p(W,Q) = \sum_{l=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,l} \delta(x_{i,j}, y_{l,j}) \tag{2.4}$$

Onde, $w_{i,j} \in W$ e $Q_l = [q_{l,1}, q_{l,2}, \dots, q_{l,m}] \in Q$

O algoritmo *K-modes* minimiza a função de custo definida na Equação (2.4) e consiste nos seguintes passos (HUANG,1998):

1) Seleção de K inicial *modes*, um para cada grupo (*cluster*);
2) alocação dos objetos no grupo cujo *mode* é o mais próximo a ele, de acordo com a Equação (2.2);

3) depois de todos os objetos serem alocados nos grupos, reteste a similaridade dos objetos contra os *modes* atuais. Se um objeto é encontrado de tal forma que seu mais próximo *mode* pertence a outro grupo em vez de seu atual, realoca-se o objeto para seu novo grupo e atualiza-se os *modes* de ambos os conjuntos.
4) repetir o passo anterior até nenhum objeto ter mudança de grupo depois de um ciclo completo de testes de todo conjunto de dado.

2.5.3 K-medoid

K-medoid define um protótipo em termos de um medóide (*medoid*), que é o ponto mais representativo para um grupo de pontos e pode ser aplicado a uma ampla faixa de dados, já que requer apenas uma medida de proximidade para um par de objetos (TAN; STEINBACH; KUMAR, 2009).

O método *K-Medoids* foi introduzido por Kaufman e Rousseeuw (KAUFMAN; ROUSSEEUW, 1990) e não é sensível a anômalos (*outliers*) como o *K-means*. Neste método de agrupamento, cada grupo é representado pelo objeto mais centralizado conhecido como medóide. O agrupamento *K-medoid* criado por Kaufman e Rousseeuw é o algoritmo PAM (*Particioning Around Medoids*) que é descrito abaixo:

1) Escolher aleatoriamente K objetos como medóides iniciais;
2) atribuir, para cada um dos objetos restantes, um grupo que tenha o medóide mais próxima;
3) em um grupo, selecionar aleatoriamente os objetos “não medóide”, que são chamados $O_{nonmedoid}$;
4) computar o custo de trocar o medóide com $O_{nonmedoid}$. Esse custo é a diferença no erro quadrado se a medóide atual for trocada pelo $O_{nonmedoid}$. Se o custo for negativo, tornar $O_{nonmedoid}$ a medóide do grupo. O erro quadrado é novamente o somatório dos erros de todos os objetos no conjuto de dados:

$$E = \sum_{i=1}^{k} \sum_{o \in C_i} |o - O_{medoid(i)}|^2 \quad (2.5)$$

onde, $O_{medoid(i)}$ é a medóide do i^{th} grupo;
5) repetir a partir do passo 2 até não haver mais mudanças;

A complexidade computacional do algoritmo PAM estimulou o desenvolvimento do CLARA (*Clustering Large Applications*), um algoritmo *K-medoid* baseado em amostras (*sampling*). CLARA utiliza várias amostras do conjunto de dados e usa o PAM para cada uma delas. Portanto, CLARA escolhe os grupos obtidos na execução que deram o menor valor para função objetiva e atribui cada objeto dos dados para a correspondente medóide.

2.6 Avaliação: métricas para classificação

A eficiência da análise de agrupamentos voltada para classificação é comumente avaliada por três índices diferentes: *Precision*, *Recall* e *F1-score*. Essas medidas são obtidas baseada nas categorias da matriz de confusão: *True Positive* (TP), *False Negative* (FN), *False Positive* (FP), *True Negative* (TN). O cálculo de cada categoria é dado da seguinte forma:

- *True Positive* (TP) – corresponde ao número de exemplos positivos previstos corretamente pelo modelo de classificação.
- *False Negative* (FN) – corresponde ao número de exemplos positivos previstos erroneamente como negativos pelo modelo de classificação.
- *False Positive* (FP) – corresponde ao número de exemplos negativos previstos erroneamente como positivos pelo modelo de classificação.
- *True Negative* (TN) – corresponde ao número de exemplos negativos previstos corretamente pelo modelo de classificação.

A matriz de confusão que resume o número de instâncias previstas correta ou incorretamente por um modelo de classificação é mostrada na Tabela 1.

Tabela 1. Matriz de confusão para um problema de classificação binária.

[Fonte: elaboração própria.]

		Classe Prevista	
		+	-
Classe Atual	+	TP	FN
	-	FP	TN

Precision e *Recall* são duas métricas amplamente usadas em aplicações onde a detecção bem sucedida de uma das classes é considerada mais significativa do que a de outras classes. Uma definição formal destas métricas é apresentada a seguir.

2.6.1 Precision

A *precision* determina a fração de registros que realmente acabam sendo positivos no grupo que o classificador declarou como classe positiva. Quanto maior a precisão, menor o número de erros positivos falsos cometidos pelo classificador. A p*recision* é calculada pela Equação (2.6),

$$Precision = \frac{TP}{TP+FP}. \quad (2.6)$$

2.6.2 Recall

A *recall* mede a fração de exemplos positivos previstos corretamente pelo classificador. Classificadores com *recall* alta têm muito pouco exemplos positivos mal classificados como a classe negativa. Na verdade, o valor de *recall* é equivalente à taxa de positivos verdadeiros. A *recall* é descrita pela Equação (2.7).

$$Recall = \frac{TP}{TP+FN}. \quad (2.7)$$

2.6.3 F1-score

Precision e *recall* podem ser resumidas em outra métrica conhecida como a *F-score,* que é a média harmônica ponderada dos valores de *precision* e *recall*. A ponderação da *F-score* é usada para focar-se ao rigor da correção (*precision*) ou na integridade (*recall*). Na fórmula *F-score*, o valor de *β* é o coeficiente de ponderação do *precision* sobre o *recall*. Ela é calculada dividindo-se o produto das pontuações de *precision* e *recall* pela soma do *recall* e *precision* ponderada. Este é então multiplicado por um mais o peso ao quadrado,

$$F.score = (1+\beta^2) * \frac{Precision * Recall}{(\beta^2 * Precision)+Recall}. \quad (2.8)$$

Para este trabalho, utilizaremos a *balance-weighted F-score*, conhecido como *F1-Score*. Esta fórmula usa o valor de *β* igual a 1 e é calculada pela Equação (2.9),

$$F1.score = 2 * \frac{Precision * Recall}{Precision+Recall}. \quad (2.9)$$

O valor final é um número entre zero (0) e um (1), e a príncipio, *F1* representa uma média harmônica entre *Recall* e *Precision*. A média harmônica de dois números x e y tende a ficar mais próxima do menor dos números. Logo, um valor alto da *F1-score* assegura que tanto a *precision* quanto a *recall* sejam razoavelmente altas.

3 Metodologia

Nesta seção serão apresentadas todas as metodologias aplicadas para análises do histórico de revisões do *Wikipedia*. É importante destacar que todos os códigos e implementações utilizaram o *software* R. O R é um conjunto integrado de ferramentas computacionais que permitem a manipulação e análise de dados, o cálculo numérico e a produção de gráficos (VENABLES; SMITH, 2008).

3.1 Base de dados

A base de dados utilizada no desenvolvimento deste trabalho segue o mesmo padrão utilizado por Maass em sua tese (MAASS, 2013). O mesmo padrão indica que foram utilizadas as mesmas listas de artigos criadas e utilizadas por Maass, entretanto os dados do histórico de revisões do *Wikipedia* foram extraídos novamente, utilizando o software Metriki desenvolvido por Peine (PEINE, 2012), para obtenção de uma base de dados atualizada. A atualização da base de dados não interfere na comparação das análises visto que os focos são a classificação e a identificação de padrões.

Maass utilizou duas bases de dados. A primeira base, chamada *Data set* #1 e ilustrada na Tabela 2, apresenta pequenos grupos de artigos pertencentes a categorias bastante específicas. Cada categoria é formada por 20 páginas diferentes escolhidas aleatoriamente, cada uma estreitamente ligada com as outras por um tipo muito específico de tema. Um total de 14 grupos foram selecionados (MAASS, 2013). O conjunto de dados proveu 1.089.360 revisões para análise.

Tabela 2. Categorias da *Data Set* #1.

[Fonte: elaboração própria]

Pássaros	Shows de Comédia da Televisão
Batalhas Históricas	Líderes Históricos
Megacidades	Estrelas de Cinema
Rabinos	Cientistas
Teoremas	Vencedores do *World Music Award*
Empresas da *Fortune 500*	Mamíferos
Países do Norte da Europa	*Software People*

A segunda base de dados, chamada *Data Set #2* e ilustrada na Tabela 3, possui um grande número de artigos pertencentes a três categorias amplas e diferentes. Cada categoria de artigos foi composta por entre 1685 e 1707 páginas, com cada grupo representando um tipo de tópico amplo. Um total de 3 grupos foram usados, provendo 912.816 revisões para serem utilizadas na análise.

Tabela 3. Categorias da *Data Set #2*

[Fonte: elaboração própria]

Categorias	Número de Artigos
Mitologia Grega	1.707
Ciência Quântica	1.685
Medalhistas Olímpicos dos EUA	1.690

O primeiro passo depois da extração dos dados foi transformar os dados brutos do histórico de revisões do *Wikipedia* em atributos que tivessem a possibilidade de ter grande variância entre os artigos. Diferentemente da estrutura criada por Maass em sua tese, onde foram criadas 5 tabelas diferentes, a base de dados desta analise é composta por apenas 1 tabela e 12 atributos, estabelecidos para cada artigo (página):

- *RevisionCount, userNameCount*: o somatório total de revisões e de seus respectivos usuários.
- *sizeAvg*: média do tamanho das revisões.
- *TimeHaveBeingModified*: o número de anos em que o artigo vem sendo modificado.
- *NumberOfRevisionsYearPeak, NumberOfUsersYearPeak*: o somatório de revisões e usuários no ano de maior revisão do artigo.
- *Authour_1, Revision_author_1*: autor de maior destaque no artigo e o número total de suas revisões.
- *Authour_2, Revision_author_2*: autor de segundo maior destaque no artigo e o número total de suas revisões.
- *Authour_3, Revision_author_3*: autor de terceiro maior destaque no artigo e o número total de suas revisões.

Além destes, com objetivo de criar atributos que pudessem diferenciar artigos recentes, de artigos antigos, foram criados mais 9 atributos, totalizando 21, e que serão utilizados de acordo com o algoritmo de seleção de atributos, que escolhera os melhores representantes dentre eles.

- *NumberOfRevisions_since2013, NumberOfUsers_since2013*: o somatório total de revisões e de seus respectivos usuários a partir do ano de 2013.
- *sizeAvg_since2013*: média do tamanho das revisões a partir de 2013.
- *NumberOfRevisions_since2012, NumberOfUsers_since2012*: o somatório total de revisões e de seus respectivos usuários a partir do ano de 2012.
- *sizeAvg_since2012*: média do tamanho das revisões a partir de 2012.
- *NumberOfRevisions_since2011, NumberOfUsers_since2011*: o somatório total de revisões e de seus respectivos usuários a partir do ano de 2011.
- *sizeAvg_since2011*: média do tamanho das revisões a partir de 2011.

3.1.1 Pré-processamento dos dados

Para realizar a análise de mineração de dados, faz-se necessário uma série de processos, de forma que se obtenha uma base de dados melhor preparada para análise. Toda parte de pré-processamento foi desenvolvida utilizando o *software R-project* que possui funções de algoritmos já prontas para a análise.

Antes de utilizar os algoritmos de pré-processamento, foram retiradas todas as revisões feitas por ‘robôs’. Os programas de computador chamados robôs têm sido amplamente utilizados para remover vandalismos logo que eles são feitos, para corrigir erros comuns e questões estilísticas, ou para iniciar artigos, tais como entradas de geografia em um formato padrão a partir de dados estatísticos. Pelo fato de não serem usuários reais, as revisões feitas pelos robôs não interessam e não serão utilizadas nesta análise.

O primeiro processo utilizado foi o de seleção de atributos usando o algoritmo *Random Forest*, que visa escolher um subconjunto de atributos que melhor representasse a base de dados, removendo atributos desnecessários que poderiam causar ruído no resultado final da análise. É importante ressaltar que somente os atributos numéricos foram utilizados como entrada para esta etapa. Além disso, foram

utilizadas 5000 árvores na primeira floresta, 2000 árvores nas florestas adicionais das interações e uma fração de 0,2 variáveis excluídas a cada interação. A Figura 3 ilustra a saída do algoritmo *Random Forest* para seleção de atributos.

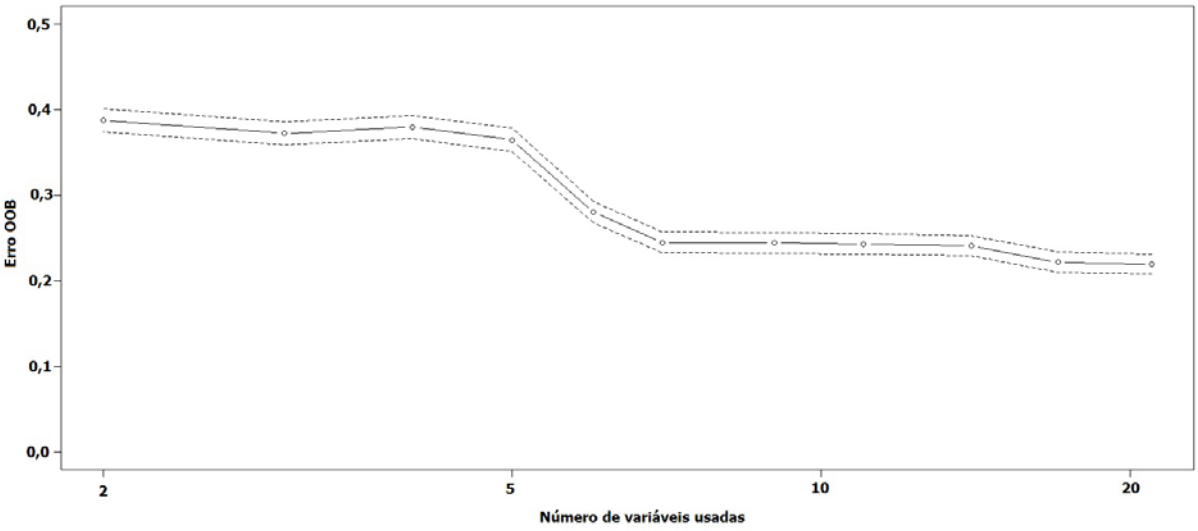

Figura 3. Erro OOB x Número de variáveis usadas

[Fonte: elaboração própria.]

O número de atributos foi escolhido de acordo com o menor erro OOB e os atributos eram escolhidos baseado na importância calculada; neste caso, onde o resultado foi extraído do *Data Set* #2, o número de variáveis escolhidas foi 17, com uma pequena diferença para o OOB com 21 atributos.

Em seguida, foi aplicado o algoritmo LOF para detecção e remoção de anômalos (*outliers*), que visa encontrar objetos que sejam diferentes da maioria dos outros objetos, pois esses objetos também podem causar ruídos e desbalancear os valores finais da análise, como por exemplo a média dos grupos. Foi utilizado um número de vizinhos para o cálculo dos fatores, já que, no algoritmo LOF a densidade local de um ponto é comparado com a densidade de seus vizinhos. Além disso, o algoritmo não estabelece um número exato de *outliers*, o resultado é um vetor com todos os fatores calculados.

Por fim, foi aplicada a etapa de padronização ou normalização dos dados que visa reduzir e até eliminar a redundância dos dados. O tipo de padronização ou normalização utilizado foi o de padronização simples de uma variável, que é a

diferença entre o valor do atributo e a média dos valores do atributo, dividido pelo seu desvio padrão como foi mostrado na Equação (2.1). Se variáveis diferentes devem ser combinadas de alguma forma, como é caso do nosso estudo, então tal transformação é muito importante para evitar se ter uma variável com valores grandes dominando os resultados do cálculo.

Finalizados os três processos, a base de dados está pronta para ser colocada na análise de agrupamentos.

3.2 Experimentos

Como descrito nos objetivos deste projeto, as análises seguiriam duas abordagens diferentes. A primeira, com foco na classificação, em que o número de grupos é exatamente igual ao número de categorias presentes na base de dados. Por exemplo, a base *Data Set* #2 possui 3 categorias, logo a análise de agrupamentos será feita utilizando o número de agrupamentos igual a 3. Os resultados desta abordagem são colocados junto às categorias originais de cada artigo para que seja construída a matriz de confusão. Assim, é possível calcular os índices de *Precision*, *Recall* e *F1-score*. Essas medidas são extremamente importantes para qualificar a previsão de classificação dada pelos algoritmos de agrupamento.

A segunda abordagem tem foco na identificação de padrões e não se limita a utilizar o número de grupos igual ao número de categorias. Neste caso a ideia é ter um número maior de grupos, na tentativa de extrair um número maior de características. Por isso, é utilizado o desenho da soma do erro quadrado (SSE - *Sum of Squared Errors*) do algoritmo versus o número de grupos, que é ilustrado na Figura 4, para auxiliar na escolha do número de grupos .

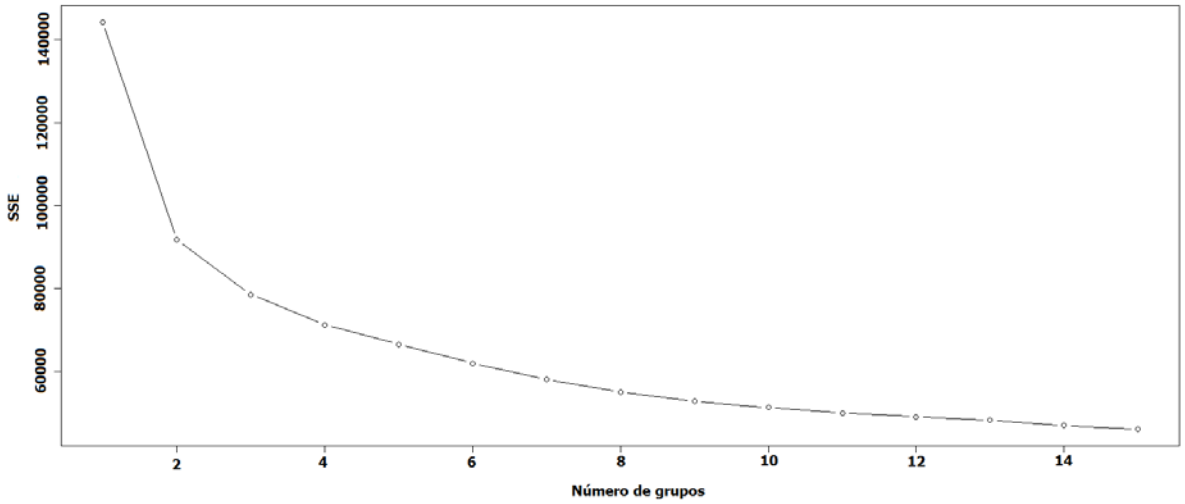

Figura 4. SSE versus número de grupos
[Fonte: elaboração própria]

Com o uso deste gráfico para auxiliar a escolha do número de grupos, tenta-se encontrar o número natural de grupos no conjunto de dados procurando o número de grupos no qual exista um joelho, pico ou queda no desenho da medida de avaliação quando for desenhada contra o número de grupos (TAN; STEINBACH; KUMAR, 2009).

Foram utilizados três algoritmos com funções já existentes no *software* R, o *K-modes* da biblioteca 'klaR', PAM e CLARA da bliblioteca 'cluster'. Devido à seleção dos *modes* inicias no algoritmo *K-modes*, que assim como o *K-means*, pode variar o resultado dependendo dos valores inicialmente escolhidos, os experimentos foram executados 30 vezes para que pudéssemos estabelecer um valor médio das análises. Além disso, a métrica euclidiana foi utilizada para o cálculo das diferenças entre observações nos algoritmos PAM e CLARA.

3.3 Implementação das métricas de classificação

Utilizando a linguagem JAVA foi implementada um código para calcular as medidas de *Precision*, *Recall* e *F1-score*. O código é simples e segue seguinte raciocínio. Primeiramente, é criada a matriz de confusão, formada pelos grupos

previstos e pelas classes atuais de cada artigo. De posse da matriz de confusão, procura-se o maior valor, e assim, de acordo com a interseção que esse valor representa, atribui-se a classe atual ao grupo previsto. Depois desse passo, repete-se o processo e procura-se novamente pelo maior valor na matriz de confusão, desta vez excluindo-se a linha e a coluna do valor que foi encontrado anteriormente, pois já possuem classe atribuída.

Como exemplo, temos a Figura 5, ela ilustra que a classe real *Quantum Science* foi atribuída para o Cluster 0, *Mythology* para o Cluster 1 e *Olympian* para o Cluster 2.

Label	Cluster 0	Cluster 1	Cluster 2
Olympian	768	758	164
Mythology	779	846	79
Quantum Science	819	668	186

Cluster 0 -> "Quantum Science"
Cluster 1 -> "Mythology"
Cluster 2 -> "Olympian"

Figura 5. Atribuição da classe real aos grupos previstos.
[Fonte: Elaboração própria.]

Após atribuir um grupo previsto a uma classe atual, é feito o cálculo das medidas de *Precision, Recall* e *F-score* utilizando as equações (2.6), (2.7) e (2.8) para cada grupo. Por fim, é calculada a média de cada medida que indicará como foi o desempenho da classificação no geral.

Dustin Maass em seu algoritmo para cálculo das métricas de classificação não considerou essa atribuição da classe atual com o grupo previsto. As medidas de *Precision, Recall* e *F-score* são baseadas no valor máximo de cada grupo mesmo sendo esses valores máximos pertencentes à mesma classe.

Comparando-se as duas implementações, pode-se ressaltar que a implementação feita neste trabalho está mais coerente, pois deseja-se a classificação feita pelo algoritmo de agrupamento, não se deve atribuir uma classe atual para mais de uma grupo previsto, como pode acontecer no algoritmo de Maass.

4 Resultados

4.1 Resultados *Data set* #1

Nesta análise, foram selecionados 9 atributos pelo algoritmo *Random Forest* dos 21 inicialmente estabelecidos, como pode-se perceber no gráfico do OOB da Figura 6. O algoritmo LOF de detecção de *outliers* utilizou 15 vizinhos, para o cálculo dos fatores, estabelecendo-se que artigos com os 25 maiores fatores seriam removidos da base de dados, pois eles representariam os *outliers*. O número de vizinhos e o número de fatores que seriam removidos foram obtidos através de testes manuais de afinação de parâmetros.

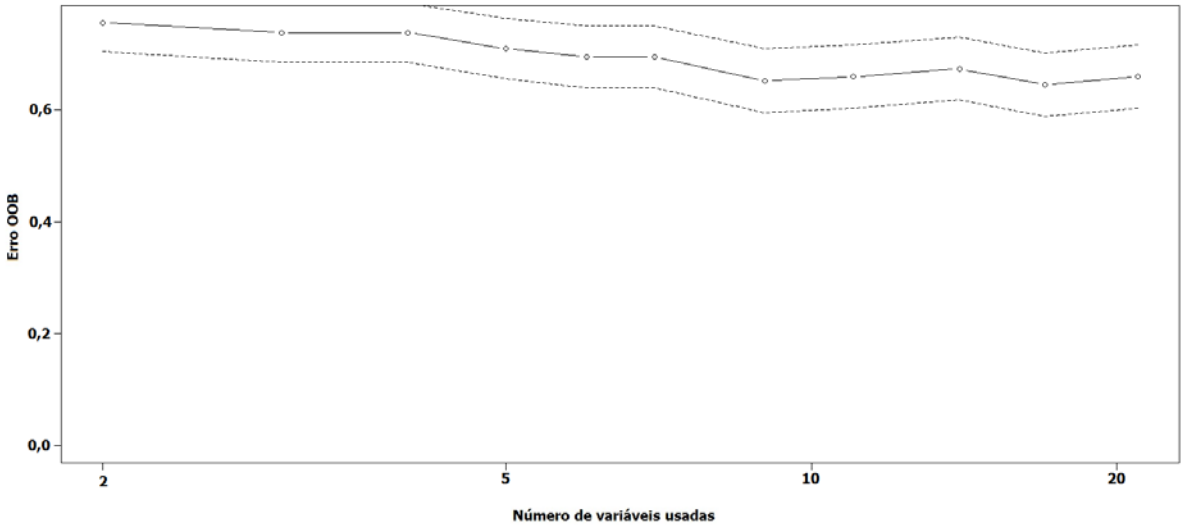

Figura 6. Erro OOB *Data set* #1 x Número de variáveis usadas.

[Fonte: elaboração própria.]

4.1.1 Primeira abordagem

Os resultados da análise e avaliação usando as métricas de classificação foram modestos. Os resultados do *F1-score* variaram entre 0,2031 à 0,3797 contabilizando os três algoritmos. O melhor resultado foi proporcionado pelo algoritmo PAM com *Precision* de 0,4143 e *Recall* igual a 0,3505, resultando em um *F1-score* de 0,3797. Sendo a variação do *F1-score* entre 0 e 1, podemos ver que o resultado desta análise ainda está longe do ideal. A Figura 7 mostra, em porcentagem, que em nenhuma das

30 execuções o algoritmo *K-modes* conseguiu ser superior aos algoritmos PAM e CLARA. Os demais resultados estão no Apêndice A.

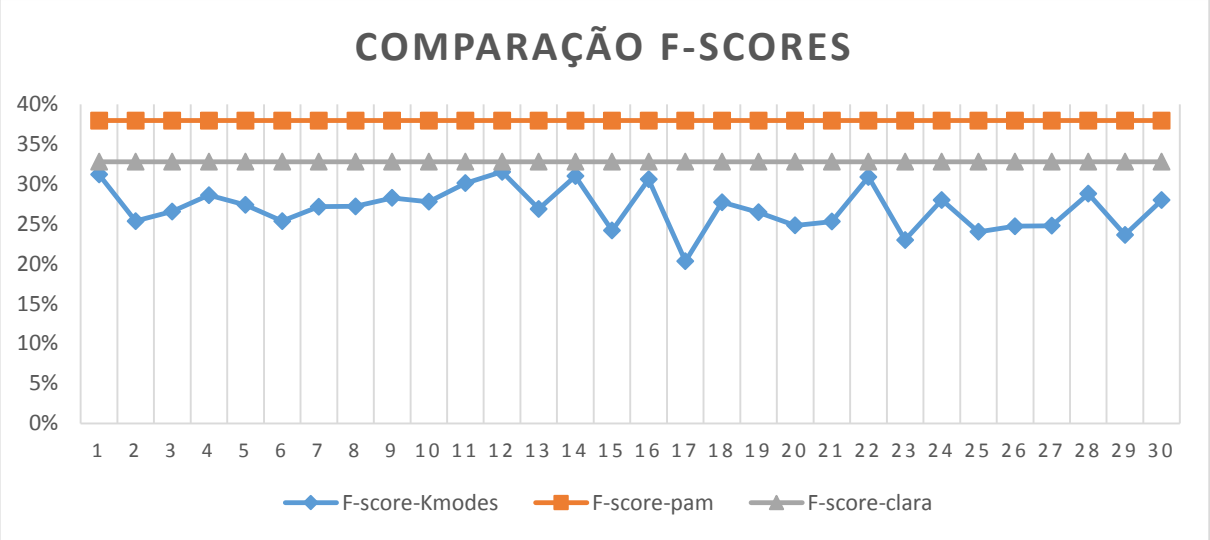

Figura 7. Comparativo entre *F-scores Data set* #1.

[Fonte: Elaboração própria.]

Analisando os grupos e a matriz de confusão na Figura 8, produzidos como resultados do algoritmo PAM, percebemos que alguns grupos possuem bons índices de *Precision*, o que indica que o grupo produzido tem características específicas que podem diferenciar ele dos demais. O grupo 0 (*cluster* 0), por exemplo, que representa a categoria de pássaros (*Birds*) apresenta um número muito pequeno de revisões comparado aos demais, por isso foi bem formado e obteve 0,8205 de *Precision.*

Label	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Cluster 8	Cluster 9	Cluster 1	Cluster 1	Cluster 12	Cluster 13
Birds	16	2	1	0	0	0	0	0	0	0	0	0	0	0
Comedy	0	0	1	2	3	2	2	3	1	2	1	1	0	0
Fortune	0	2	3	0	0	3	6	0	2	0	0	0	1	1
Historica	0	1	0	2	3	4	3	0	1	0	6	0	0	0
Historica	1	0	3	0	1	0	0	0	5	3	0	0	3	1
Laws	0	8	1	1	1	4	2	0	0	0	3	0	0	0
Mammal	0	4	3	0	5	1	2	0	0	0	0	0	2	1
Megaciti	1	0	3	0	0	1	2	0	0	0	3	1	0	7
Movie	0	4	4	0	9	0	0	0	1	0	0	0	0	1
Northern	0	0	2	0	0	1	2	0	0	1	1	5	0	2
People	1	1	0	7	0	2	0	0	0	1	6	0	0	0
Rabbis	1	1	0	0	0	10	1	0	0	0	6	0	0	0
Scientist	0	2	2	0	5	0	2	0	2	1	0	1	4	0
World	0	2	2	0	1	4	1	0	0	1	1	1	1	3

Cluster 0 -> "Birds"
Cluster 1 -> "Laws"
Cluster 2 -> "Mammals"
Cluster 3 -> "People"
Cluster 4 -> "Movie"
Cluster 5 -> "Rabbis"
Cluster 6 -> "Fortune"
Cluster 7 -> "Comedy"
Cluster 8 -> "Historical_Leaders"
Cluster 9 -> "World"
Cluster 10 -> "Historical_Battles"
Cluster 11 -> "Northern"
Cluster 12 -> "Scientists"
Cluster 13 -> "Megacities"

Acc	P_0	R_0	F_0	P_1	R_1	F_1	P_2	R_2	F_2
0.31102	0.8	0.84211	0.82051	0.2963	0.52632	0.379147	0.12	0.47368	0.191489

P_3	R_3	F_3	P_4	R_4	F_4	P_5	R_5	F_5
0.58333	0.4	0.47458	0.32143	0.38889	0.351955	0.3125	0.38889	0.346535

P_6	R_6	F_6	P_7	R_7	F_7	P_8	R_8	F_8
0.26087	0.3	0.27907	1	0.33333	0.5	0.41667	0.35714	0.384615

P_9	R_9	F_9	P_10	R_10	F_10	P_11	R_11	F_11
0.11111	0.29412	0.16129	0.22222	0.21053	0.216216	0.55556	0.16667	0.25641

P_12	R_12	F_12	P_13	R_13	F_13
0.36364	0.16667	0.22857	0.4375	0.05882	0.103704

Precision: 0.41436569181910793
Recall: 0.3505114326432327
F-Score0.3797731887453532

Figura 8. Matriz de confusão e resultado *Data set* #1 usando PAM.

[Fonte: Elaboração própria.]

4.1.2 Segunda Abordagem

Para esta abordagem, utilizou-se o gráfico do SSE versus o número de grupos, usamos o *K-modes* para criação do gráfico, ilustrado na Figura 9, para esta base de dados e o resultado nos auxiliou a escolher o número de grupos igual a 8 pois acima deste valor poderíamos correr risco de ter um *overfit* do modelo, quando o modelo se ajusta mais do que deveria.

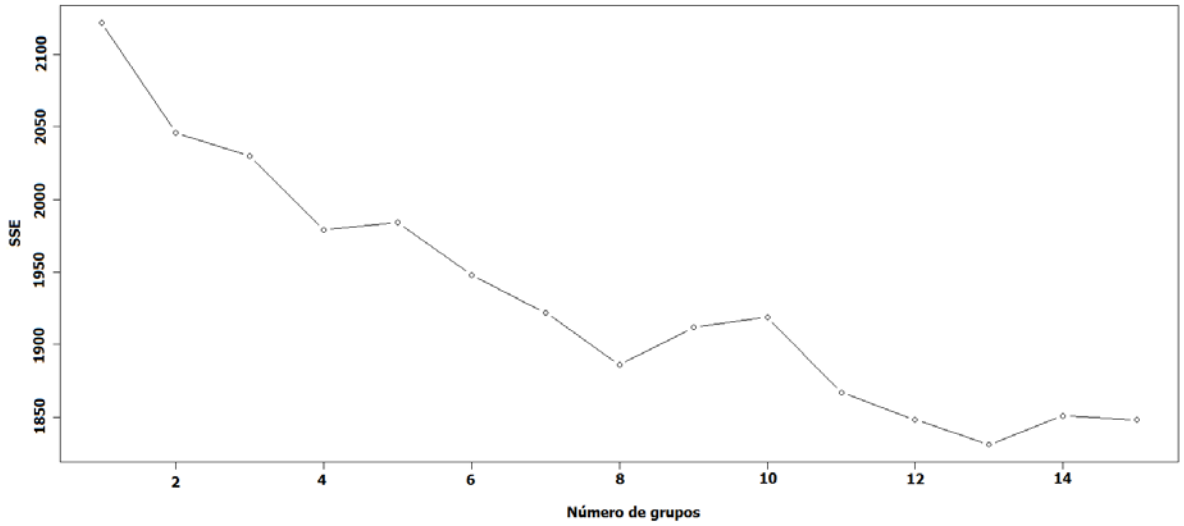

Figura 9. SSE versus número de grupos para *Data set* #1

[Fonte: Elaboração própria.]

Os resultados mostraram nada muito além de divisões de grupo de acordo com a quantidade de revisões, ou seja, grupos com muitas revisões, grupos com poucas revisões e grupos sendo intermediários como mostra a Figura 10. O gráfico mostra que alguns grupos tiveram um decréscimo na média do número de revisões, onde outros apresentaram um aumento. Como este decréscimo foi relativamente pequeno, não trouxe muitas conclusões.

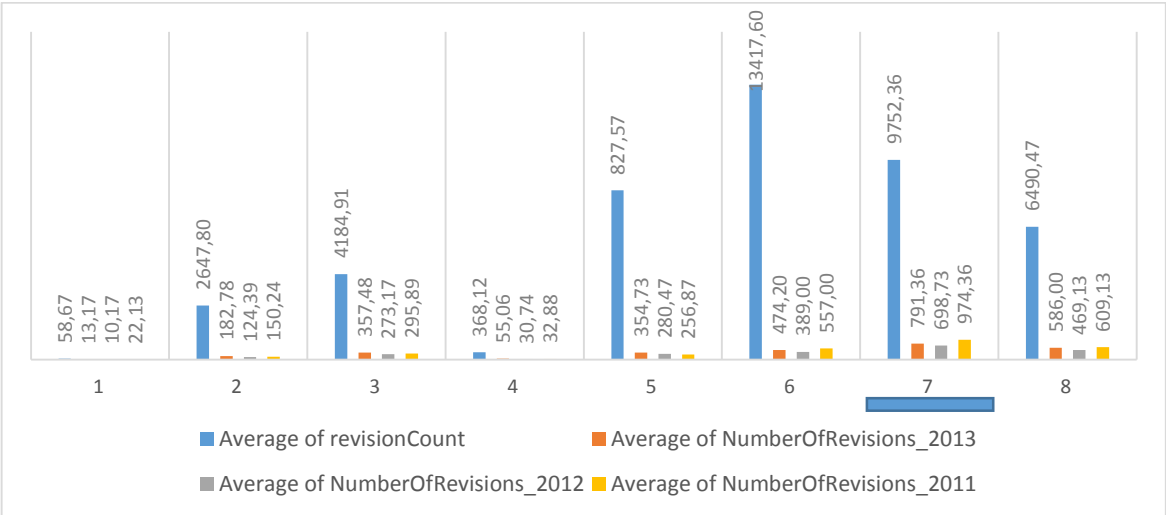

Figura 10. Média de revisões dos grupos usando CLARA.

4.2 Resultados *Data set* #2

Nesta análise, foram selecionados 17 atributos pelo algoritmo *Random Forest* dos 21 inicialmente estabelecidos. O algoritmo LOF de detecção de *outliers* utilizou 50 vizinhos para o cálculo do fatores. E estabeleceu-se que artigos com os 150 maiores fatores seriam removidos da base de dados, pois eles representariam os *outliers*. O número de vizinhos e o número de fatores que seriam removidos foram obtidos através de testes manuais de afinação de parâmetros.

4.2.1 Primeira abordagem

Os resultados da análise na *Data set* #2 proporcionaram um *F-score* melhor do que o da *Data set* #1. O *F-score* variou entre 0,3033 à 0,5738 contabilizando os três algoritmos. O algoritmo *K-modes* foi o que obteve melhor resultado mesmo com a variação proporcionada pela escolha dos *modes* iniciais, ilustrado na Figura 11. Entretanto, os grupos formados pelo algoritmo são desbalanceados como mostrado na Figura 12. Um ponto positivo é o grupo 1 (*cluster* 1) que possui apenas páginas de Olimpíada (Olympian), que nos mostra que parte das páginas de Olimpíada tem uma característica própria.

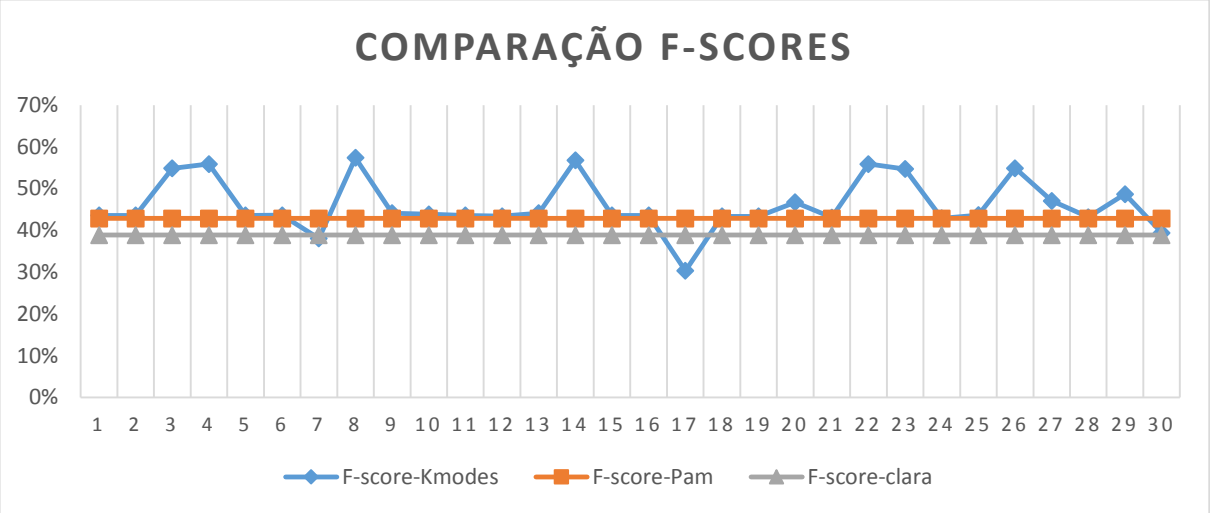

Figura 11. Comparação entre *F-scores Data set* #2

[Fonte: Elaboração própria.]

Label	Cluster 0	Cluster 1	Cluster 2
Olympian	1259	404	0
Mythology	1656	0	0
Quantum Science	1594	0	4

Cluster 0 -> "Mythology"
Cluster 1 -> "Olympian"
Cluster 2 -> "Quantum Science"

Acc	P_0	R_0	F_0	P_1	R_1	F_1	P_2	R_2	F_2
0.419768151	0.36727	1	0.53723	1	0.24293	0.3909	1	0.0025	0.00499

Precision: 0.7890884896872921
Recall: 0.41514586157130173
F-Score0.5440582567086779

Figura 12. Matriz de confusão e resultado *Data set* #2 usando *K-modes*

[Fonte: Elaboração própria.]

Havia uma expectativa maior quanto ao algoritmo CLARA para esta base de dados. Isso porque o CLARA foi desenvolvido como uma extensão do algoritmo PAM para tratamento de grandes quantidades de dados, como é o caso da *Data set* #2 que apresenta mais de 5.000 artigos. Entretanto, os resultados mostraram que o PAM conseguiu obter resultados melhores que o algoritmo CLARA. Todos os resultados estão listados no Apêndice B.

4.2.2 Segunda Abordagem

O gráfico do SSE versus o número de grupos, utilizando o *K-modes*, é ilustrado na Figura 13, e o resultado nos auxiliou, coincidentemente com a escolha na *Data set* #1, a escolher o número de grupos igual a 8 também para evitar um *overfit* do modelo.

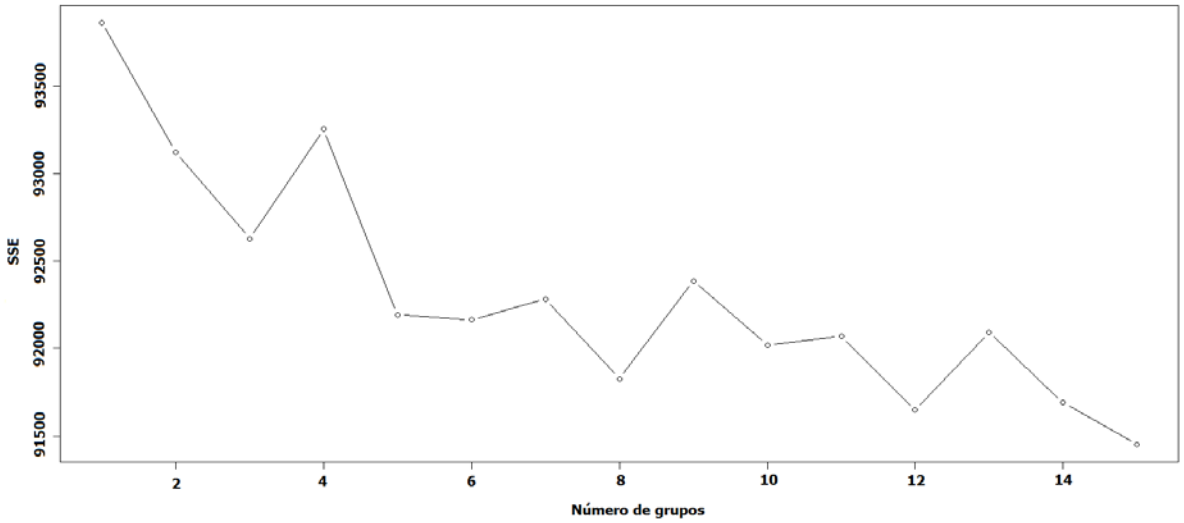

Figura 13. SSE versus número de grupos para *Data set* #2.

[Fonte: Elaboração própria.]

Os resultados para esta base de dados seguiram o mesmo padrão da *Data set* #1, grupo com muitas revisões, grupos com poucas revisões e grupos sendo intermediários. Entretanto, pode-se notar algumas curiosidades.

Os grupos formados pelo algoritmo CLARA, por exemplo, indicaram que no geral houve um aumento no número média de revisões de 2011 a 2013, e foram encontrados dois grupos com alto número de revisões, grupos 5 e 6, mas que possuíam um tamanho médio das revisões menor que outro grupo, grupo 8, cujo número de revisões médio é intermediário, como podem ser vistos nas Figura 14 e Figura 15.

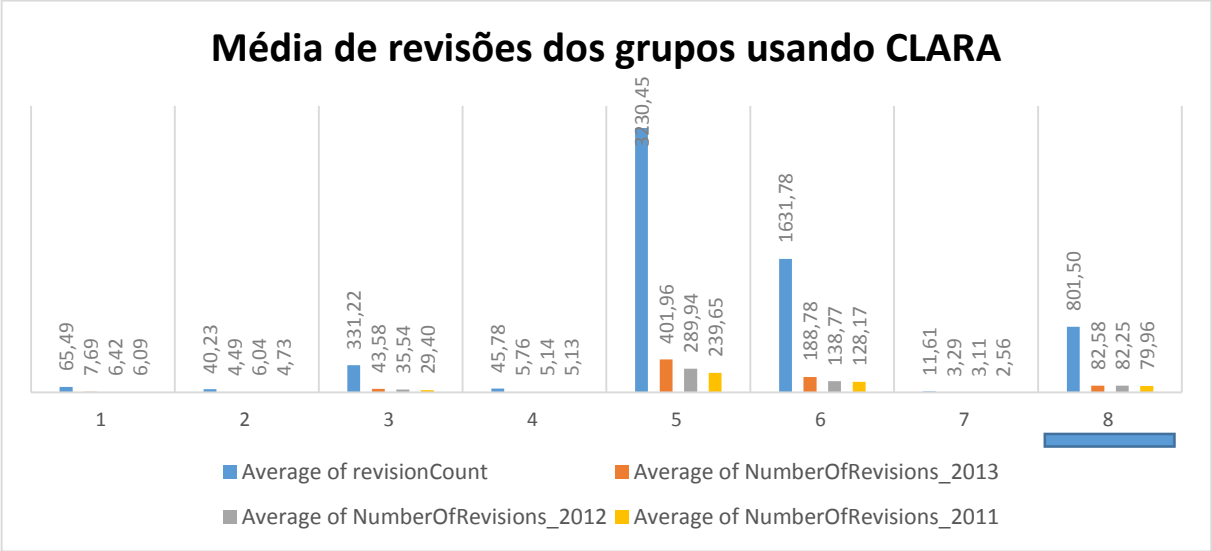

Figura 14. Média de revisões dos grupos usando CLARA.

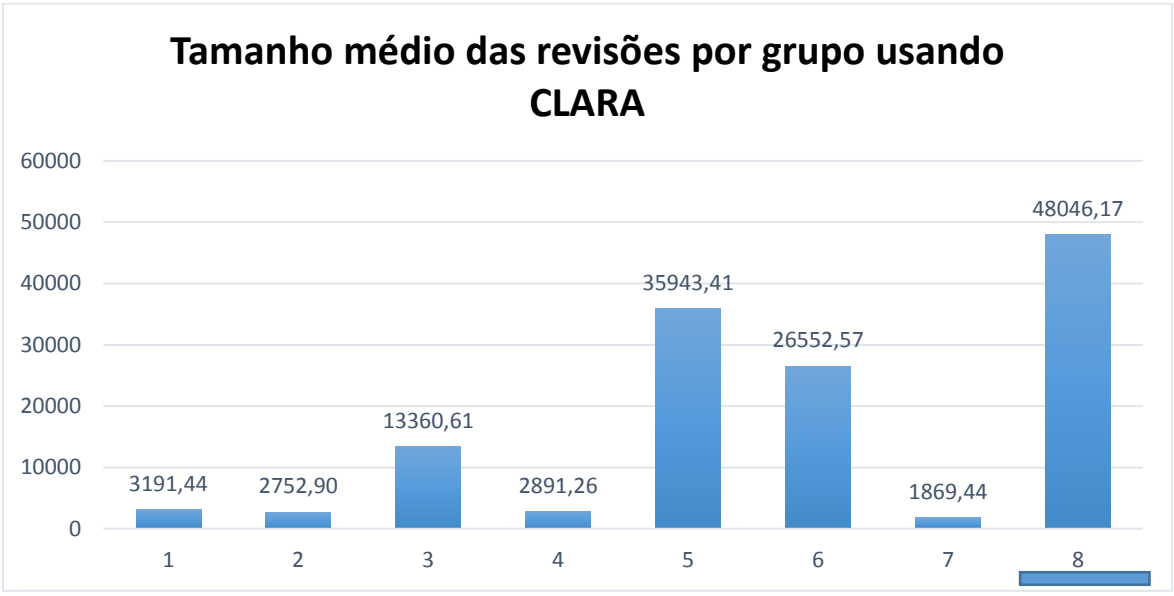

Figura 15. Tamanho médio das revisões por grupo usando CLARA.

Esta observação pode indicar que artigos no Grupo 8, mesmo com número de revisões intermediárias, são artigos onde cada revisão apresenta muito conteúdo, o que torna seu tamanho maior e assim esse grupo ter se destacado como maior tamanho médio das revisões.

Além disso, foi observado que nos três algoritmos o grupo com maior número de revisões representa um grupo seleto de poucos artigos, sendo formado, na sua maioria, por atletas da categoria Olimpíada (*Olympic*).

4.3 Comparação com os resultados de Dustin Maass

Dustin Maass definiu e dividiu as duas base de dados, *Data set* #1 e *Data set* #2, em cinco formas diferentes na tentativa de identificar métricas que mostrassem grande variabilidade entre os artigos. Essas formas de organização das base de dados são descritas a seguir (MAASS, 2013):

- Edições ao longo de um tempo fixo;
- Edições ao longo do tempo de vida (*lifetime*);
- Autores ao longo de um tempo fixo;
- Média do tamanho das edições ao longo do tempo;
- Combinação das formas anteriores.

Dustin utilizou o algoritmo *K-means* para as análises e os resultados obtidos por ele na primeira base de dados, *Data set* #1, variaram o *F1-score* entre 0.0095 à 0,3557, com média de 0,1882. Já na segunda base de dados, *Data set* #2, os resultados variaram o *F1-score* entre 0,1676 à 0,4357 com uma média de 0,3096.

Os resultados encontrados neste trabalho foram para *Data set* #1 um *F1-score* variando entre 0,2031 à 0,3797, com média de 0,3257. Na *Data set* #2, o *F1-score* variou entre 0,3033 à 0,5738, com média de 0,4260.

Comparando os resultados de Dustin com os resultados obtidos neste trabalho, pode-se dizer que houve um pequeno aumento nos índices de *F1-score*. Entretanto, essa pequena diferença torna-se mais relevante se considerarmos que Dustin não fez atribuição das classes reais aos grupos previstos durante o cálculo das medidas de *Precision*, *Recall* e *F-score*.

Os resultados comparados estão no Apêndice C.

5 Conclusões e Trabalhos Futuros

5.1 Conclusões

A análise feita no histórico de revisões dos artigos do *Wikipedia* foram modestas e razoavelmente conclusivas. A primeira aboradadem, análise focada na classificação, embora tivesse havido um aumento nas métricas de avaliação (*Precision, Recall e F1-score*), principalmente com o algoritmo *K-modes*, apresenta grupos desbalanceados e mesmo com o melhor *F1-score* de 0,5738, ainda necessita-se de um melhor desempenho para criação de um sistema aceitável para classificação automática das páginas do *Wikipedia*.

A segunda abordagem, na tentativa de identificar padrões interessantes, também foram modestos e pouco conclusivos. Dos grupos formados, apenas um, na *Data set* #2, apresentou caracterização e tinha o padrão de ser formado por sua maioria atletas com grande número de revisões.

Embora os resultados não tenham sido o esperado, a metodologia aplicada pode ajudar a guiar novas pesquisas inovadoras focando a análise dos dados do *Wikipedia*. Pois, como vimos, existe muita informação que pode ser útil na base de dados do *Wikipedia* e o desafio existente para minerar esta base de dados só deve servir como estímulos para novos estudos.

5.2 Trabalhos Futuros

Para trabalhos futuros, pode-se realizar novas análises utilizando as observações retiradas deste trabalho. A primeira observação é quanto a organização dos dados, de forma a tentar encontrar melhores padrões, acredita-se que a base de dados poderia ser dividida em artigos com poucas revisões e artigos com muita revisões. Feito isso, os atributos seriam criados baseados em um tempo fixo, abrangendo todo tempo de vida dos artigos. Para os artigos com poucas revisões, seriam criados atributos para cada ano. Já para os artigos com muitas revisões, seriam criados atributos para cada mês. As duas formas seriam analisadas separadamente.

Novos algoritmos poderiam ser testados, mas também seria importante ser feito um melhor estudo na afinação dos parâmetros utilizados em cada algoritmo. No algoritmo *K-modes*, por exemplo, poderia ser investigado algum método que atribuísse os *modes* inicias de forma mais controlada, removendo a aleatoriedade dessa escolha.

Outro ponto a ser estudado e aperfeiçoado é a seleção de atributos. O método *Random Forest* utilizado neste trabalho também possui uma certa aleatoriedade, o que não é suficiente para estabelecer um sistema estável para análises, visto que a cada execução do algoritmo pode-se mudar os atributos escolhidos.

Bibliografia

BOULD, D.; HLADKOWICZ, E.; PIGFORD, A.; UFHOLZ, L.; POSTONOGOVA, T.; SHIN, E.; et al. *References that anyone can edit: review of Wikipedia citations in peer reviewed health science literature*. British Medical Journal, 2014.

BREIMAN, L. *Random Forests*. Machine Learning. 2001.

BREUNIG, M.; KRIEGEL, H.; NG, R.; SANDER, J. *LOF: Identifying Density-Based Local Outliers*. 2000.

GOWDA, K.C. e DIDAY, E. *Symbolic clustering using a new dissimilarity measure. Pattern Recognition*. 1991. 567–578p

KAUFMAN, L. e ROUSSEEUW, P.J. *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, New York. 1990.

LAZAREVIC, A.;OZGUR, A.; ERTOZ, L.; SRIVASTAVA, J.; KUMAR, V. *A comparative study of anomaly detection schemes in network intrusion detection*. Proc. 3rd SIAM International Conference on Data Mining. 2003. 25–36.

MAASS, D. *Data mining revision controlled document history metadata for automatic classification*. 2013. 28 f. Tese (Mestrado em Ciência da Computação). University of Wisconsin-Milwaukee. Milwaukee. 2013.

MACQUEEN, J. B. *Some methods for classification and analysis of multivariate observations*. 1967.

PEINE, Z. "*Metriki*". Master's Project Report, University of Wisconsin-Milwaukee, 2012.

RALAMBONDRAINY, H. *A conceptual version of the k-means algorithm. Pattern Recognition Letters*. 1995. 1147–1157p

SHAH, N. "*Capstone Project Report*". Master's Project Report, University of Wisconsin – Milwaukee, 2012.

TAN, P.; STEINBACH, M.; KUMAR, V. *Introdução ao Data Mining – Mineração de dados*. Rio de Janeiro: Ciência Moderna, 2009.

VENABLES, W.; SMITH, D.; e the R Core Team. *Uma Introdução ao R*. 2008

Wikipedia. *História da Wikipédia*. Acesso em: 11 de Novembro de 2014. Disponível em: < http://pt.*Wikipedia*.org/wiki/História_da_*Wikipedia* >.

Apêndice A

Resultados da análise *Data set* #1

	K-modes			Pam			Clara		
Experimento	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
1	0,4290	0,2452	0,3121	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
2	0,3338	0,2045	0,2536	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
3	0,3439	0,2165	0,2657	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
4	0,3457	0,2441	0,2862	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
5	0,3847	0,2127	0,2740	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
6	0,3516	0,1985	0,2538	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
7	0,3373	0,2275	0,2718	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
8	0,3456	0,2245	0,2721	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
9	0,3497	0,2374	0,2828	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
10	0,3297	0,2401	0,2779	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
11	0,4362	0,2300	0,3012	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
12	0,4437	0,2447	0,3155	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
13	0,3456	0,2202	0,2690	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
14	0,4001	0,2533	0,3102	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
15	0,2354	0,2487	0,2419	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
16	0,4063	0,2454	0,3060	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
17	0,2114	0,1956	0,2032	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
18	0,3639	0,2241	0,2774	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
19	0,3669	0,2070	0,2647	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
20	0,3266	0,2003	0,2483	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
21	0,2880	0,2257	0,2531	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
22	0,4984	0,2241	0,3092	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
23	0,2436	0,2176	0,2299	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
24	0,3509	0,2331	0,2801	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
25	0,2915	0,2044	0,2403	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
26	0,2585	0,2371	0,2473	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
27	0,3374	0,1958	0,2478	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
28	0,3497	0,2449	0,2881	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
29	0,2371	0,2355	0,2363	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281
30	0,3470	0,2346	0,2799	0,4144	0,3505	0,3798	0,3192	0,3375	0,3281

Apêndice B

Resultados da análise *Data set #2*

	K-modes			Pam			Clara		
Experimento	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
1	0,4549	0,4187	0,4360	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
2	0,4549	0,4187	0,4360	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
3	0,7883	0,4208	0,5487	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
4	0,7872	0,4332	0,5589	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
5	0,4548	0,4187	0,4360	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
6	0,4552	0,4187	0,4362	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
7	0,4458	0,3323	0,3808	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
8	0,7881	0,4511	0,5738	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
9	0,4669	0,4187	0,4415	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
10	0,6136	0,3412	0,4385	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
11	0,4549	0,4187	0,4360	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
12	0,4558	0,4141	0,4340	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
13	0,4669	0,4187	0,4415	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
14	0,7933	0,4418	0,5676	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
15	0,4549	0,4187	0,4360	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
16	0,4549	0,4187	0,4360	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
17	0,2791	0,3321	0,3033	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
18	0,4556	0,4141	0,4339	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
19	0,4556	0,4141	0,4339	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
20	0,7795	0,3339	0,4676	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
21	0,4529	0,4109	0,4308	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
22	0,7872	0,4332	0,5589	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
23	0,7881	0,4189	0,5470	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
24	0,4539	0,4067	0,4290	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
25	0,4568	0,4187	0,4369	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
26	0,7883	0,4210	0,5489	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
27	0,5418	0,4160	0,4707	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
28	0,6127	0,3335	0,4320	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
29	0,7817	0,3535	0,4868	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890
30	0,4481	0,3521	0,3943	0,4547	0,4054	0,4286	0,3939	0,3842	0,3890

Apêndice C

Tabela comparativa dos resultados obtidos com os resultados de Dustin Maass

Trabalho	Base de Dados	Algoritmo	Valores	Precision	Recall	F1-score
Dustin Maass	Data set#1	K-means	maxímo	0,5595	0,2750	0,3557
			mínimo	0,0051	0,0714	0,0095
			média	0,2345	0,1780	0,1883
Alexandre Azevedo Filho	Data set#1	K-modes	maxímo	0,4984	0,2533	0,3155
			mínimo	0,2114	0,1341	0,2032
			média	0,3387	0,2247	0,2671
		PAM	maxímo	0,4144	0,3505	0,3798
			mínimo	0,3168	0,2673	0,2900
			média	0,4108	0,3476	0,3766
		CLARA	maxímo	0,3907	0,3375	0,3504
			mínimo	0,2253	0,2319	0,2285
			média	0,3302	0,3318	0,3301
Dustin Maass	Data set#2	K-means	maxímo	0,5595	0,3505	0,3798
			mínimo	0,0051	0,0714	0,0095
			média	0,3278	0,2428	0,2695
Alexandre Azevedo Filho	Data set#2	K-modes	maxímo	0,7933	0,4511	0,5738
			mínimo	0,2791	0,3321	0,3033
			média	0,5624	0,4021	0,4604
		PAM	maxímo	0,4546	0,4053	0,4286
			mínimo	0,4546	0,4053	0,4286
			média	0,4546	0,4053	0,4286
		CLARA	maxímo	0,3939	0,3842	0,3890
			mínimo	0,3939	0,3842	0,3890
			média	0,3939	0,3842	0,3890