UNIVERSIDADE CANDIDO MENDES – UCAM
PROGRAMA DE PÓS-GRADUAÇÃO EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL
CURSO DE MESTRADO EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL

Fernando José Ferreira Andinós Júnior

CATEGORIZAÇÃO AUTOMÁTICA DE ARTIGOS DA ENGENHARIA DE PRODUÇÃO UTILIZANDO MÉTODOS DE APRENDIZAGEM DE MÁQUINA

CAMPOS DOS GOYTACAZES, RJ
Março de 2013

UNIVERSIDADE CANDIDO MENDES – UCAM
PROGRAMA DE PÓS-GRADUAÇÃO EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL
CURSO DE MESTRADO EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL

Fernando José Ferreira Andinós Júnior

CATEGORIZAÇÃO AUTOMÁTICA DE ARTIGOS DA ENGENHARIA DE PRODUÇÃO UTILIZANDO MÉTODOS DE APRENDIZAGEM DE MÁQUINA

Dissertação apresentada ao Programa de Pós-Graduação em Pesquisa Operacional e Inteligência Computacional da Universidade Candido Mendes – Campos/RJ, para obtenção do grau de MESTRE EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL.

Orientadora: Prof[a].: Geórgia Regina Rodrigues Gomes, D.Sc.

CAMPOS DOS GOYTACAZES, RJ

Março de 2013

Fernando José Ferreira Andinós Júnior

CATEGORIZAÇÃO AUTOMÁTICA DE ARTIGOS DA ENGENHARIA DE PRODUÇÃO UTILIZANDO MÉTODOS DE APRENDIZAGEM DE MÁQUINA

Dissertação apresentada ao Programa de Pós-Graduação em Pesquisa Operacional e Inteligência Computacional da Universidade Candido Mendes – Campos/RJ, para obtenção do grau de MESTRE EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL.

Aprovado em: 15/03/2013

BANCA EXAMINADORA

_____

Prof[a]. Geórgia Regina Rodrigues Gomes, D.Sc. – Orientadora
UNIVERSIDADE CANDIDO MENDES - UCAM

_____

Prof. Dalessandro Soares Vianna, D.Sc.
UNIVERSIDADE FEDERAL FLUMINENSE - UFF

_____

Prof. Mark Douglas de Azevedo Jacyntho, D.Sc.
UNIVERSIDADE CANDIDO MENDES - UCAM

_____

Prof. Helder Gomes Costa, D.Sc.
UNIVERSIDADE FEDERAL FLUMINENSE - UFF

CAMPOS DOS GOYTACAZES, RJ

Março de 2013

Agradecimentos

Agradeço primeiramente a Deus, meu melhor amigo. A minha esposa Luciana, companheira de vida, pela sua força, carinho e compreensão em todos os momentos. As minhas filhas Natália e Bárbara, que mesmo sem saber, me ensinam algo novo a cada dia. A minha orientadora Geórgia Gomes, por compartilhar seu conhecimento e seu tempo, sempre paciente e otimista. Ao meu gerente José Carlos Ruela, pela compreensão e apoio durante todo o curso, me permitindo conciliar o trabalho e os estudos.

Bem-aventurado o homem que acha sabedoria, e o homem que adquire conhecimento;
Porque é melhor a sua mercadoria do que artigos de prata, e maior o seu lucro que o ouro mais fino.
Provérbios 3:13-14

RESUMO

CATEGORIZAÇÃO AUTOMÁTICA DE ARTIGOS DA ENGENHARIA DE PRODUÇÃO UTILIZANDO MÉTODOS DE APRENDIZAGEM DE MÁQUINA

O presente trabalho apresenta três métodos tradicionais de aprendizagem de máquina: *Naive Bayes*, *k-Nearest Neighbor* (k-NN) e *Support Vector Machines* (SVM) e propõe um método de grupo para realizar a categorização de artigos da Engenharia de Produção, que atualmente no Brasil, divide-se em onze áreas principais de publicação, com o objetivo de auxiliar alunos e professores na escolha da melhor área para submissão de seus trabalhos. Para isso, os métodos utilizados baseiam-se no conteúdo textual do documento, tendo como insumo de aprendizagem, artigos previamente publicados em anais de dois dos principais congressos de Engenharia de Produção, o Encontro Nacional de Engenharia de Produção (ENEGEP) e o Simpósio de Engenharia de Produção (SIMPEP). Baseado nos resultados experimentais apresentados, o método de grupo proposto obteve melhor desempenho nas métricas definidas (acurácia, precisão e abrangência) que os métodos tradicionais isoladamente. Os principais fatores para a elaboração desse trabalho foram a dificuldade exposta por alunos e professores em algumas vezes escolher a área de submissão mais adequada para seus trabalhos, somado ao crescimento observado no número de artigos publicados nesses congressos nos últimos anos. Espera-se que este trabalho contribua para o crescimento, organização e qualidade da produção científica em Engenharia de Produção no Brasil.

PALAVRAS-CHAVE: Mineração de Texto, Categorização de Documentos, Gestão do Conhecimento.

ABSTRACT

AUTOMATIC CLASSIFICATION OF INDUSTRIAL ENGINEERING PAPERS USING MACHINELEARNING METHODS

This work presents three traditional methods of machine learning: Naive Bayes, k-Nearest Neighbor (k-NN) and Support Vector Machines (SVM) and proposes a grouping method to perform the categorization of Industrial Engineering papers, with the goal of helping students and teachers to choose the best area for paper submission. Currently, in Brazil, Industrial Engineering is divided into eleven main publication areas. To achieve its goal, the methods use as input of learning, the textual content of the papers previously published in proceedings of two major Industrial Engineering conferences, the ENEGEP and SIMPEP. Based on the experimental results, the proposed group method performed better on defined metrics (accuracy, precision and recall) than traditional methods alone. The main motivational factors for the development of this work have been the difficult exposed sometimes by students and teachers to choose the most suitable submission area to their papers, coupled with the growth in the number of papers published in these conferences in recent years. It is hoped that this work will contribute to the growth, organization and quality of scientific production in Production Engineering in Brazil.

KEYWORDS: Text Mining, Document Categorization, Knowledge Management.

FERNANDO JOSÉ FERREIRA ANDINOS JUNIOR

CATEGORIZAÇÃO AUTOMÁTICA DE ARTIGOS DA ENGENHARIA DE PRODUÇÃO UTILIZANDO MÉTODOS DE APRENDIZAGEM DE MÁQUINA.

Dissertação apresentada ao Programa de Pós-Graduação em Pesquisa Operacional e Inteligência Computacional, da Universidade Candido Mendes-Campos/RJ, para a obtenção do grau de MESTRE EM PESQUISA OPERACIONAL E INTELIGÊNCIA COMPUTACIONAL.

Aprovado em 15 de março de 2013.

BANCA EXAMINADORA

Profª. Geórgia Regina Rodrigues Gomes, D.Sc - Orientadora
Universidade Candido Mendes

Prof. Dalessandro Soares Vianna, D.Sc
Universidade Candido Mendes
Universidade Federal Fluminense

Prof. Mark Douglas de Azevedo Jacyntho, D.Sc
Universidade Candido Mendes

Prof. Helder Gomes Costa, D.Sc
Universidade Federal Fluminense

CAMPOS DOS GOYTACAZES, RJ
2013

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

ABEPRO	Associação Brasileira de Engenharia de Produção
AM	Aprendizagem de máquina
CAPES	Coordenação de Aperfeiçoamento de Pessoal de Nível Superior
CSV	Comma Separated Values (valores separados por vírgula)
ENCEP	Encontro Nacional de Coordenadores de Cursos de Engenharia de Produção
ENEGEP	Encontro Nacional de Engenharia de Produção
GT	Grupo de Trabalho
IE	Information Extraction
k-NN	k-Nearest Neighbor
PDF	Portable Document Format
RI	Recuperação de Informações
SIMPEP	Simpósio Brasileiro de Engenharia de Produção
SVM	Support Vector Machines

SUMÁRIO

1. INTRODUÇÃO

Neste capítulo são apresentados a motivação, os objetivos e a organização da dissertação. Na primeira seção faz-se uma breve descrição do assunto e sua importância. Em seguida são apresentados os objetivos do trabalho. Ao final, descreve-se a forma segundo a qual a dissertação está organizada.

1.1. MOTIVAÇÃO

O Brasil atualmente possui 486 cursos de graduação em Engenharia de Produção reconhecidos pelo Ministério da Educação e Cultura (MEC) (NUPENGE, 2012) e 58 cursos de pós-graduação *strictu-senso* recomendados pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), sendo estes: 32 de mestrado acadêmico, 16 de doutorado e 10 de mestrado profissional (CAPES, 2012). Além de atender a demanda crescente do mercado de trabalho, boa parcela desses indivíduos contribui com a produção científica, gerada principalmente por professores e alunos dos cursos de pós-graduação existentes no país.

A escolha da melhor área para submissão de artigos científicos em Engenharia de Produção, que possui uma característica abrangente e multidisciplinar, pode não ser trivial. De acordo com o último documento elaborado pela Comissão de Graduação da Associação Brasileira de Engenharia de Produção (ABEPRO), aprovado nas reuniões do GT de

Graduações ocorridas no Encontro Nacional de Coordenadores de Cursos de Engenharia de Produção (ENCEP) realizado em 2008 e no Encontro Nacional de Engenharia de Produção (ENEGEP) 2008, a Engenharia de Produção atualmente divide-se em 11 áreas passíveis de publicação em congressos no Brasil, enumeradas na Tabela 1, subdivididas em 58 subáreas (ABEPRO, 2012).

Tabela 1 - Áreas da Engenharia de Produção passíveis de publicação no Brasil (ABEPRO, 2012).

Áreas da Engenharia de Produção (Categorias)	
1	GESTÃO DA PRODUÇÃO
2	GESTÃO DA QUALIDADE
3	GESTÃO ECONÔMICA
4	ERGONOMIA E SEGURANÇA DO TRABALHO
5	GESTÃO DO PRODUTO
6	PESQUISA OPERACIONAL
7	GESTÃO ESTRATÉGICA E ORGANIZACIONAL
8	GESTÃO DO CONHECIMENTO ORGANIZACIONAL
9	GESTÃO AMBIENTAL
10	EDUCAÇÃO EM ENGENHARIA DE PRODUÇÃO
11	ENG. PROD., SUSTENTABILIDADE E RESPONSABILIDADE SOCIAL

Diante disso, professores e alunos em alguns momentos demonstram dificuldade em decidir a área mais adequada para o envio de seus trabalhos. Então, se existisse uma ferramenta que baseada no conteúdo textual, os auxiliasse sugerindo a área mais apropriada para submissão do artigo, a probabilidade de aceitação aumentaria, pois seriam direcionados a avaliadores mais indicados. Além disso, uma vez aprovado e categorizado na área mais aderente ao seu conteúdo, o trabalho teria melhor divulgação e atingiria o público esperado pelos autores.

Além dos fatores descritos anteriormente, observa-se uma tendência crescente no número de artigos publicados nos últimos anos em dois dos principais congressos nacionais com abrangência internacional da área de Engenharia de Produção: O ENEGEP, promovido pela ABEPRO e o SIMPEP (Simpósio de Engenharia de Produção), organizado pelo Departamento de

Engenharia de Produção da Universidade Estadual Paulista - Campus Bauru (DEP-UNESP), que pode ser comprovado pelo gráfico da Figura 1. Porém, apesar do grande número de artigos publicados, de 2006 a 2011 a taxa de aprovação de artigos no ENEGEP foi de 54,61% (informação pessoal)[1]. Espera-se que utilizando a metodologia proposta neste trabalho, este índice seja melhorado.

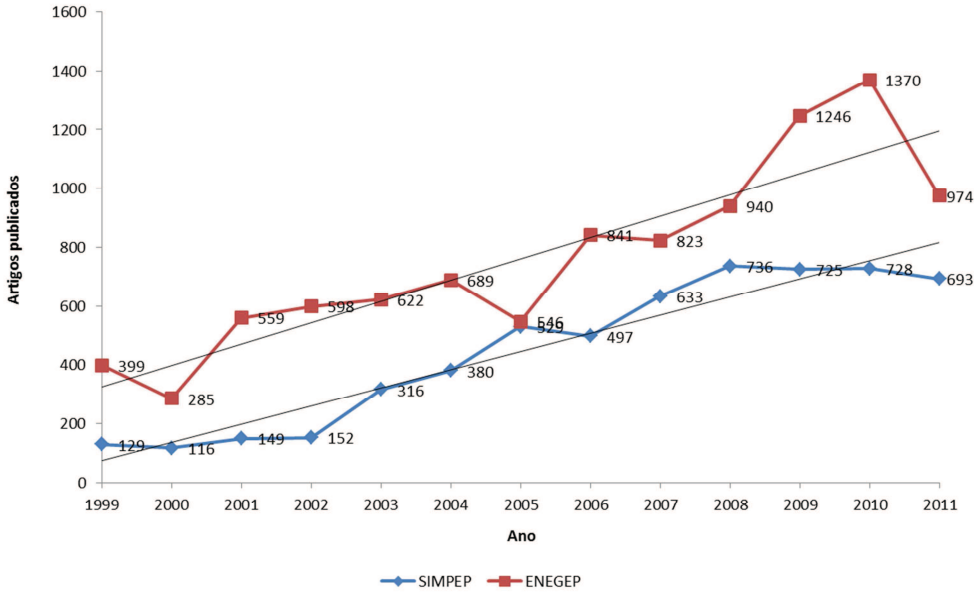

Figura 1 - Número de artigos publicados no ENEGEP e SIMPEP nas edições de 1999 a 2011 (ABEPRO, 2011) (SIMPEP, 2011).

1.2. OBJETIVOS DA DISSERTAÇÃO

O objetivo deste trabalho é utilizar técnicas de Aprendizagem de Máquina (AM) e Mineração de Textos, para que a partir de artigos previamente categorizados, isto é, publicados em edições anteriores do ENEGEP e SIMPEP em uma determinada área, consiga-se predizer a categoria (área de publicação) de novos artigos, auxiliando os autores na escolha da melhor área para submetê-lo em congressos de Engenharia de Produção.

Os objetivos específicos deste trabalho consistem em:

- Estudar os três principais métodos de aprendizagem de máquina: *Naive Bayes*, *k-Nearest Neighbor* (k-NN) e *Support Vector Machines* (SVM) para categorização de documentos;

[1]Informações obtidas com o setor de comunicação da ABEPRO através do e-mail secretaria@abepro.org.br em 4 dez. 2012.

- Propor um método de grupo para realizar a categorização de artigos da Engenharia;
- Fazer um estudo de caso com cada método e compará-los com os resultados do método de grupo proposto no trabalho.

1.3. ORGANIZAÇÃO DA DISSERTAÇÃO

Esta dissertação está organizada da seguinte forma:

- O capítulo 2 apresenta a fundamentação, ou seja, os conceitos teóricos necessários para o entendimento do trabalho.
- O capítulo 3 descreve a metodologia adotada para utilizar as técnicas de Mineração de Textos e apresenta o método de grupo proposto pelo trabalho.
- O capítulo 4 os resultados experimentais são apresentados e analisados conforme as métricas de avaliação de desempenho definidas no capítulo 3.
- O capítulo 5 apresenta as conclusões do trabalho, contribuições, publicações e propostas de trabalhos futuros.

2. FUNDAMENTAÇÃO

2.1. APRENDIZAGEM DE MÁQUINA

As técnicas de Aprendizagem de Máquina (AM) empregam um princípio de inferência chamado indução, onde se obtém conclusões genéricas a partir de um conjunto particular de exemplos. O aprendizado indutivo pode ser dividido em supervisionado e não supervisionado. No aprendizado supervisionado o conhecimento é apresentado através de conjuntos de exemplos na forma de uma entrada e saída desejada (HAYKIN, 1999). O algoritmo que implementa a técnica de AM extrai a representação do conhecimento a partir desses exemplos. O objetivo é que a representação gerada seja capaz de produzir saídas corretas para novas entradas não apresentadas previamente. Neste caso, tem-se uma categorização.

Segundo Souto et al (2003), no aprendizado não-supervisionado não existem exemplos previamente categorizados. O algoritmo aprende a representar as entradas de acordo com uma medida de qualidade. Utiliza-se dessas técnicas principalmente quando o objetivo for encontrar padrões ou tendências que auxiliem no entendimento todos dados, por exemplo, em clusterização ou agrupamento. Na Figura 2, é ilustrada a diferença entre a clusterização e a categorização. No fluxo apresentado na Figura 2(a), como não existe nenhum conhecimento prévio a respeito da coleção de documentos, o algoritmo, representado graficamente pela “Ferramenta de Clusterização” irá agrupar os documentos de acordo com a semelhança entre

eles, criando os chamados “clusters”. Na Figura 2(b), que representa a categorização, primeiramente define-se em quais categorias os documentos serão categorizados e antes de efetivamente se apresentar a coleção de documentos, são utilizados exemplos de cada uma das categorias de forma que seja criado um modelo de representação das categorias que o algoritmo irá utilizar para decidir qual delas representa o documento.

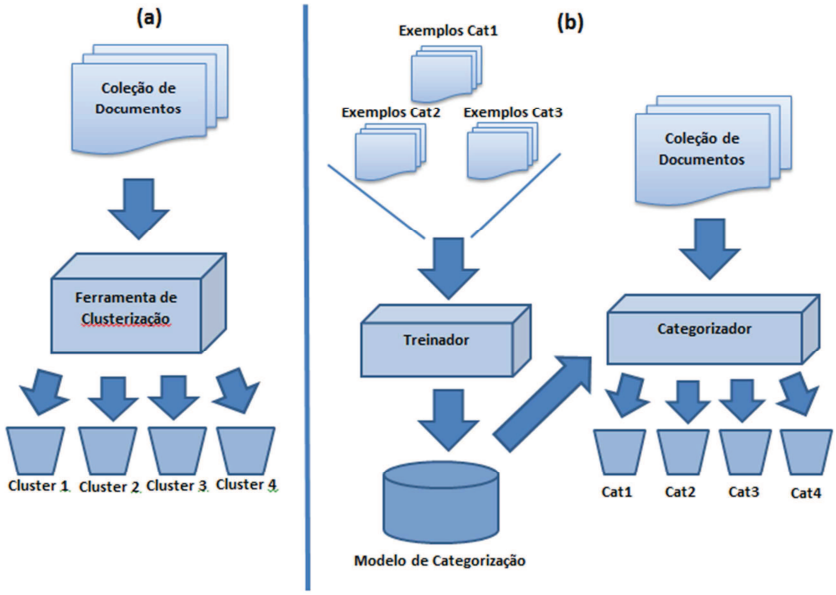

Figura 2 - Clusterização (a) versus Categorização (b) (DORRE; GERSTL; SEIFFERT, 1999).

No presente trabalho, utiliza-se de técnicas de aprendizado supervisionado. Sendo assim, dado um conjunto de n exemplos categorizados na forma $(x_i;y_i)$, em que x_i representa um exemplo i e y_i denota sua categoria (com $1 \leq i \leq n$), deve-se produzir um categorizador que consiga predizer a categoria de novos dados. Esse processo de indução de um categorizador, tendo como insumo uma amostra de dados, é chamado treinamento.

O categorizador obtido também pode ser visto como uma função f, a qual recebe um dado x e fornece uma predição y.

As categorias representam o fenômeno de interesse sobre o qual se deseja fazer previsões. Neste trabalho, em que as predições assumem valores discretos $(1,...,k)$, tem-se um problema de categorização. Caso as predições possuam valores contínuos, tem-se uma regressão. Um problema de ca-

tegorização onde $k = 2$ denomina-se binário. Nos casos onde $k > 2$, o problema é denominado multi-classes.

Segundo Lorena e Carvalho (2007), cada exemplo é normalmente representado por um vetor de características (*feature vectors*). Cada característica, também denominada atributo, expressa um determinado aspecto do exemplo.

De forma geral, existem dois tipos de atributos: discretos e contínuos. Um atributo é dito como discreto quando possui um conjunto de valores enumeráveis. Tais atributos são muitas vezes representados usando variáveis de números inteiros, e podem ser categorizados. Um caso especial de atributos discretos são os atributos binários, que assumem apenas dois valores, por exemplo, verdadeiro e falso. Atributos binários são muitas vezes representados como variáveis booleanas ou como variáveis inteiras que só recebem os valores 0 e 1. Os atributos contínuos, por sua vez são do tipo real e é possível definir uma ordem linear nos valores assumidos, por exemplo, temperatura, altura e peso. (TAN; STEINBACH; KUMAR, 2009).

Ainda de acordo com Tan, Steinbach e Kumar (2009), um requisito importante para as técnicas de AM é a capacidade de lidar com dados imperfeitos, chamados ruídos. Um ruído é um componente aleatório de um erro de medição, que pode envolver a distorção de um valor ou a adição de objetos ilegítimos. A técnica de AM deve ser capaz de lidar com ruídos presentes nos dados, procurando não fixar a obtenção dos categorizadores sobre esse tipo de caso. Deve-se também minimizara influência de *outliers* no processo de indução. Os *outliers* são exemplos muito distintos dos demais presentes no conjunto de dados. Esses dados podem ser ruídos ou casos muito particulares, raramente presentes no domínio. Os conceitos referentes à geração de um categorizador a partir do aprendizado supervisionado são representados de forma simplificada na Figura 3. Tem-se nessa figura um conjunto com n dados. Cada dado x_i, onde $1 \leq i \leq n$, possui m atributos, ou seja, $x_i=(x_i,...,x_{im})$. As variáveis y_i representam as categorias. A partir dos exemplos e as suas respectivas categorias, o algoritmo de AM extrai um categorizador.

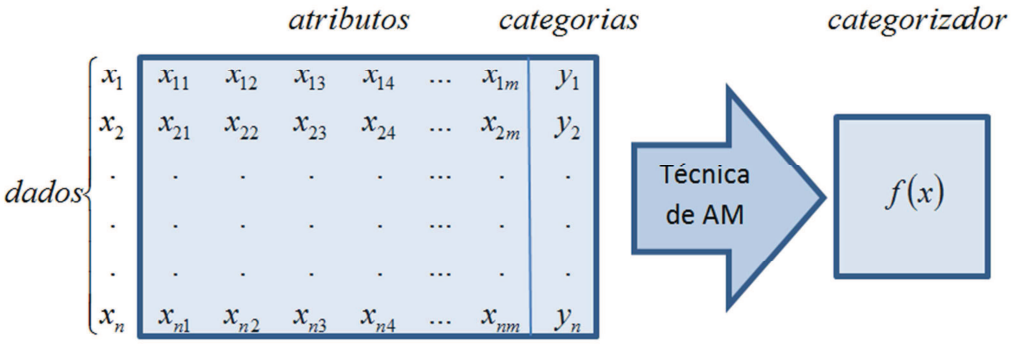

Figura 3 - Indução de um categorizador em aprendizado supervisionado (LORENA; CARVALHO, 2007).

De forma a estimar a taxa de predições corretas e incorretas de um determinado categorizador, divide-se o conjunto de exemplos em dois subconjuntos: um de treinamento e outro de teste. O subconjunto de treinamento é utilizado no aprendizado, para extração do conhecimento e criação do modelo de categorização. Já o subconjunto de teste é utilizado para medir a eficácia do aprendizado com a predição da categoria de exemplos desconhecidos.

Um conceito comumente empregado em AM é o de generalização de um categorizador, definida como a sua capacidade de prever corretamente a categoria de novos dados. Quando o modelo se especializa nos dados utilizados em seu treinamento, apresentando uma baixa taxa de acerto quando confrontado com novos dados, tem-se a ocorrência de um superajustamento (*overfitting*). É também possível induzir hipóteses que apresentem uma baixa taxa de acerto mesmo no subconjunto de treinamento, configurando uma condição de subajustamento (*underfitting*). Essa situação pode ocorrer, por exemplo, quando os exemplos de treinamento disponíveis são pouco representativos ou quando o modelo obtido é muito simples (MONARD; BARANAUSKAS, 2003).

2.2. MINERAÇÃO DE TEXTOS

Segundo Feldman e Sanger (2007), a Mineração de Textos, ou *Text Mining*, pode ser definida como um processo de descoberta de conhecimento intensivo no qual o usuário interage com uma coleção de documentos textuais não estruturados ou semiestruturados, por meio de um conjunto de ferramentas de análise, buscando extrair conhecimento útil.

Aplicações clássicas da Mineração de Textos originam-se da Mineração de Dados, ou *Data Mining*, como clusterização e categorização de documentos. Em ambos, a ideia é transformar o texto em um formato estruturado baseado na frequência de seus termos e posteriormente aplicar técnicas estatísticas e de AM. (FEINERER; HORNIK; MEYER, 2008).

> Através da análise de textos é possível então a descoberta de conceitos, classificações automatizadas e sumarizações para documentos não estruturados. Trata-se de um campo multidisciplinar que envolve várias técnicas, tais como recuperação de informação, análise de texto e categorização de texto, extração de informação. (Gomes, 2005)

Dorre, Gerstl e Seiffert (1999) destacam que o grande desafio da Mineração de Textos é exatamente o fato de a informação estar na forma textual não estruturada, e por esse motivo, não está pronta para ser utilizada por computadores. Essa é sua principal diferença para a Mineração de Dados e também seu maior desafio: a complexa fase preparatória de seleção de características (atributos) e representação dos documentos. Além disso, a cardinalidade do conjunto de recursos que podem ser extraídos de uma coleção de documentos geralmente é muito alta, facilmente chegando a milhares. Há duas consequências disto que afetam o processo de mineração de textos:

1- A tarefa de seleção de características deve ser automática, uma vez que não é mais viável ter um ser humano para analisar cada recurso para decidir se quer usá-lo ou não.

2- O passo de análise de distribuição tem de ser capaz de lidar com vetores de alta dimensionalidade, porém escassamente povoados (a maioria das palavras aparece em poucos documentos). Isso mui-

tas vezes requer versões especiais e implementações dos algoritmos analíticos utilizados em mineração de dados.

Várias outras áreas desempenham papéis importantes na Mineração de Dados e Mineração de Textos. Os sistemas de Bancos de Dados, em especial, são necessários para fornecer eficiente suporte ao armazenamento, indexação e processamento de consultas. Técnicas de computação de alto desempenho (paralela) são muitas vezes importantes para abordar o tamanho volumoso de alguns conjuntos de dados. Técnicas distribuídas também podem auxiliar a abordar a questão do tamanho e são essenciais quando os dados não podem ser consolidados em um único local (TAN; STEINBACH; KUMAR, 2009).

Um trabalho de Mineração de Textos, de forma geral divide-se em três fases principais, representadas na Figura 5: A fase de **Pré-processamento** ou preparação dos dados; a fase de **Processamento**, que compreende a extração e análise dos dados; e a fase de **Pós-processamento,** onde é feito pelo usuário, a análise das descobertas realizadas. Alguns termos existentes na Figura 5 serão detalhados nas próximas subseções.

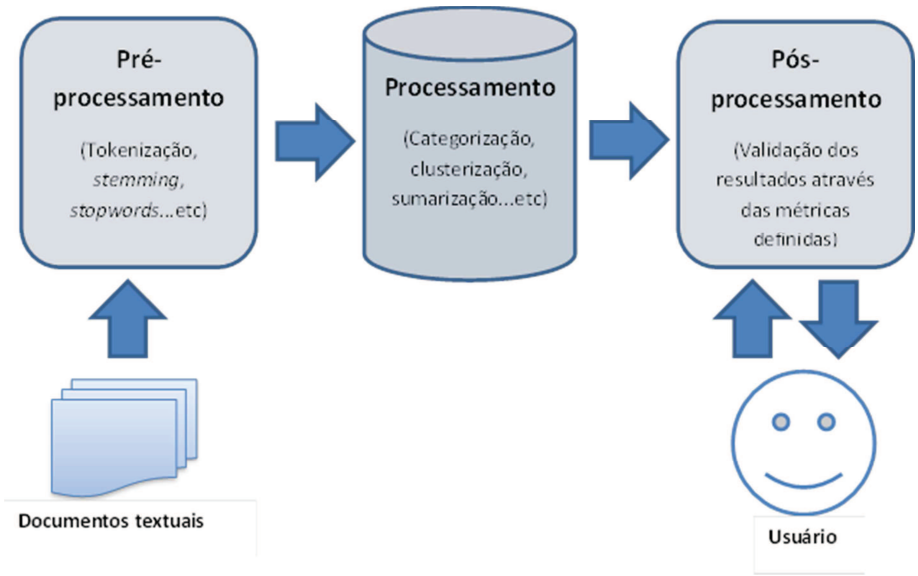

Figura 4 - Principais fases da Mineração de Textos (Feldman e Sanger, 2007).

2.2.1. Fase de Pré-processamento

Essa etapa é de suma importância para o processo de mineração de textos, pois diz respeito à limpeza e preparação dos documentos, culminando em sua representação de forma que possam ser utilizados na fase seguinte.

Algumas subetapas realizadas nesta fase são: a tokenização, a remoção de *Stopwords* e o *stemming*. Cada uma dessas etapas deve ser aplicada, nesta ordem, não sendo mandatória a execução de todas elas. A explicação sobre o papel de cada uma dessas subetapas será fornecida adiante.

2.2.1.1. Representação dos documentos

Os algoritmos de aprendizagem (categorizadores) não podem processar os documentos textuais diretamente em sua forma original. Por isso, durante a fase de pré-processamento, dá-se a conversão em uma representação mais manipulável. Tipicamente, os documentos são representados por vetores de características, que neste caso são compostos de termos com seus referidos pesos.

O modelo mais comum de representação é o saco de palavras, do inglês *bag of words*, que utiliza todos os termos como características, não considerando a relação semântica entre eles (FEINERER; HORNIK; MEYER, 2008). Dessa forma, a dimensão do espaço de características é igual ao número de termos diferentes encontrados em todos os documentos da coleção. Os métodos de atribuir pesos aos termos podem variar. O mais simples é o binário, onde o peso de um termo é 1, se ele está presente no documento, ou 0, se não está presente. Esquemas mais complexos levam em consideração a frequência do termo no documento, na categoria e em toda coleção. A medida mais amplamente utilizada para atribuir pesos é a TF-IDF (*Term Frequency - Inverse Document Frequency*), representada na equação (1).

$$TF - IDF(w,\vec{d}) = TermFreq(w,\vec{d}) \cdot \log\left(\frac{N}{DocFreq(w)}\right) \quad (1)$$

Onde:

$TermFreq(w,\vec{d})$: Frequência do termo no documento;

$DocFreq(w)$: Número de documentos contendo o termo *w*;

N: Total de documentos;

2.2.1.2. Tokenização

É o processo de dividir o fluxo contínuo de caracteres de um documento em componentes que sejam significativos para o objetivo do processo de mineração. Esta divisão pode ocorrer em vários níveis diferentes. Os documentos podem ser divididos em capítulos, seções, parágrafos, frases, palavras ou até mesmo sílabas ou fonemas.

A abordagem mais frequentemente encontrada em sistemas de mineração de texto envolve a quebra do texto em frases e termos. Neste processo também pode ocorrer, caso seja de interesse do usuário, a transformação dos termos em maiúsculas ou minúsculas, remoção de dígitos, pontuações e caracteres especiais, dentre outros critérios de exclusão.

2.2.1.3. Remoção de *stopwords*

O processo de remoção de *stopwords* é utilizado para remover um conjunto de palavras que são tão comuns na língua que o valor de sua informação é praticamente nulo. Essas palavras geralmente são preposições, artigos, conjunções, alguns verbos, nomes, adjetivos e advérbios. Para isto, deve-se criar uma lista, denominada *stop-list* no idioma referente ao domínio estudado, contendo essas palavras irrelevantes. Como benefício, tem-se a redução

da dimensão do vetor de representação do documento, facilitando o processo de mineração. (BARION; LAGO, 2008)

2.2.1.4. *Stemming*

Stemming é o processo automático de remoção dos prefixos e sufixos dos termos e extração de seus radicais, ou *stems*. É uma técnica amplamente utilizada na mineração de textos, pois reduz sua complexidade sem qualquer perda significativa para a maioria das aplicações, especialmente se adotado o modelo de representação *bag of words* (FEINERER; HORNIK; MEYER, 2008).

Viera e Virgil (2007) fazem uma ótima revisão deste tema, enumerando vários algoritmos já produzidos para essa finalidade, como o de Porter, criado inicialmente para a língua inglesa na década de 80 e, desde então, vem sendo adaptado para diversas outras línguas e o algoritmo de Orengo (ORENGO, 2001), criado especificamente para a língua portuguesa.

Os algoritmos de *stemming* baseiam-se em aplicação de regras ou critérios para realizar as transformações necessárias. Para exemplificar, seguem de forma simplificada os passos realizados pelo algoritmo de Porter, utilizado neste trabalho pela sua implementação na linguagem *snowball* (PORTER, 2011):

1- Tratamento das vogais nasalizadas *ã* e *õ* como vogais seguidas por consoante da seguinte forma: Transformando em a~ e o~, onde ~ é o caracter separador interpretado pelo algoritmo como consoante;
2- Remoção dos sufixos;
3- Remoção dos sufixos verbais, se o passo anterior não tratou;
4- Remoção do sufixo i, se precedido de c;
5- Remoção dos sufixos residuais *os*, *a*, *i*, *o*, *á*, *í*, *ó*;
6- Remoção dos sufixos *e*, *é*, *ê* e do *ç* (caso a palavra termine com ele);
7- Retorno das vogais nasalizadas à forma original, isto é, transformando ~a e ~o em ã e õ;

2.2.2. Fase de Processamento

Esta fase consiste especificamente na aplicação dos algoritmos e técnicas de Mineração de Textos propriamente ditos, com dois objetivos principais: Recuperação de Informações (RI), que utiliza técnicas para gerar conhecimento a partir de informações contidas em um determinado texto, como clusterização, sumarização e categorização; e a Extração de informações, que utiliza técnicas para retirar conhecimento já explícito no texto, como os mecanismos de busca da web.

2.2.3. Pós-processamento

Consiste na avaliação e validação dos resultados obtidos na fase anterior. Em geral, o principal objetivo dessa etapa é melhorar a compreensão do conhecimento descoberto pelo algoritmo minerador, validando-o através de medidas da qualidade da solução e da percepção de um analista de dados.

2.3. CATEGORIZAÇÃO DE TEXTOS

A categorização de textos (ou classificação de textos) é a atribuição de documentos escritos em linguagem natural a categorias pré-definidas, de acordo com o seu conteúdo (SEBASTIANI, 2002). Apesar do estudo da categorização automática de textos ter iniciado nos anos 60 com Maron e Kuns (1961), a partir da década de 90 que esse campo vem se desenvolvendo, devido ao crescimento do número de documentos disponibilizados em formato digital, viabilizado pelo surgimento da internet, gerando assim, a necessidade de organizá-los para facilitar seu acesso e manuseio.

Hoje em dia, a categorização automática de textos é aplicada em vários contextos, desde a indexação automática ou semiautomática de textos (SIMPSON et al., 2009) até filtros de spam (ALMEIDA; YAMAKAMI; ALMEIDA, 2010) e detecção de conteúdo adulto (ZHANG; QIN; YAN, 2006).

Existem duas principais abordagens para a categorização de textos: uma é conhecida como engenharia do conhecimento (*knowledge engineering*), onde o próprio especialista codifica o sistema através de regras que definem cada categoria da coleção de documentos, como a que foi utilizada no desenvolvimento da ferramenta CADWeb (CADWeb, 2012) por Gomes e Moraes Filho (2011); e outra, utilizada neste trabalho, que usa técnicas de aprendizagem de máquina. Nessa abordagem, o classificador é construído automaticamente, aprendendo as propriedades das categorias a partir de um conjunto de documentos de treinamento previamente classificados (FELDMAN; SANGER, 2007). No conceito de aprendizagem de máquina, esse processo é chamado de aprendizado supervisionado.

Segundo Sebastiani (2002), a vantagem dessa abordagem é a precisão comparável às atingidas pelos especialistas com consideráveis economias em termos de mão-de-obra, uma vez que não existe a necessidade de intervenção humana para a construção do classificador ou adaptação para outro domínio de conhecimento.

Existem diversos algoritmos utilizados na tarefa de categorização de textos e este trabalho utiliza três dos principais: *Naive Bayes*, *k-Nearest Neighbor* (k-NN) e *Support Vector Machines* (SVM), trata-se de algoritmos com resultados comprovadamente satisfatórios, que utilizam métodos distintos para abordar o problema de categorização (YANG; LIU, 1999). Combinando os resultados dos métodos citados, propõe-se um método de grupo. Nas seções seguintes é dada uma descrição do funcionamento de cada um dos métodos tradicionais (*Naive Bayes*, k-NN e SVM) e a proposta do método de grupo é descrita no capítulo 3.

2.3.1. Naive Bayes

O *Naive Bayes* é um categorizador probabilístico, baseado no teorema de Bayes, definido na equação (2). Esse tipo de classificador computa a probabilidade de um documento $\vec{d}$ pertencer à classe c_i, assumindo que a pre-

sença de um termo em uma categoria não está condicionada a presença de qualquer outro. Devido à independência dos termos, apenas as variações para cada classe necessitam ser determinadas, e não a matriz de covariância completa (ZHANG, 2004). Segundo Domingos e Pazzani (1997), a independência de termos na maioria dos casos não prejudica a eficiência do categorizador.

$$P(c_i \mid \vec{d}) = P(c_i)\frac{P(\vec{d} \mid c_i)}{P(\vec{d})} \tag{2}$$

Para ilustrar o funcionamento deste categorizador e facilitar o seu entendimento, considere duas categorias: Tecnologia (*T*) e Esporte (*E*), cinco documentos de exemplo e um documento para teste, cada um contendo apenas uma frase, conforme Tabela 2, sendo w_k os termos válidos (em negrito) após a remoção das *stopwords* (as outras etapas de pré-processamento foram ignoradas para simplificar o exemplo).

Tabela 2 - Documentos utilizados no exemplo de funcionamento do método Naive Bayes e suas respectivas categorias.

Documento	Categoria
Brasil é o **terceiro** em **usuários** do **facebook.**	Tecnologia
O **Flamengo** é o **melhor time** de **futebol** do **campeonato**.	Esporte
Hackers invadem site da **Amazon.**	Tecnologia
Brasil vence mais uma no **campeonato** de **vôlei.**	Esporte
Site da **Amazon apresenta novo kindle.**	Tecnologia
Hackers invadem o **site** do **Flamengo**.	?

Na Tabela 3, tem-se o número de ocorrências de cada termo nas categorias Tecnologia e Esporte.

Tabela 3 - Número de ocorrências de cada termo nas categorias Esporte e Tecnologia.

Termo	Esporte	Tecnologia
brasil	1	1
terceiro	0	1
usuários	0	1
facebook	0	1
flamengo	1	0
melhor	1	0
time	1	0
futebol	1	0
campeonato	2	0
hackers	0	1
invadem	0	1
site	0	2
amazon	0	2
vence	1	0
vôlei	1	0
apresenta	0	1
novo	0	1
Kindle	0	1

Como o exemplo é composto apenas de duas categorias, considera-se a priori que $P(T) = P(E) = 0{,}5$. Generalizando, $P(c_i) = 0{,}5$.

A probabilidade de um termo w_k pertencer a uma determinada categoria c_i é determinada pela equação (3).

$$P(w_k \mid c_i) = \frac{P(c_i) + ocorrências\_de\_w_k\_em\_c_i}{1 + número\_de\_documentos\_em\_c_i} \tag{3}$$

Ao substituir o valor de $P(c_i)$, tem-se a equação (4):

$$P(w_k \mid c_i) = \frac{(0{,}5 + ocorrências\_de\_w_k\_em\_c_i)}{(1 + número\_de\_documentos\_em\_c_i)} \tag{4}$$

Após o cálculo de cada termo, tem-se o resultado final com a probabilidade em cada categoria conforme Tabela 4.

Tabela 4 - Probabilidade de cada termo nas categorias Esporte e Tecnologia.

Termo	Esporte	Tecnologia
brasil	0,50	0,38
terceiro	0,17	0,38
usuários	0,17	0,38
facebook	0,17	0,38
flamengo	0,50	0,13
melhor	0,50	0,13
time	0,50	0,13
futebol	0,50	0,13
campeonato	0,83	0,13
hackers	0,17	0,38
invadem	0,17	0,38
site	0,17	0,63
amazon	0,17	0,63
vence	0,50	0,13
vôlei	0,50	0,13
apresenta	0,17	0,38
novo	0,17	0,38
Kindle	0,17	0,38

Com essas informações é possível determinar a categoria do documento 6 da Tabela 2 utilizando o teorema de Bayes, conforme a equação (5). Observa-se que agora $P(c_i)$ passa a representar a probabilidade a posteriori, considerando a evidência do número de documentos pertencentes a cada categoria.

$$P(T \mid \vec{d}) = \underbrace{P(c_i)}_{\text{documentos\_em\_T / total\_de\_documentos} = 0,6} * \underbrace{P(\vec{d} \mid c_i)}_{\text{P(hackers | T) * P(invadem | Tecnologia)* P(site | T)*P(flamengo | Tecnologia)}} / \underbrace{P(\vec{d})}_{\text{não importa}} \quad (5)$$

Com isso,

$$P(T \mid \vec{d}) = 0,6 * 0,38 * 0,38 * 0,63 * 0,13 / P(\vec{d}) = 0,00709581$$

Analogamente,

$$P(E \mid \vec{d}) = 0{,}4 * 0{,}17 * 0{,}17 * 0{,}17 * 0{,}5 \,/\, P(\vec{d}) = 0{,}0009826$$

Então, como $P(T \mid \vec{d}) > P(E \mid \vec{d})$, o documento pertence à categoria Tecnologia.

2.3.2. k-Nearest Neighbor (k-NN)

O k-NN, é a base dos algoritmos conhecidos como preguiçosos (*lazy algorithms*). Ele armazena todo conjunto de treinamento e empenha todo o esforço em direção à generalização indutiva até o momento da classificação (WETTSCHERECK; AHA; MOHRI, 1997). Esse classificador representa cada exemplo como um ponto de dado em um espaço *d*-dimensional, onde *d* é o número de atributos. Dado um exemplo de teste, calcula-se a proximidade (ou similaridade) com o resto dos pontos de dados no conjunto de treinamento usando uma função (TAN; STEINBACH; KUMAR, 2009). Para cada novo exemplo *x*, é computado seus *k* vizinhos mais próximos em um conjunto de exemplos já classificados, de forma a assinalar *x* à classe mais representativa entre os vizinhos. A Figura 5 apresenta o algoritmo do *k* vizinho mais próximo (k-NN).

Algoritmo para encontrar os *K* vizinhos mais próximos

1: ***para i=1*** até número de objetos de dados ***faça***
2: Encontre as distâncias do objeto de índice i até todos os outros objetos.
3: Ordene essas distâncias em ordem crescente, registrando a relação (objeto, distância).
4: ***retorne*** os objetos associados com as primeiras *K* distâncias da lista ordenada.
5: ***fim para***

Figura 5 - Algoritmo para encontrar os *k* vizinhos mais próximos.

Exemplos de função de proximidade são: correlação, distância euclidiana, medida de similaridade de *Jaccard* e medida de similaridade do cosseno. Segundo Feldman e Sanger (2007), a medida de similaridade do cosseno é a medida mais indicada e utilizada em dados de alta dimensionalidade, que é o caso de documentos textuais, lembrando que nesse caso, a dimensão é a

quantidade de termos encontrados em todos os documentos da coleção, chegando facilmente a milhares.

Segundo Tan, Steinbach e Kumar (2009) essa indicação é fundamentada pela capacidade de lidar com dados esparsos, pois embora os documentos tenham normalmente milhares de dezenas de atributos (termos), poucos são diferentes de zero. Assim, a similaridade não deve depender do número de valores “0” compartilhados, já que quaisquer dois documentos provavelmente não conterão muitas das mesmas palavras, e, portanto, se essas correspondências forem consideradas, a maioria dos documentos serão muito semelhantes. Portanto, a medida de semelhança utilizada deve ser capaz de ignorar essas correspondências de forma que não influenciem no resultado.

O valor de similaridade do cosseno encontra-se no intervalo [0,1]. Quanto mais próximo de 1, isto é, quanto menor o ângulo entre os vetores, mais similares são os documentos, uma vez que *cos(0)=1*. O cálculo da similaridade do cosseno é realizado dividindo-se o produto escalar dos vetores representativos dos documentos, pelo produto de seus módulos. Na equação (6), é demonstrado a formula para se calcular a similaridade do cosseno tomando-se como exemplo os documentos *A* e *B* (SEBASTIANI, 2002).

$$similaridade(A,B) = \cos(\theta) = \frac{A \bullet B}{\| A \| \cdot \| B \|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^{\,2}} \sqrt{\sum_{i=1}^{n} B_i^{\,2}}} \qquad (6)$$

Para ilustrar graficamente o seu funcionamento, a Figura 6 representa de forma simplificada o algoritmo k-NN. Na Figura 6(a) o vizinho mais próximo do ponto *x* é um exemplo negativo (vermelho), portanto é atribuída a categoria negativa ao ponto *x*. Se o número de vizinhos for três conforme a Figura 6(c), o ponto *x* é atribuído a classe positiva (azul), pois são dois vizinhos positivos próximos contra um negativo. No caso da figura 6(b), pode-se escolher aleatoriamente uma categoria para atribuir ao ponto *x*.

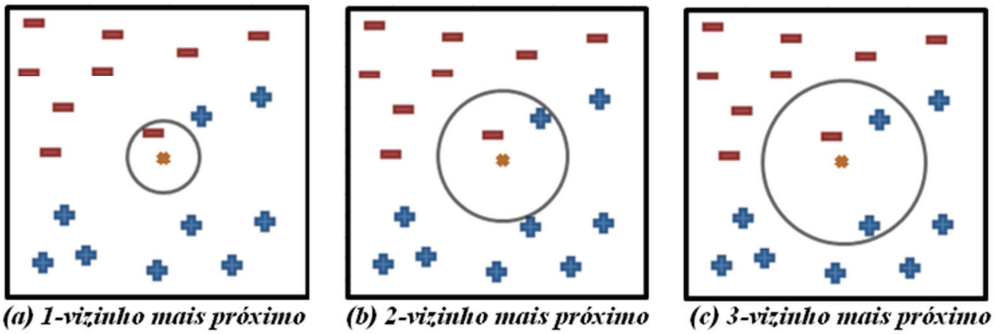

Figura 6 - Os 1, 2 e 3 vizinhos mais próximos de uma instância (TAN; STEINBACH; KUMAR, 2009).

2.3.3. Support Vector Machines (SVM)

O SVM constitui uma técnica da teoria do aprendizado estatístico, baseado no princípio de minimização do risco estrutural introduzido por Vapnik (2000). O objetivo desse algoritmo é encontrar o hiperplano de separação linear ótimo entre duas categorias, maximizando a margem entre seus pontos mais próximos. O hiperplano de categorização é escolhido durante a fase de treinamento como o único que separa as instâncias positivas conhecidas das instâncias negativas com a margem máxima entre elas (FELDMAN; SANGER, 2007). Os exemplos mais próximos do hiperplano são chamados vetores de suporte (*support vectors*). A Figura 7 ilustra esses conceitos apresentando um exemplo de duas categorias linearmente separáveis, onde os vetores de suporte, em cinza, definem a margem de maior separação entre elas.

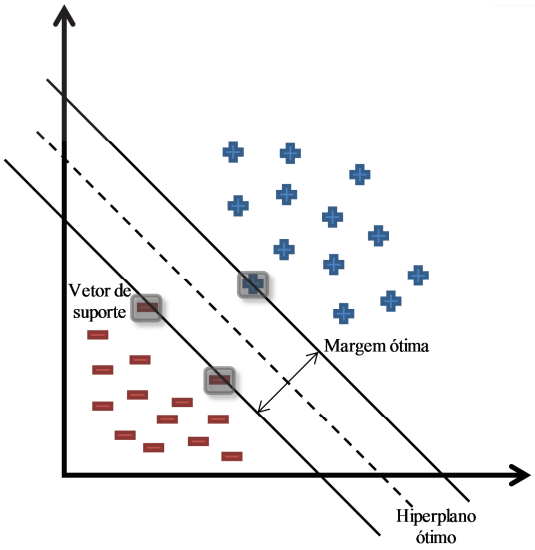

Figura 7 - Categorias separadas linearmente em um espaço bi-dimensional (VAPNIK, 1995).

Para o caso onde existam mais de duas categorias, esse categorizador implementa a abordagem “um contra um”. Isto é, considere *n* é o número de categorias e *x* o ponto de dados que se deseja categorizar. Primeiramente, *n(n-1)/2* categorizadores binários são construídos, onde cada um deles é treinado com os exemplos de duas categorias. Ao final, utiliza-se uma estratégia de votação, onde o resultado de cada categorização binária para o ponto *x* é considerado um voto e o ponto é designado para a categoria com o maior número de votos. (CHANG; LIN, 2011)

Para lidar com casos onde os exemplos de treinamento não são completamente separáveis e um pequeno erro é permitido, utiliza-se o conceito de margens suaves (*soft margins*), que introduz um parâmetro de custo *C*, especificado pelo próprio usuário que determina o nível aceitável de tolerância a erros (BERRY; KOGAN, 2010). Outro parâmetro importante a ser configurado é o critério de parada ε (*epsilon*), para evitar um loop infinito na busca pelo hiperplano ótimo. Neste trabalho, utilizou-se a implementação LIBSVM, criada por Chang e Lin (2011).

3. METODOLOGIA

O software *open source Rapidminer* versão 5.2 com a extensão *Text Processing* versão 5.2.4, criado por Mierswa et al (2006), foi utilizado para implementar todas as técnicas de Mineração de Textos, incluindo os categorizadores, nas etapas experimentais do trabalho. Os principais fatores que motivaram esta escolha foram o fato de possuir todas as implementações dos algoritmos necessários, se tratar de uma ferramenta *open source* e ser totalmente desenvolvido na linguagem de programação *Java*, de forma que todos os objetos utilizados no modelo possam facilmente ser integrados a aplicações desenvolvidas de forma independente. Uma breve descrição das ferramentas de Mineração de Textos disponíveis atualmente pode ser encontrada em Feinerer, Hornik e Meyer (2008). Na Figura 8, tem-se um quadro comparativo com as funcionalidades de cada uma delas.

Produto	Pré-processamento	Associação	Clusterização	Sumarização	Categorização	API
Comercial						
Clearforest	x	x	x	x		
Copernic Summarizer	x			x		
dtSearch	x	x		x		
Insightful Infact	x	x	x	x	x	x
Inxight	x	x	x	x	x	x
SPSS Clementine	x	x	x	x	x	
SAS Text Miner	x	x	x	x	x	
TEMIS	x	x	x	x	x	
WordStat	x	x	x	x	x	
Open Source						
GATE	x	x	x	x	x	x
RapidMiner	x	x	x	x	x	x
Weka/KEA	x	x	x	x	x	x
R/tm	x	x	x	x	x	x

Figura 8 - Quadro comparativo entre as ferramentas de Mineração de Textos pelas suas funcionalidades. (FEINERER; HORNIK; MEYER, 2008).

Utilizou-se neste trabalho, 4336 artigos em língua portuguesa publicados nas edições de 2010 e 2011 do ENEGEP (ABEPRO, 2011) e nas edições de 2007, 2008, 2009, 2010 e 2011 do SIMPEP (SIMPEP, 2011), sendo 3408 para treinamento dos categorizadores e 928 para testes. A Figura 9 apresenta a distribuição dos documentos, considerando os documentos de treinamento e testes nas 11 categorias da Engenharia de Produção. Observa-se que devido a algumas áreas possuírem mais artigos publicados que outras, as categorias não estão balanceadas.

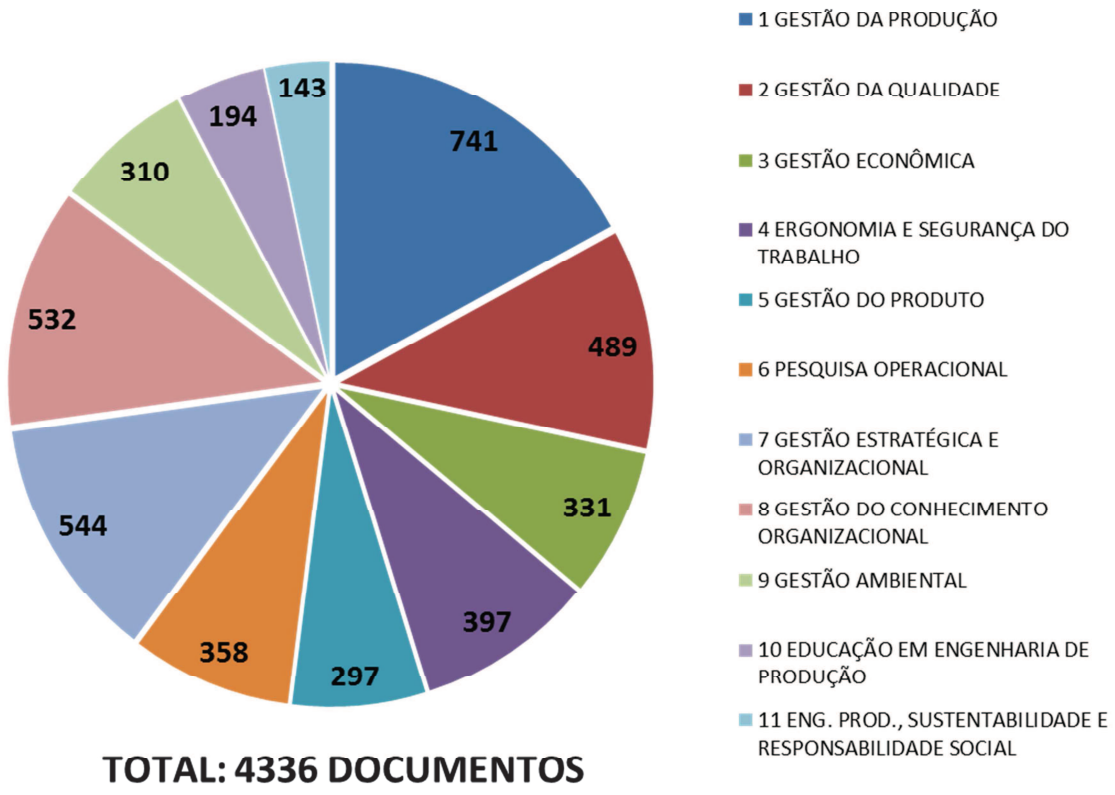

Figura 9 - Distribuição dos 4336 artigos dentre as 11 categorias da Engenharia de Produção.

Na Figura 10, são apresentadas de forma macro, as quatro etapas de Mineração de Textos envolvidas neste trabalho e que serão explicadas a seguir.

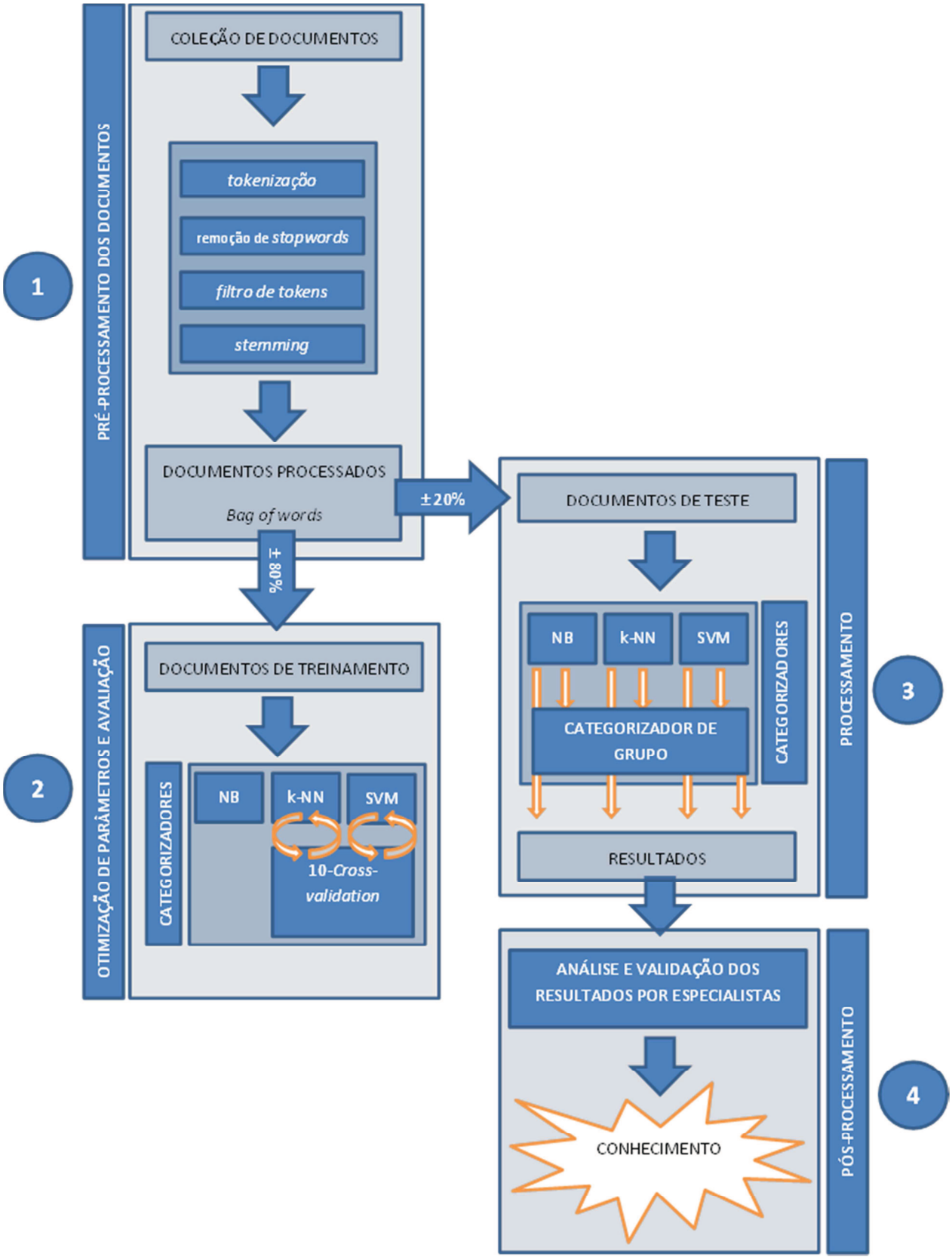

Figura 10 - Etapas de Mineração de Textos utilizadas para categorização dos documentos.

3.1. PRÉ-PROCESSAMENTO DOS DOCUMENTOS

O objetivo dessa etapa é preparar os documentos e representa-los de forma que possam ser processados pelos algoritmos de aprendizagem. Primeiramente todos os documentos adquiridos em formato PDF foram transformados em texto simples, através do software *Some PDF to TXT converter v1.0* (FREE PDF TO TXT CONVERTER, 2011). A escolha desse software deveu-se à gratuidade do mesmo e da funcionalidade de conversão de documentos em lote, essencial devido a grande quantidade de documentos utilizados. A motivação para esta transformação foi o ganho em desempenho na execução do processamento dos documentos, na ordem de vinte vezes aproximadamente. Após essa conversão, se manteve apenas o conteúdo textual dos documentos, figuras e opções de formatação foram automaticamente ignoradas.

Em seguida, foi necessário excluir o tema do congresso nos documentos onde o mesmo foi identificado no corpo do texto, pois os termos que o compõem não representam com fidelidade o conteúdo do documento.

Como os algoritmos de aprendizagem de máquina não são capazes de processar documentos em texto diretamente em seu formato original, durante a etapa de pré-processamento foi realizada a representação dos documentos nos chamados vetores de características (*feature vectors*) na representação *bag of words* (saco de palavras), que utiliza todos os termos do documento como características. Dessa forma, a dimensão do espaço de características é igual ao número de termos diferentes encontrados em todos os documentos. Existem várias formas de atribuir pesos aos termos do documento. Nesse trabalho utilizou-se a frequência do termo (*term frequency*) normalizada *TF* (RAPID-I, 2012), modelada matematicamente na equação (7), pois se mostrou mais eficaz no domínio estudado que a *TF-IDF*, detalhada na equação (1) da seção 2.2.1.1.

$$TF = \frac{TermFreq(w,\vec{d})}{\sqrt{TermFreq\left(w,\vec{d}\right)^2 + TermFreq\left(w+1,\vec{d}\right)^2 + \ldots + TermFreq\left(w+n,\vec{d}\right)^2}} \quad (7)$$

Onde:

$TermFreq(w,\vec{d})$: Frequência do termo no documento;

TF: Frequência do termo normalizada.

Antes de efetivamente gerar o vetor de características para cada documento, cinco processos são executados sequencialmente com o objetivo de reduzir a dimensão do espaço de representação dos documentos:

- Tokenização: Neste processo, os documentos foram divididos em tokens, sendo cada um deles um termo do documento. Neste processo também se deu a transformação dos termos em letras minúsculas, remoção de dígitos, pontuações e caracteres especiais.

- Remoção de *stopwords*: O objetivo deste processo é remover termos que não apresentam um conteúdo semântico significativo no contexto em que se apresentam no documento. Geralmente trata-se de palavras auxiliares ou conectivas (por exemplo: a, de, aos, com), que não fornecem nenhuma informação que venha a representar conteúdo dos documentos. A exclusão se deu com base em um arquivo texto com a lista dos termos (*stoplist*). A lista completa com as *stopwords* utilizadas no trabalho pode ser consultada no APÊNDICE D.

- Filtro de *tokens*: Especifica critérios de eliminação. Neste trabalho, obtiveram-se melhores resultados removendo as palavras com menos de 5 letras e ignorando os termos que ocorreram em menos de 4% e em mais de 99% dos documentos.

- *Stemming*: O objetivo deste processo é a remoção do sufixo e prefixo dos termos que possam vir a representar uma variação verbal ou plural, ge-

rando apenas os radicais de acordo com as regras gramaticais da língua utilizada. Por exemplo: os termos computação, computador e computar são transformados em comput. A principal finalidade desse processo é a redução do espaço dimensional. Neste trabalho, utilizou-se o algoritmo de Porter adaptado para a língua portuguesa na linguagem *snowball*, criada pelo próprio Porter. Informações sobre ao algoritmo podem ser obtidas em Willet (2006) e sobre a linguagem *snowball* em Porter (2011).

A Figura 11 ilustra a sequência de processos executados para geração do vetor de características do modelo *bag of words*.

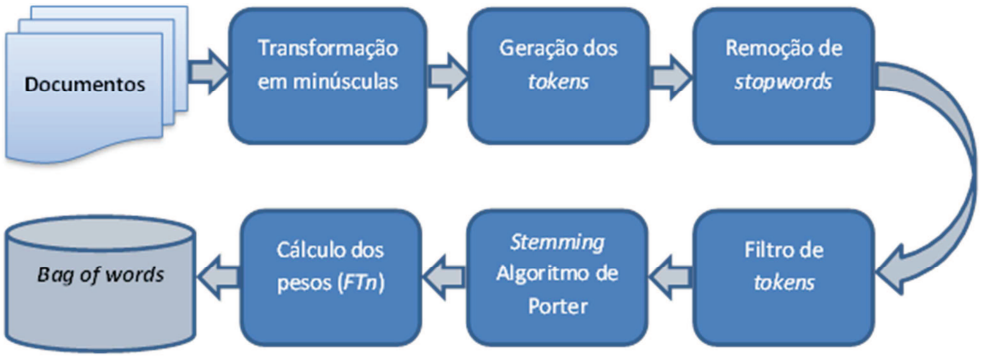

Figura 11 - Etapas do pré-processamento em ordem de execução.

Dessa forma, ao término dessa etapa, obtiveram-se todos os 4336 documentos representados no modelo *bag of words*, com uma redução do espaço dimensional na ordem de 50,3%, isto é, de 6132 para 3044 termos. A Tabela 4 apresenta os dez termos com maior ocorrência no domínio estudado, após a etapa de pré-processamento, tanto no total quanto em número de documentos, com os destaques de cada um em vermelho.

Tabela 5 - Os dez termos com mais ocorrências no total e em número de documentos.

	Stem	Total de ocorrências	Número de documentos
1	**empres**	**87251**	3050
2	**process**	74358	3324
3	**produt**	72509	3110
4	**trabalh**	56148	3341
5	**utiliz**	43564	**3352**
6	**desenvolv**	41688	3269
7	**produçã**	41173	3023
8	**sistem**	39350	3023
9	**pesquis**	38850	3091
10	**estud**	36364	3306

3.2. MEDIDAS DE AVALIAÇÃO

Como o presente trabalho objetiva-se na tarefa de categorização, as métricas utilizadas na avaliação do desempenho dos categorizadores utilizados foram: acurácia (*accuracy*), precisão (*precision*), abrangência (*recall*) e F_1. A acurácia, *a*, é a medida mais básica de eficiência do categorizador, sendo a fração de documentos corretamente categorizados. Abrangência, *r*, é definida como a fração dos documentos de uma categoria corretamente categorizados. Precisão, *p*, é definida como a fração de documentos corretamente categorizados dentre todos os documentos atribuídos pelo categorizador a uma categoria. Portanto, uma abrangência perfeita é alcançada caso todos os documentos da categoria em questão sejam nela categorizados, independentemente se outros documentos de outras categorias sejam também atribuídos a ela. Por outro lado, uma boa precisão é alcançada ao evitar que documentos provenientes de diferentes categorias sejam atribuídos a uma só. Em virtude da variedade de aspectos de avaliação, uma abordagem mais usual para avaliar o desempenho da categorização é F_1, uma combinação entre precisão e abrangência, dada pela média harmônica dessas duas métricas (ZELAIA; ALEGRIA, 2011).

As equações (8), (9), (10) e (11) definem as métricas citadas anteriormente:

$$a = \frac{DC}{TD} \tag{8}$$

$$p = \frac{VP}{VP + FP} \tag{9}$$

$$r = \frac{VP}{VP + FN} \tag{10}$$

$$F_1 = \frac{2}{\left(\left(\frac{1}{p}\right) + \left(\frac{1}{r}\right)\right)} \tag{11}$$

Onde:

DC : Documentos corretamente categorizados;

TD: Total de documentos;

VP: Verdadeiro-positivos;

FP: Falso-positivos;

FN: Falso-negativos.

3.3. OTIMIZAÇÃO DE PARÂMETROS E AVALIAÇÃO PRELIMINAR DOS ALGORITMOS

Com os documentos devidamente representados no modelo *bag of words*, o próximo passo foi utilizar os 3408 documentos separados inicialmente para otimização dos parâmetros e treinamento dos algoritmos de forma a maximizar a acurácia, estimando assim o desempenho do categorizador quando apresentados ao conjunto de teste. A busca pelos parâmetros ótimos deu-se de forma empírica utilizando duas técnicas em conjunto para auxiliar neste processo: o *grid search*, onde se varia automaticamente um parâmetro dentro de uma faixa pré-estabelecida de valores incrementada por alguma

função em uma busca exaustiva pelo seu valor ótimo; e a validação cruzada, que consiste em dividir o conjunto de treinamento em *x* subconjuntos de tamanhos iguais, testando sequencialmente cada subconjunto no categorizador treinado com os elementos dos subconjuntos *x*-1 restantes (HSU; CHANG; LIN, 2010). Neste trabalho realizou-se o *grid search* utilizando validação cruzada com *x* = 10 e a acurácia obtida foi armazenada. Esse processo foi utilizado na escolha do valor de *k* (número de vizinhos) para o algoritmo k-NN, e do valor de *C* (nível de tolerância a erros) e ε (critério de parada) para o SVM. O categorizador *Naive Bayes* não necessita de customização de parâmetros.

Para o algoritmo k-NN, o objetivo foi encontrar o valor de *k* que maximizasse a acurácia do modelo. Para isso, realizou-se uma busca por força bruta com 50 valores de *k* em escala logarítmica na faixa de 1 a 100 utilizando a validação cruzada com *x* = 10 para avaliação de cada iteração. Ao final do processo, chegou-se ao número de 23 vizinhos como sendo este o que forneceu a maior acurácia ao modelo. A Figura 12 demonstra graficamente o resultado desse processo. A tabela com o resultado completo deste processo de busca pode ser consultado no APÊNDICE A.

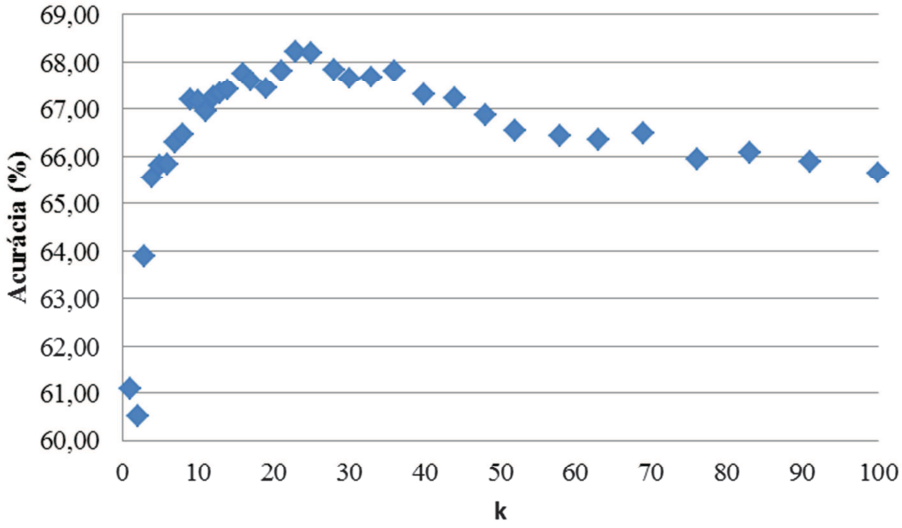

Figura 12 - Resultado do processo de busca pelo valor de k do algoritmo k-NN.

Diferentemente do processo utilizado para encontrar o melhor valor de *k*, do categorizador k-NN, para o categorizador SVM não se utilizou todo o conjunto de treinamento devido ao alto custo computacional e tempo exigido

para a conclusão do processo. Por esta razão, optou-se por utilizar 10% do conjunto de treinamento, tomando-se documentos aleatórios e respeitando o balanceamento entre as categorias. Conforme sugerido por Hsu, Chang e Lin, 2010, os valores de C e ε variaram exponencialmente da seguinte forma: $C = \{2^{-5}, 2^{-3}, ..., 2^{9}\}$ e $\varepsilon = \{2^{-15}, 2^{-13}, ..., 2^{1}\}$. Apesar do resultado da busca apontar para os valores $C = 2$ e $\varepsilon = 0{,}00003$, quando se utilizou todo o conjunto de treinamento, a acurácia foi menor que a obtida com os valores padrão ($C = 0$ e $\varepsilon = 0{,}001$), portanto, esses últimos foram utilizados. A tabela com o resultado completo deste processo de busca pode ser consultado no APÊNDICE B.

Os resultados da métrica F_1 de cada categorizador obtidos nessa etapa foram armazenados para serem utilizados pelo método de grupo que será posteriormente detalhado.

Cabe ressaltar que caso haja inclusão ou exclusão de documentos no conjunto de treinamento, com o objetivo de aumentar a inteligência do sistema, toda etapa de otimização de parâmetros e avaliação preliminar dos algoritmos deverá ser executado novamente, primeiramente para constatar se de fato os novos documentos melhoram o desempenho dos algoritmos nas métricas definidas e depois para que os pesos utilizados pelo método de grupo sejam atualizados.

3.4. GERAÇÃO DOS MODELOS DE CATEGORIZAÇÃO

Superada a etapa de otimização e avaliação preliminar dos algoritmos, gerou-se o modelo de categorização, isto é, o categorizador propriamente dito de cada um deles com os parâmetros ótimos encontrados e com os 3408 documentos de treinamento servindo como base de aprendizagem. Os categorizadores gerados neste processo foram utilizados para categorizar os 928 documentos separados para teste. O fluxo apresentado na Figura 13 ilustra esse processo.

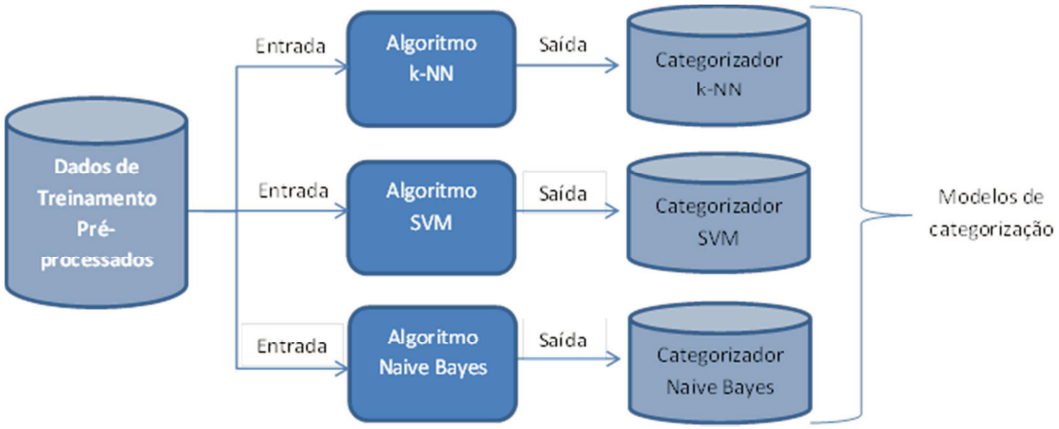

Figura 13 - Fluxo de geração dos modelos de categorização k-NN, SVM e *Naive Bayes*.

Observa-se que o mesmo conjunto de treinamento é utilizado para todos os categorizadores.

3.5. MÉTODO DE GRUPO

O objetivo desta técnica é melhorar o desempenho da categorização agregando a previsão de múltiplos categorizadores. Agregar diferentes categorizadores consiste em aplica-los na mesma tarefa de categorização combinando suas saídas apropriadamente (ZELAIA; ALEGRIA, 2011). Segundo Feldman e Sanger (2007), para obter bons resultados, os classificadores devem ser significativamente diferentes, seja na representação dos documentos ou no método de aprendizagem. Neste trabalho, é proposto um método de grupo utilizando os métodos *Naive Bayes*, k-NN e SVM, descritos nas Seções 2.3.1, 2.3.2, 2.3.3 e 2.3.4, respectivamente.

Em cada método, gera-se o valor de confiança (*confidence*), *Conf*, para cada par $(\vec{d}, c_i)$, sendo $\vec{d}$, o documento e c_i, a categoria. A confiança, também chamada de valor de status de categorização (*categorization status value)* (FELDMAN; SANGER, 2007), é um valor normalizado dentro do intervalo [0,1] que representa o nível de pertinência de um dado documento $\vec{d}$ à categoria c_i. Em outras palavras, a probabilidade de um documento $\vec{d}$ pertencer à categoria c_i, segundo um categorizador em particular. Cada categorizador,

isoladamente, determina como categoria de um documento sempre aquela que obtiver o maior valor de confiança dentre as opções existentes (ZELAIA; ALEGRIA, 2011). Como esse valor é calculado de formas diferentes para cada categorizador, iremos descrever abaixo cada um deles:

- *Naive Bayes*: A confiança é própria probabilidade calculada diretamente pelo algoritmo para cada par $(\vec{d}, c_i)$.

- k-NN: No classificador k-NN, a confiança é dada da conforme equação (12).

$$Conf(\vec{d}, c_i) = \frac{k'}{k} \tag{12}$$

Onde:

k: Número de vizinhos considerados pelo algoritmo para aferir a categoria do documento $\vec{d}$. Dentre as categorias dos *k* vizinhos, aquela que obtiver a maior representatividade atribui à categoria do documento.

k´: Dentre os *k* vizinhos mais próximos de $\vec{d}$, aqueles que pertencem a categoria c_i.

- SVM: Para o categorizador SVM, a confiança é calculada através de uma implementação aprimorada da probabilidade a posteriori de Platt (PLATT, 2000) através da função *svm_predict_probability()* da biblioteca *libsvm*, que retorna uma lista com os valores de confiança para cada categoria. Detalhes podem ser encontrados em Chang e Lin (2011).

3.5.1. Funcionamento

Este trabalho propõe os seguintes passos para realizar a categorização de documentos utilizando-se do conhecimento adquirido pelos categorizadores SVM, k-NN e *Naive Bayes*, de forma a agrega-los maximizando o desempenho final de categorização:

1. Para cada um dos 3 métodos, gerar o valor de confiança para cada par $(\vec{d}, c_i)$;

2. A confiança obtida em cada categoria, é multiplicada por um peso p_m, igual à média da métrica F_1 de cada categorizador normalizada, obtida na etapa de otimização de parâmetros e avaliação preliminar (Seção 3.3). A equação (13) demonstra como os pesos são calculados. Observe que *m* varia de 1 a 3 devido ao número de métodos utilizados (*Naive Bayes*, k-NN e SVM);

$$p_m = \frac{(F_1, c_i)}{\sum_{m=1}^{3} (F_1, c_i)} \tag{13}$$

Onde:

(F_1, c_i): Valor da métrica F_1 do método *m* para a categoria c_i

3. A categoria atribuída ao documento é aquela que obtiver a maior pontuação, isto é, a soma das confianças, multiplicada pelo peso p_m calculado anteriormente para cada um os três métodos utilizados. A equação (14) demonstra matematicamente o cálculo da pontuação C_i de cada categoria c_i e a equação (15), a atribuição da categoria que obtiver maior pontuação ao documento $\vec{d}$, sendo *n*, o número total de categorias possíveis. No caso deste trabalho, são 11 categorias, que representam as 11 áreas principais de publicação na área de Engenharia de Produção.

$$C_i = \sum_{m=1}^{3} \left(Conf_m(\vec{d}, c_i) \cdot p_m \right) \tag{14}$$

$$c(\vec{d}) \leftarrow \max\left(C_1, C_2, ..., C_n\right) \tag{15}$$

A Figura 14 ilustra graficamente o funcionamento do método de grupo.

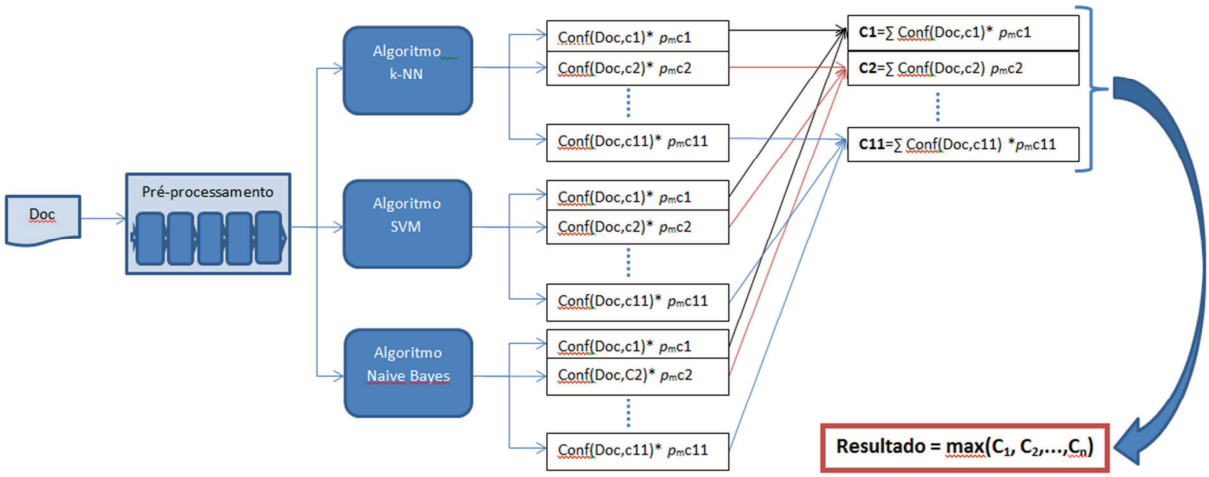

Figura 14 – Funcionamento do método de grupo.

3.6. TESTES

Na etapa de testes, realizou-se a categorização propriamente dita, onde os 928 documentos separados para essa finalidade foram submetidos aos três categorizadores, e o resultado, incluindo a confiança para cada par $(\vec{d}, c_i)$, gravado em um arquivo no formato CSV, de forma a ser utilizada como entrada para o método de grupo. Vale ressaltar que esses documentos não foram utilizados na etapa de otimização e avaliação preliminar dos algoritmos. A Figura 15 ilustra graficamente o fluxo descrito anteriormente.

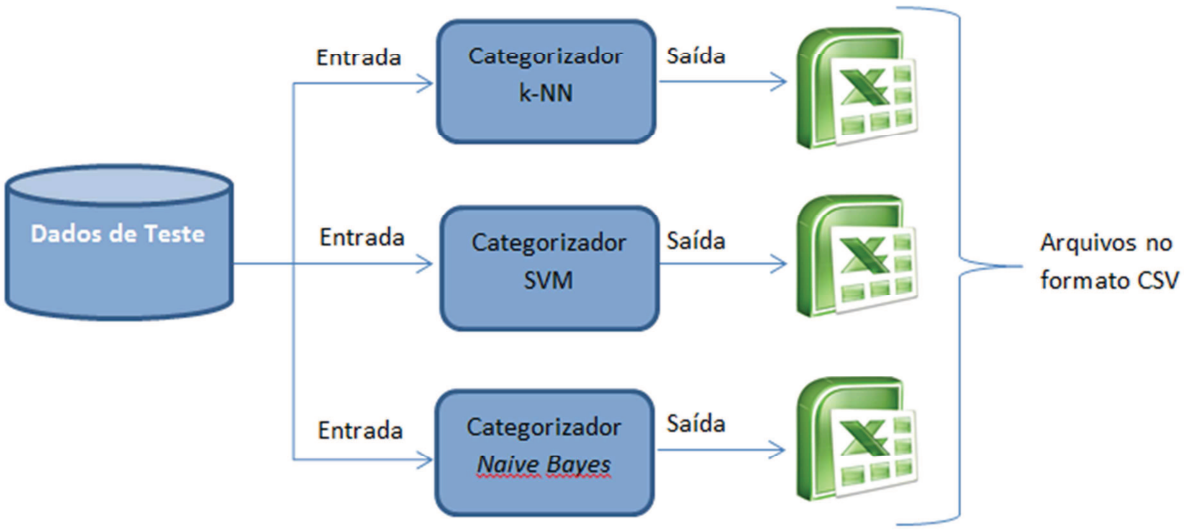

Figura 15–Fluxo de categorização dos dados de teste com os resultados armazenados em arquivo CSV.

Além do experimento principal, que é o teste de categorização, realizaram-se dois experimentos complementares com o objetivo de servir como insumo de avaliação do desempenho dos categorizadores:

- **Experimento 1:** Medir o grau de similaridade entre os documentos das onze categorias da Engenharia de Produção utilizados nos testes deste trabalho, utilizando a medida de similaridade do cosseno. Neste experimento, utilizaram-se os 928 documentos de teste, sendo realizado, um total de 430128 medições.
- **Experimento 2:** Estimar o desempenho dos classificadores Naive Bayes, k-NN e SVM com a mesma metodologia utilizada na etapa de otimização de parâmetros e avaliação preliminar dos algoritmos em uma coleção composta por 87 documentos no formato PDF obtidos em anais de congressos de cada uma das seguintes áreas do conhecimento: Direito (CONPEDI, 2012), odontologia (COB, 2012) e Veterinária (COMBRAVET, 2012), sendo 29 documentos de cada uma delas. Segundo Feldman e Sanger (2007), uma regra prática para pequenos experimentos, é utilizar 30 exemplos de cada categoria. No presente caso, 1 documento de cada categoria foi descartado por estar protegido, o que impede o software de acesso ao seu conteúdo.

4. RESULTADOS E DISCUSSÕES

Esta seção apresenta os resultados obtidos desde a etapa de otimização de parâmetros e avaliação preliminar até os resultados de cada categorizador quando confrontados com o conjunto de teste, incluindo o método de grupo proposto, bem como os resultados dos dois experimentos descritos na seção anterior.

Na Figura16 é apresentado o resultado da acurácia alcançado pelos categorizadores SVM, k-NN e *Naive Bayes* após etapa de otimização de parâmetros e avaliação preliminar, utilizando o conjunto de treinamento.

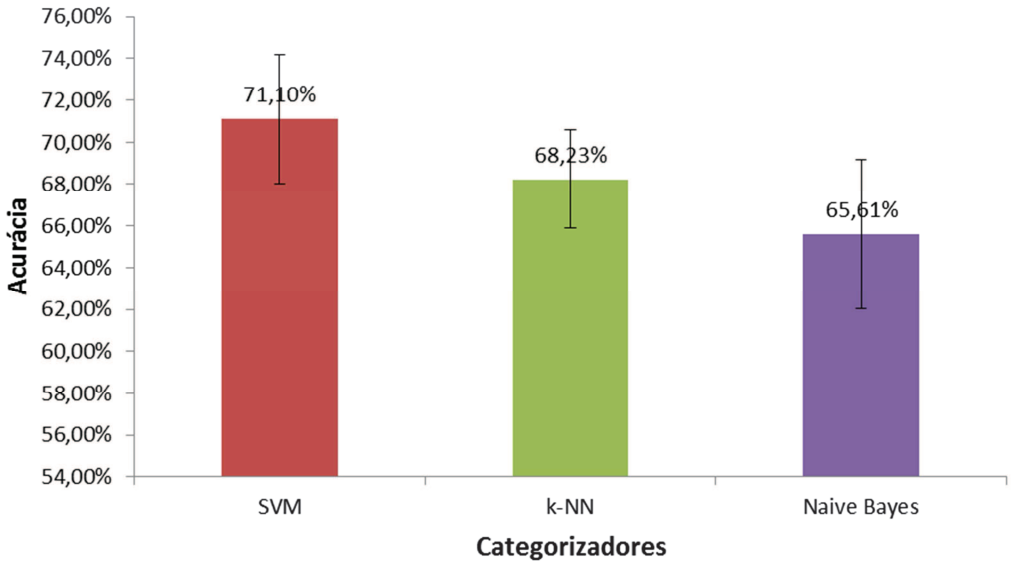

Figura 16 - Acurácia dos categorizadores SVM, k-NN e *Naive Bayes* na etapa de Otimização de parâmetros e avaliação preliminar.

Na Figura 17, tem-se o resultado da média da métrica F_1 alcançado pelos categorizadores SVM, k-NN e *Naive Bayes* após etapa de otimização de parâmetros e avaliação preliminar, utilizando o conjunto de treinamento.

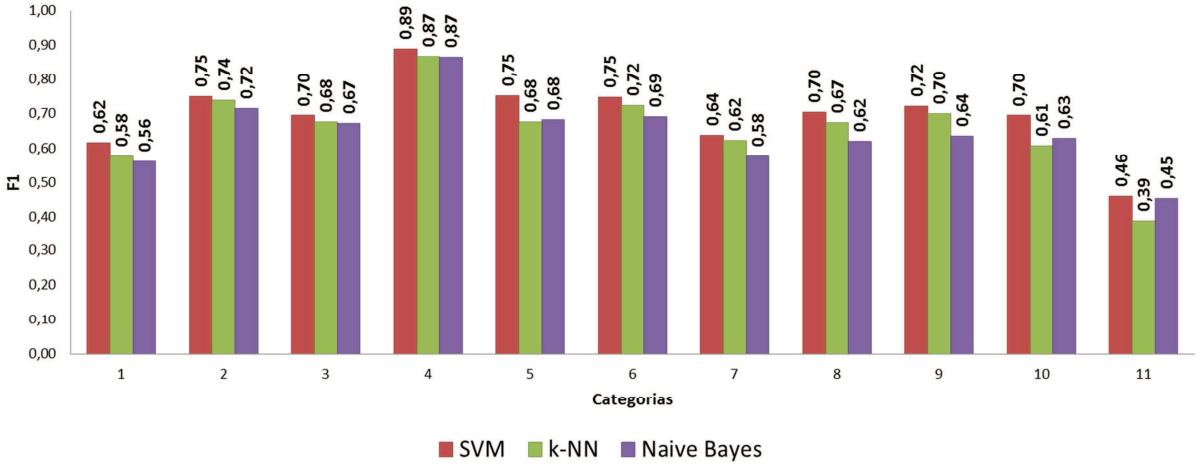

Figura 17 – Média da métrica F_1 dos categorizadores SVM, k-NN e Naive Bayes obtida na etapa de Otimização de parâmetros e avaliação preliminar.

De acordo com os resultados obtidos nessa etapa, o classificador SVM obteve melhor média de acurácia, com 71,10%, o k-NN foi o segundo colocado com 68,12%, com o menor desvio-padrão entre os três e o *Naive Bayes* o terceiro com 65,61%. Esse ranking se mantém quando analisados os resultados da média da métrica F_1, com exceção das categorias 10 (Educação em Engenharia de Produção) e 11 (Eng. Prod., Sustentabilidade e Responsabilidade Social), onde o categorizador *Naive Bayes* obteve um desempenho superior ao k-NN e na categoria 4 (Ergonomia e Segurança do Trabalho), onde os mesmos ficaram empatados com 0,87.

Com os valores da média da métrica F_1 obtida pelos categorizadores em cada categoria, definiram-se os pesos p_m apresentados na Tabela 6 aplicando a equação (13) da seção 3.5.1, que foram utilizados pelo método de grupo na etapa de testes.

Tabela 6 - Pesos obtidos utilizando a técnica de método de grupo.

CATEGORIAS	Pesos		
	SVM	**Naive Bayes**	**k-NN**
1 GESTÃO DA PRODUÇÃO	0,352	0,318	0,330
2 GESTÃO DA QUALIDADE	0,339	0,326	0,335
3 GESTÃO ECONÔMICA	0,341	0,327	0,332
4 ERGONOMIA E SEGURANÇA DO TRABALHO	0,338	0,331	0,331
5 GESTÃO DO PRODUTO	0,355	0,322	0,322
6 PESQUISA OPERACIONAL	0,347	0,319	0,333
7 GESTÃO ESTRATÉGICA E ORGANIZACIONAL	0,348	0,315	0,337
8 GESTÃO DO CONHECIMENTO ORGANIZACIONAL	0,352	0,312	0,337
9 GESTÃO AMBIENTAL	0,350	0,311	0,340
10 EDUCAÇÃO EM ENGENHARIA DE PRODUÇÃO	0,361	0,325	0,314
11 ENG. PROD., SUSTENTABILIDADE E RESPONSABILIDADE SOCIAL	0,354	0,346	0,300

No APÊNDICE C pode ser consultada a tabela com o resultado da etapa de Otimização de parâmetros e avaliação preliminar consolidados, incluindo a precisão e abrangência por categoria.

Na etapa de testes, com os dados apresentados na Figura 18, observa-se que houve uma queda na acurácia dos categorizadores SVM, k-NN e *Naive Bayes* quando comparado ao desempenho esperado obtido na etapa de otimização de parâmetros e avaliação preliminar (Figura 16). O categorizador SVM, apesar de manter-se como o de melhor desempenho dentre os três citados anteriormente nesta métrica, apresentou a maior queda entre as etapas, na ordem de 7,04% com relação à média obtida na etapa anterior. O desempenho do categorizador k-NN manteve-se praticamente constante apresentando a menor variação entre o estimado e o real, e o categorizador *Naive Bayes* permaneceu na ultima posição com o pior resultado.

O método de grupo proposto obteve a melhor acurácia (71,1%), superando o desempenho individual dos categorizadores SVM, k-NN e *Naive Bayes*, tendo um desempenho 7,57% melhor que o segundo colocado.

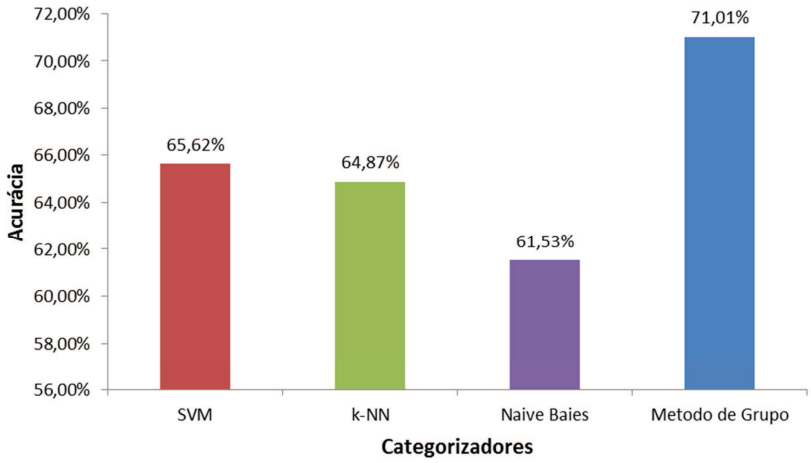

Figura 18 - Acurácia dos categorizadores SVM, k-NN e *Naive Bayes* na etapa de testes.

Avaliando o desempenho dos categorizadores segundo a métrica F_1, através da Figura 19, observa-se que o desempenho do método de grupo foi bastante satisfatório, sendo superior aos demais em praticamente todas as categorias, com exceção da categoria 11 (Eng. Prod., Sustentabilidade e Responsabilidade Social), onde obteve desempenho apenas 1 ponto percentual abaixo do categorizador *Naive Bayes*. Além disso, mais de 50% das categorias obtiveram um valor de F_1 acima de 0,70. A categoria 4 (Ergonomia e Segurança do Trabalho) obteve melhor valor de F_1, isto é, tem o melhor desempenho de classificação combinando a precisão e a abrangência (0,86).

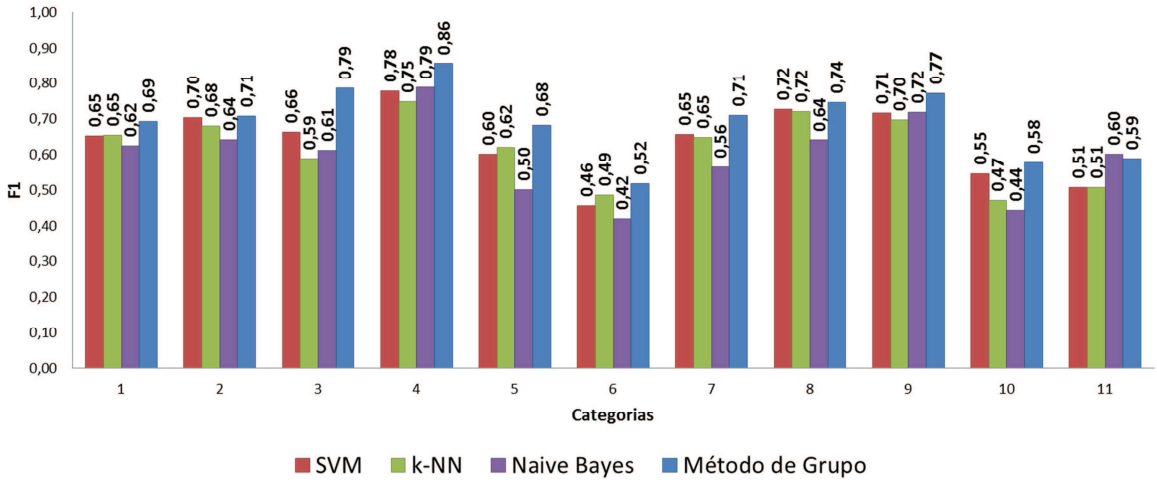

Figura 19 - Métrica F_1 dos categorizadores SVM, k-NN, Naive Bayes e o método de grupo na etapa de testes.

Nas figuras 20, 21, 22 e 23, temos os gráficos com as medidas de abrangência e precisão para os categorizadores: SVM, k-NN, *Naive Bayes* e método de grupo, respectivamente.

De forma geral, observando os resultados dos gráficos, o método de grupo apresenta desempenho inferior em apenas 6 das 88 medições de precisão e abrangência, isto é, em menos de 7%, e, com exceção da categoria 11 (Eng. Prod., Sustentabilidade e Responsabilidade Social), o valor da outra medida que compõe o F_1, seja abrangência ou precisão, compensa de forma a superar o desempenho individual dos categorizadores nesta métrica.

A categoria 4 (Ergonomia e Segurança do Trabalho), foi a categoria que atingiu o maior nível de abrangência no método de grupo, com 98% dos documentos pertencentes à ela corretamente categorizados. Na prática isso se traduz em um alto número de verdadeiro-positivos. Pode-se afirmar, que os documentos pertencentes a essa categoria, possuem uma grande quantidade de termos que pesam em sua representação de forma a diferenciá-la das demais.

A categoria que atingiu o maior nível de precisão foi a categoria 1 (Gestão da Produção), também no método de grupo, com 0,88. Isto representa um baixo número de falso-positivos.

Considerando apenas o método de grupo (Figura 23), a categoria 11 (Eng. Prod., Sustentabilidade e Responsabilidade Social) demonstrou-se como a de menor abrangência (0,46) e a 10 (Educação em Engenharia de Produção) de menor precisão (0,52).

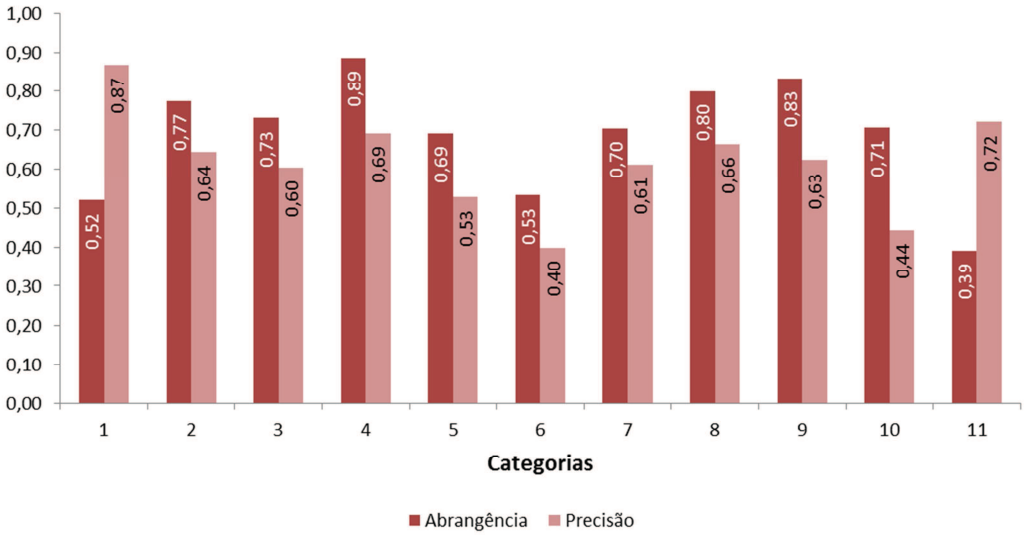

Figura 20 - Métricas Abrangência e Precisão do categorizador SVM.

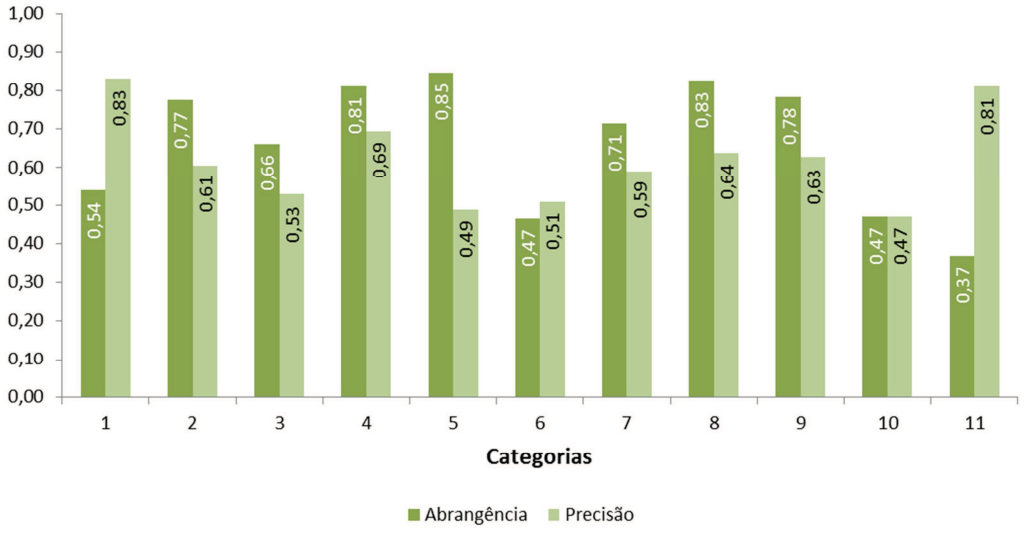

Figura 21 - Métricas Abrangência e Precisão do categorizador k-NN.

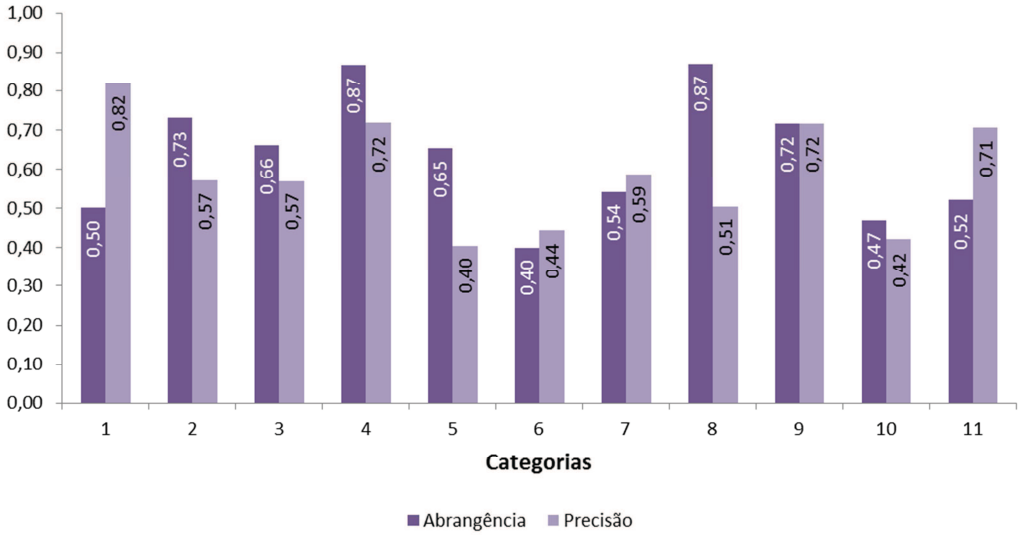

Figura 22 - Métricas Abrangência e Precisão do categorizador *Naive Bayes*.

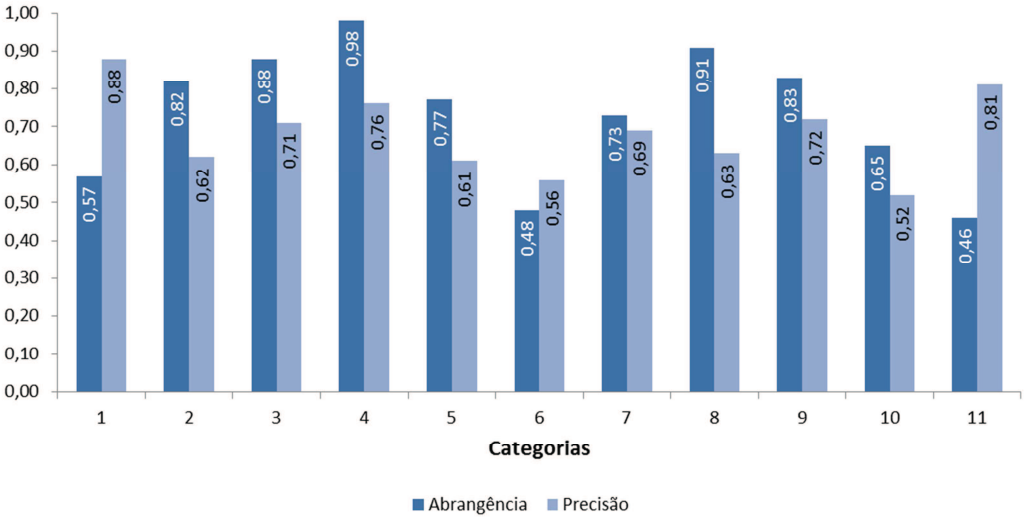

Figura 23 - Métricas Abrangência e Precisão do método de grupo na etapa de testes.

Através dos gráficos das próximas figuras, provenientes do Experimento 1 realizado para medir o grau de similaridade do cosseno dos 928 documentos utilizados na etapa de testes é possível fazer algumas observações baseadas em análises estatísticas desse conjunto de dados.

Primeiramente, através do histograma geral de frequência da similaridade apresentado na Figura 24, baseado em todas as medições realizadas, observa-se que se trata de uma distribuição mais alta e concentrada que uma distribuição normal (curtose > 0), com a grande maioria das observações dentro de uma pequena faixa de valores e com a cauda direita mais pesada (as-

simetria > 0), com isso conclui-se que é relativamente fácil obter-se valores que se afastam positivamente da média a vários múltiplos do desvio padrão.

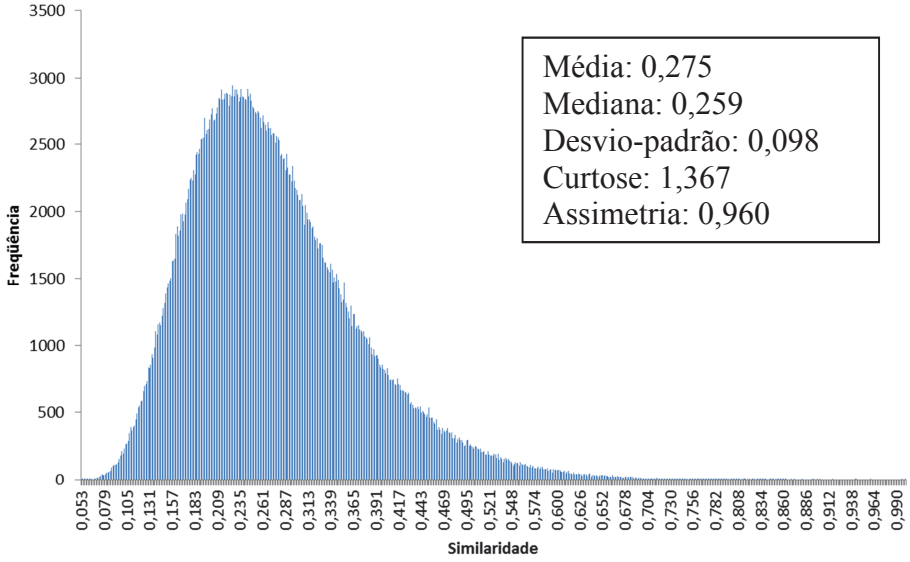

Figura 24 - Histograma de frequência da similaridade entre os 928 documentos de testes das 11 categorias da Engenharia de Produção.

Através do gráfico da Figura 25, constata-se que apesar do grau de similaridade entre os documentos de categorias distintas não ser muito elevado, este cenário não se altera significativamente para documentos pertencentes à mesma categoria. Inclusive, observando o desvio-padrão, verifica-se uma grande quantidade de documentos apresentarem graus de similaridade com documentos de outras categorias equivalentes a similaridade com os demais documentos de sua categoria. Isso justifica a dificuldade de separação dos documentos, acrescido o fato de cada uma das 11 áreas possuírem de 3 a 8 subáreas, o que contribui para uma diversidade dentro de cada categoria da coleção.

Também através da Figura 25 observa-se que a categoria 6 (Pesquisa Operacional), que possui o pior desempenho na métrica F_1 do método de grupo, apresenta o menor grau de similaridade entre documentos, tanto pertencentes a própria categoria quanto a outras categorias, e a menor diferença entre eles, mesmo quando observado o máximo do desvio-padrão.

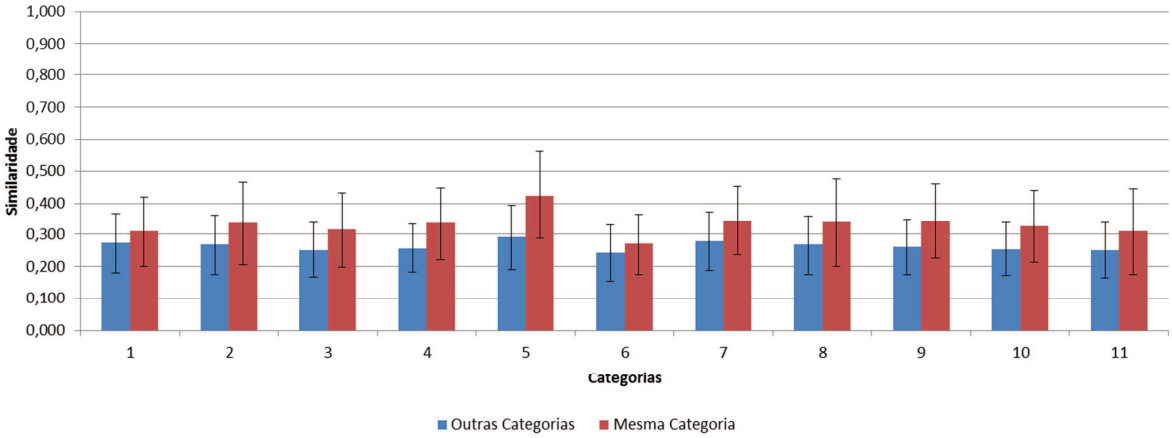

Figura 25 - Médias e desvio-padrão da similaridade entre documentos das 11 categorias em relação a documentos de outras categorias e documentos da mesma categoria.

As Figuras 26 e 27 ilustram o comportamento de similaridade dos documentos através de histograma seguindo a mesma linha da Figura 25, isto é, comparando os documentos entre categorias e dentro da mesma categoria. Observa-se que de forma geral a curva não se altera significativamente quanto a sua forma e amplitude em praticamente todas as categorias, apenas com um deslocamento positivo do mínimo e máximo no eixo de similaridade.

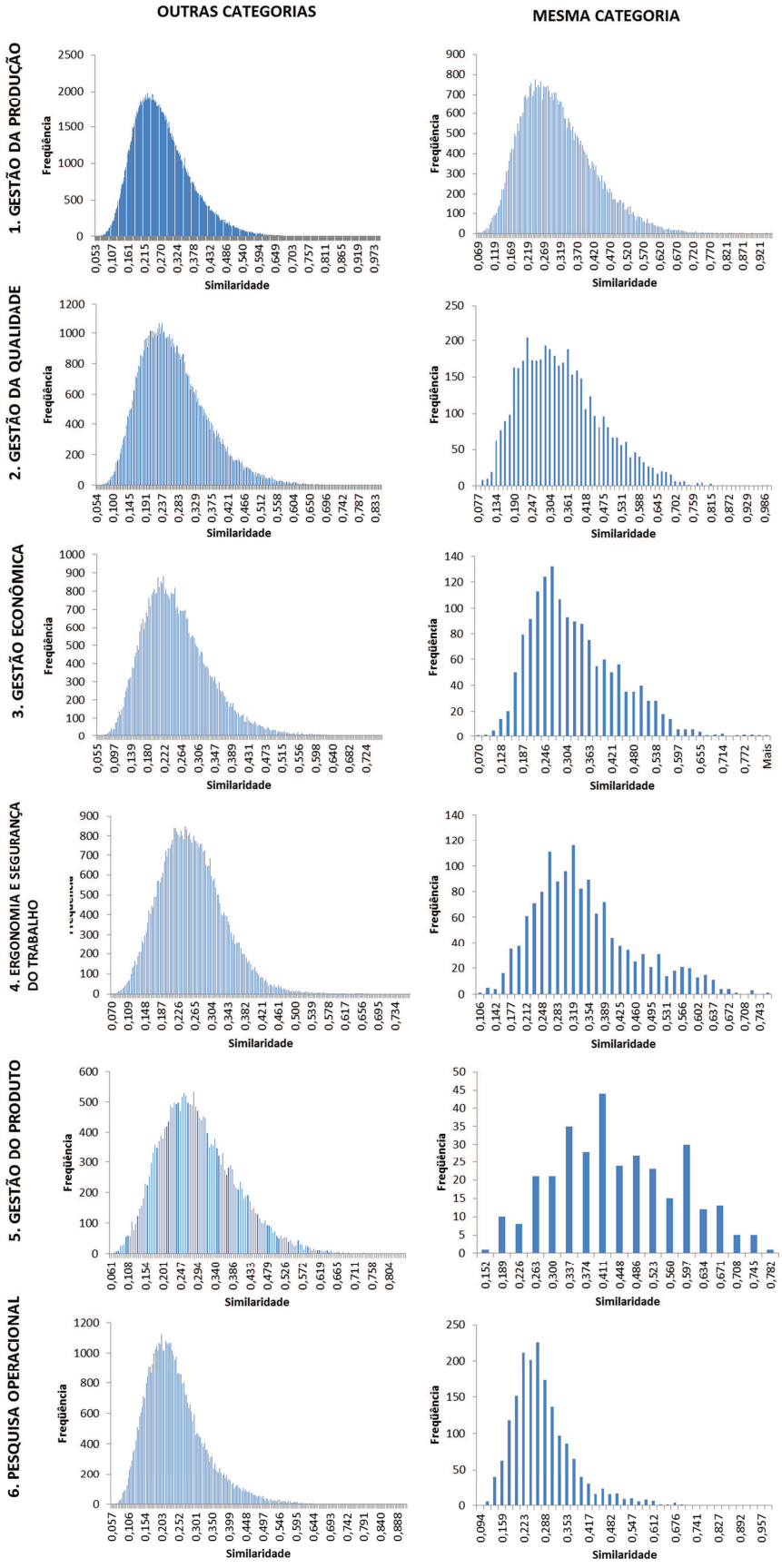

Figura 26 – Histograma de frequência da similaridade das categorias 1 a 6, considerando documentos de outras categorias e documentos da mesma categoria.

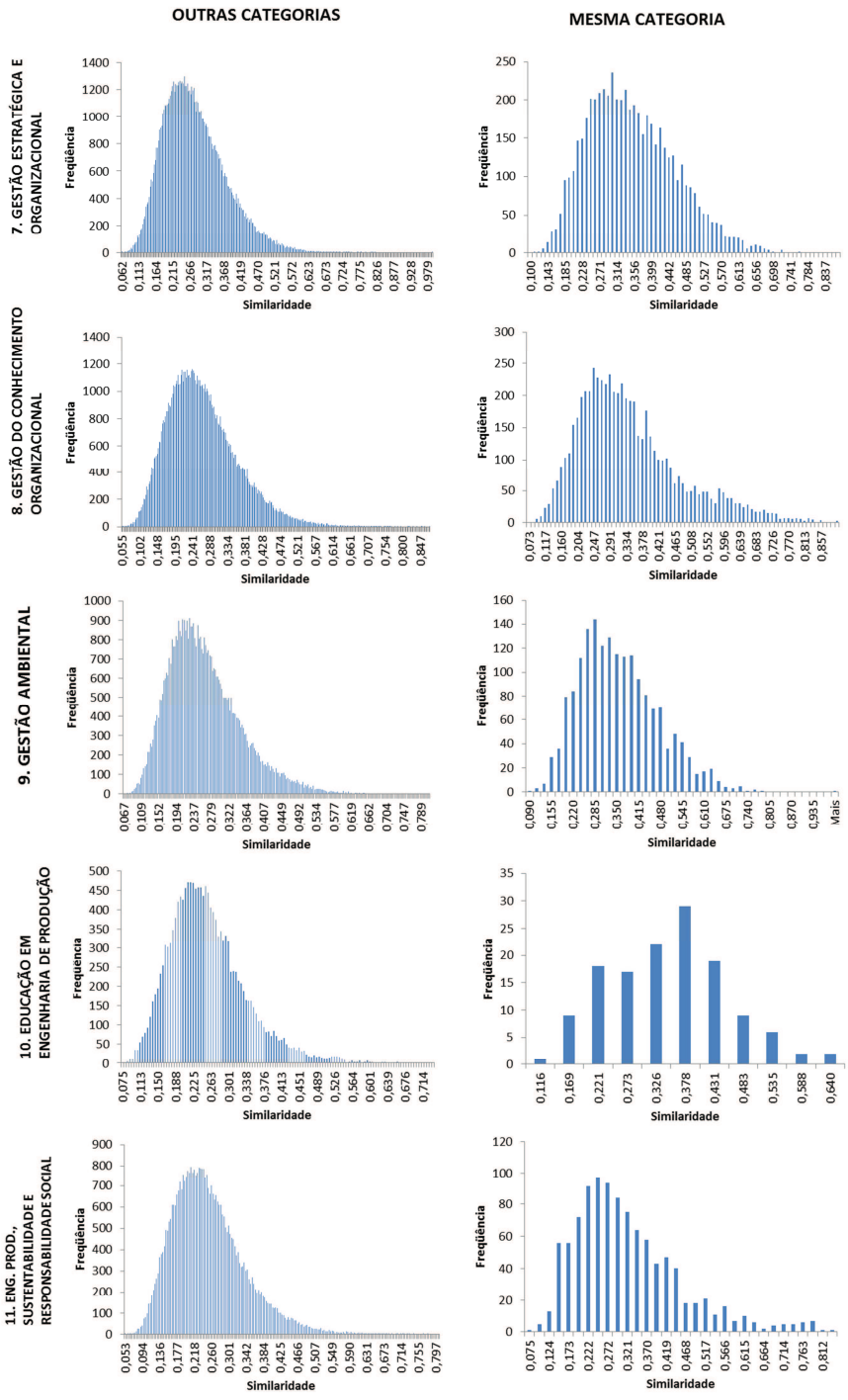

Figura 27 - Histograma de frequência da similaridade das categorias 7 a 11, considerando documentos de outras categorias e documentos da mesma categoria.

As Figuras 28 e 29 apresentam os resultados do Experimento 2. Para o categorizador k-NN utilizou-se k=1, optou-se por essa simplificação devido a pouca quantidade de documentos. Os categorizadores *Naive Bayes* e SVM não foram customizados. Os excelentes resultados observados, chegando a 100% na maioria das métricas com áreas completamente distintas do conhe-

cimento ressalta o grau de dificuldade em separar as categorias do domínio abordado neste trabalho (Engenharia de Produção), contribuindo uma ótima avaliação dos resultados obtidos.

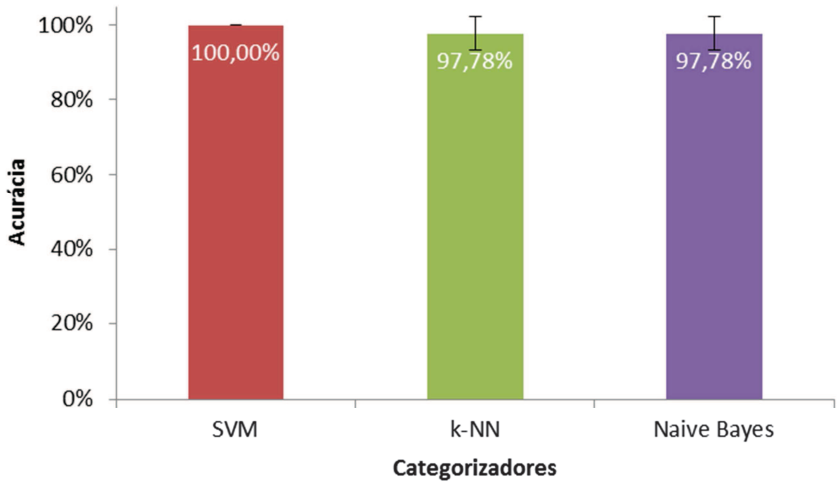

Figura 28 - Acurácia dos categorizadores SVM, k-NN e Naive Bayes no Experimento 2.

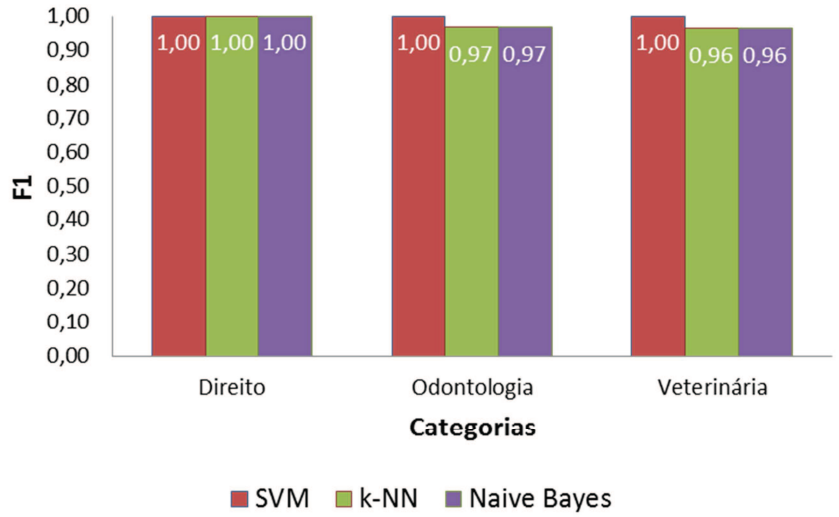

Figura 29 – Média da métrica F_1 dos categorizadores SVM, k-NN e Naive Bayes no Experimento 2.

De forma a apresentar um caso prático de utilização do método de grupo proposto neste trabalho, a escolha da área de submissão de um artigo extraído dos resultados desta dissertação e aceito para apresentação oral no ENEGEP 2012 foi realizada utilizando o classificador proposto, sendo a categoria 8, Gestão do Conhecimento Organizacional, a categoria de publicação

do trabalho. A Tabela 7 apresenta o resultado da votação, dentro do intervalo [0,1] que determinou essa escolha.

Tabela 7 – Resultado da votação pelo método proposto de artigo submetido ao ENEGEP 2012.

1	2	3	4	5	6	7	8	9	10	11
0,05	0,15	0,05	0,02	0,03	0,18	0,00	**0,29**	0,01	0,22	0,00

5. CONSIDERAÇÕES FINAIS

Considerando as características de similaridade entre os documentos das categorias envolvidas no trabalho, comprovado experimentalmente, e o fato de não existir na literatura um valor mínimo estipulado para determinar se os valores das métricas: acurácia, precisão, abrangência e F_1 são satisfatórios, trazendo essa subjetividade aos especialistas do domínio do conhecimento estudado, conclui-se que técnicas de aprendizagem de máquina aplicadas na categorização de textos, podem ser utilizadas como uma ferramenta de apoio a professores e alunos da área de Engenharia de Produção, de forma a auxiliá-los no processo de escolha da melhor área para publicação do seus artigos.

Para evidenciar o bom resultado atingido neste trabalho, com 71,1% de acurácia na categorização de documentos, tomam-se como parâmetro os resultados obtidos por Gomes e Moraes Filho (2011), que atingiram 84,6% de acurácia, utilizando um método baseado na engenharia do conhecimento, trabalhando com documentos de: Informática, Direito e Física e Galho (2003) que atingiu 91% de acurácia com um método baseado em aprendizagem de máquina utilizando documentos de Economia, Esportes, Policial, Saúde e Tecnologia. Observa-se que ambos os trabalhos utilizaram-se de categorias formadas por domínios do conhecimento completamente distintas, ao passo que no presente trabalho, apenas um domínio do conhecimento (Engenharia de Produção) foi envolvido.

De forma geral, a utilização de técnicas automatizadas de categorização de textos contribui com profissionais de diversas áreas na árdua tarefa de organização e recuperação de conteúdo em grandes volumes de documentos não estruturados, principalmente nos dias de hoje, onde inúmeras coleções de documentos científicos, como livros, teses e artigos ficaram ao alcance da comunidade acadêmica em formato digital.

5.1. CONTRIBUIÇÕES

Espera-se que a implementação da metodologia sugerida neste trabalho contribua para o crescimento, organização e qualidade da produção científica em Engenharia de Produção no Brasil e estimule outros estudos voltados para utilização da inteligência computacional na automatização de tarefas simples, porém bastante custosas de serem feitas manualmente, permitindo que os esforços sejam empenhados em tarefas mais nobres.

Com a realização deste trabalho foram publicados dois artigos:

1. Apresentação oral do artigo Categorização automática de artigos científicos da Engenharia de Produção utilizando métodos de aprendizagem de máquina no XXXII ENEGEP, realizado em 2012.

2. Publicação do artigo Categorização de documentos científicos de engenharia utilizando aprendizagem de máquina no XL Congresso Brasileiro de Educação em Engenharia (COBENGE), também realizado em 2012.

5.2. TRABALHOS FUTUROS

Com o desenvolvimento do presente trabalho, abrem-se oportunidades para continuidade da pesquisa voltada ao aprimoramento da categorização automática de textos, voltada ao auxílio e melhoria da qualidade da produção científica no Brasil. A seguir estão enumeradas algumas ideias de trabalhos futuros:

1. Implementar o modelo proposto dentro de uma aplicação de submissão de artigos de congresso, como ferramenta de auxílio ao pesquisador;

2. Realizar a categorização de 2º nível, sugerindo asub-área de submissão de um artigo de Engenharia de Produção;

3. Incluir outros métodos de aprendizagem de máquina no método de grupo proposto neste trabalho;

4. Adaptar o modelo para outra área do conhecimento;

5. Utilizar medidas de similaridade para selecionar os dados de treinamento de forma utilizar-se de documentos mais representativos para as categorias;

6. Utilizar o resultado da votação do método de grupo proposto, bem como os valores de confiança para tirar conclusões a respeito da relação do artigo com as áreas de publicação.

6. REFERÊNCIAS BIBLIOGRÁFICAS

ALMEIDA, Tiago A.; YAMAKAMI, Akebo; ALMEIDA, Jurandy. **Probabilistic anti-spam filtering with dimensionality reduction**. Proceedings Of The 2010 Acm Symposium On Applied Computing, Sierre, Switzerland, p.1802-1806, 2010. Disponível em: <http://dl.acm.org/citation.cfm?id=1609067.1609149>. Acesso em: 23 mar. 2012.

BARION, Eliana Cristina Nogueira; LAGO, Decio. **Mineração de Textos**. Revista de Ciências Exatas e Tecnologia, São Paulo, v. 3, n. 3, p.123-140, 2008.

BERRY, Michael W.; KOGAN, Jacob. **Text Mining Applications and Theory**. Wiley, 2010. 223 p.

BRASIL. ASSOCIAÇÃO BRASILEIRA DE ENGENHARIA DE PRODUÇÃO (ABEPRO). **ANAIS ENEGEP**. Disponível em: <http://www.abepro.org.br/publicacoes/>. Acesso em: 19 fev. 2011.

BRASIL. ASSOCIAÇÃO BRASILEIRA DE ENGENHARIA DE PRODUÇÃO (ABEPRO). **Áreas e Subáreas para envio de artigos**. Disponível em: <http://www.abepro.org.br/internasub.asp?m=1061&ss=42&c=1104 >. Acesso em: 08 abr. 2012.

BRASIL. COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR (CAPES). **Relação de Cursos Recomendados e Reconhecidos**. Disponível em: <http://conteudoweb.capes.gov.br/conteudoweb/ProjetoRelacaoCursosServlet?acao=pesquisarIes&codigoArea=30800005&descricaoArea=ENGENHARIAS+&descricaoAreaConhecimento=ENGENHARIA+DE+PRODU%C7%C3O&descricaoAreaAvaliacao=ENGENHARIAS+III>. Acesso em: 19 mar. 2012.

BRASIL. CONGRESSO ODONTOLÓGICO DE BAURU (COB). **Anais do 25◦ Congresso Odontológico de Bauru** Disponível em: <http://www.cobusp.com.br/>. Acesso em: 10 out. 2012.

BRASIL. CONGRESSO BRASILEIRO DE MEDICINA VETERINÁRIA (CONBRAVET). **Trabalhos do 35º Congresso Brasileiro de**

Medicina Veterinária. Disponível em: <http://www.sovergs.com.br/conbravet 2008/anais/cd/listaresumos.htm>. Acesso em: 10 out. 2012.

BRASIL. CONSELHO NACIONAL DE PESQUISA E PÓS-GRADUAÇÃO EM DIREITO (CONPEDI). **Anais do XXI Encontro Nacional do Conselho Nacional de Pesquisa e Pós Graduação em Direito**. Disponível em: <http://www.publicadireito.com.br/publicacao/?evento=37>. Acesso em: 10 out. 2012.

BRASIL. SIMPÓSIO BRASILEIRO DE ENGENHARIA DE PRODUÇÃO (SIMPEP). ANAIS SIMPEP. Disponível em: <http://www.simpep.feb.unesp.br/anais.php>. Acesso em: 19 mar. 2011.

CADWeb, Disponível em: <http://www.net.ucam-campos.br/>. Acesso em: 04 abr. 2012.

CHANG, Chih-chung; LIN, Chih-jen.**LIBSVM: A library for support vector machines**. Acm Trans. Intell. Syst. Technol., New York, p.1-27, 2011. Disponível em: <http://doi.acm.org/10.1145/1961189.1961199>. Acesso em: 20 maio 2011.

CORTES, Corinna; VAPNIK, Vladimir. **Support-Vector Networks**. Machine Learning, v. 20, p.273-297, 1995.

DOMINGOS, P.; PAZZANI, M. **On The Optimality of the Simple Bayesian Classifier Under Zero-one Loss**. Machine Learning, 29 (2/3), 103, 1997.

DORRE, J.; GERSTL, P.; SEIFFERT, R..**Text mining: finding nuggets in mountains of textual data**. Conf. On Knowledge Discovery And Data Mining (kdd-99), New York, USA, p.398-401, 1999.

FEINERER, Ingo; HORNIK, Kurt; MEYER, David. **Text Mining Infrastructure in R**. Journal Of Statistical Software, USA, v. 25, n. 5, p.1-54, 10 fev. 2008. Disponível em: <http://www.jstatsoft.org/v25/i05>. Acesso em: 02 maio 2011.

FELDMAN, Ronen; SANGER, James. **The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data**. New York: Cambridge University Press, 2007. 422 p.

FREE PDF to TXT Converter, Disponível em: <http://www.somepdf.com/some-pdf-to-txt-converter.html>. Acesso em: 20 mar. 2011.

GALHO, Thais Silva. **Categorização Automática de Documentos de Texto Utilizando Lógica Difusa.** 2003. 79 f. Monografia (Graduação) - Ulbra, Gravataí, 2003.

GOMES, Geórgia Regina Rodrigues. **Integração de Repositórios de Sistemas de Bibliotecas Digitais e de Sistemas de Aprendizagem**. Tese (Doutorado em Informática), Pontifícia Universidade Católica, Rio de Janeiro, 2005.

GOMES, Georgia Regina Rodrigues; MORAES FILHO, Rubens de Oliveira.

CADWeb – Categorização automática de documentos digitais. Ci. Inf., Brasília, v. 1, n. 40, p.68-76, jan. 2011.

HAYKIN, S.. **Neural Networks - A Compreensive Foundation**. 2. ed. New Jersey: Prentice-hall, 1999.

HSU, Chih-wei; CHANG, Chih-chung; LIN, Chih-jen. **A Practical Guide to Support Vector Classification**. Bioinformatics, v. 1, p.1-16, 2010. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.6.3096&rep=rep1&type=pdf>. Acesso em: 20 maio 2011.

LORENA, A. C.; CARVALHO, A. C. P. L. F. **Uma introdução às Support Vector Machines**. RITA, v.14, n. 2, 2007.

MARON, M. E.; KUHNS, J. L.. **On Relevance, Probabilistic Indexing and Information Retrieval**. Journal Of The Acm (jacm), New York, v. 8, n. 3, p.216-244, jul. 1961.

MIERSWA, Ingo et al. **YALE: Rapid Prototyping for Complex Data Mining Tasks**. Proceedings Of The 12th Acm Sig kdd International Conference On Knowledge Discovery And Data Mining: KDD , Philadelphia, p.935-940, 2006. Disponível em: <http://rapid-i.com/component/option,com_docman/task,doc_download/gid,25/Itemid,62/>. Acesso em: 02 maio 2011.

MONARD, M. C.; BARANAUSKAS, J. A.. **Conceitos de aprendizado de máquina**. In S. O. Rezende, editor, Sistemas Inteligentes - Fundamentos e Aplicações, p.89-114. Editora Manole, 2003.

BRASIL. NÚCLEO DE PESQUISA EM ENGENHARIA (NUPENGE). **CURSOS DE GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO**. Dados organizados pelo NUPENGE (Núcleo de Estudos e Pesquisas sobre Formação e Exercício Profissional em Engenharia da UFJF) com base nos dados coletados do site http://emec.mec.gov.br. Revisado em julho de 2011. Apoio: ABEPRO. Disponível em: <http://www.ufjf.br/proengprod/files/2010/05/cursosEP.xls>. Acesso em: 19 mar. 2012.

ORENGO, Viviane M; HUYCK, Christian. **A Stemming Algorithm for the Portuguese Language**, School of Computing Science, Middlesex University, London, England, 2001.

PLATT, J. C.. **Probabilistic outputs for support vector machines and comparison to regularized likelihood methods**, Cambridge, MA, MIT Press, 2000.

PORTER, Martin F.. **Snowball: A language for stemming algorithms**. Disponível em: <http://snowball.tartarus.org/texts/introduction.html>. Acesso em: 20 maio 2011.

RAPID-I (Alemanha) (Org.). **How does RapidMiner calculate Term Frequency (TF)?** Disponível em: <https://rapid-i.com/rapidforum/index.php?topic=3728.0>. Acesso em: 10 dez. 2012.

SEBASTIANI, Fabrizio. **Machine learning in automated text categorization**. Acm Computing Surveys, v. 34, n. 1, p.1-47, 2002.

SIMPSON, Matthew et al. **Using non-lexical features to identify effective indexing terms for biomedical illustrations**. Proceedings Of The 12th Conference Of The European Chapter Of The Association For Computational Linguistics: EACL '09, Stroudsburg, Pa, USA, p.737-744, 2009. Disponível em: <http://dl.acm.org/citation.cfm?id=1609067.1609149>. Acesso em: 23 mar. 2012.

SOUTO, M. C. P.; LORENA, A. C.; DELBEM A. C. B.; CARVALHO, A. C. P. L. F.. **Técnicas de Aprendizado de Máquina para problemas de Biologia Molecular**, p.103–152. Minicursos de Inteligência Artificial, Jornada de Atualização Científica em Inteligência Artificial, XXIII Congresso da Sociedade Brasileira de Computação, 2003.

TAN, Pang-ning; STEINBACH, Michael; KUMAR, Vipin. **Introdução ao Data Mining Mineração de Dados**. Rio de Janeiro: Ciência Moderna Ltda, 2009. 900 p.

VAPNIK, Vladmir. **The Nature of Statistical Learning Theory**. 2. ed. New York: Springer, 2000. 314 p.

VIERA, Angel F. G.; VIRGIL, Johnny. **Uma revisão dos algoritmos de radicalização em língua portuguesa.** Information Research, vol.12, n. 3, 2007. Disponível em http://informationr.net/ir/12-3/paper315.html. Acesso em 04 abr. 2012.

WETTSCHERECK, Dietrich; AHA, David W.; MOHRI, Takao. **A Review and Empirical Evaluation of Feature Weighting Methods for a Class of Lazy Learning Algorithms**. Artificial Intelligence Review, Springer Netherlands, v. 11, n. 1, p.273-314, 01 fev. 1997. Disponível em: <http://dx.doi.org/10.1023/A:1006593614256>. Acesso em: 04 abr. 2012.

WILLETT, Peter. **The Porter stemming algorithm: then and now**. Program: Electronic Library And Information Systems, v. 40, n. 3, p.219-223, 2006.

YANG, Yming; LIU, Xin. **A re-examination of text categorization methods.** Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, New York, p.42-49, 1999.

ZHANG, H..**The optimality of naive bayes**. Proceedings Of The Seventeenth International Florida Artificial Intelligence Research Society Conference, Miami Beach, p.562-567, 2004. Disponível em: <http://www.springerlink.com/content/51t4233286xn76rr/fulltext.pdf>. Acesso em: 23 mar. 2012.

ZELAIA, A.; ALEGRIA, I.. **A Multiclass/Multilabel Document Categorization System: Combining Multiple Classifiers in a Reduced Dimension**. Applied Soft Computing Journal, 2011. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1568494611002201>. Acesso em: 04 abr. 2012.

ZHANG, Jianping; QIN, Jason; YAN, Qiuming. **The Role of URLs in Objectionable Web Content Categorization**. Proceedings Of The 2006 Ieee/wic/acm International Conference On Web Intelligence, Washington, DC, USA, p.277-283, 2006. Disponível em: <http://dx.doi.org/10.1109/WI.2006.170>. Acesso em: 23 mar. 2012.

APÊNDICE A – Resultado do processo de busca pelo melhor valor de *k* do algoritmo k-NN.

k	acurácia (%)
1	61,09
2	60,51
3	63,91
4	65,55
5	65,81
6	65,84
7	66,30
8	66,47
9	67,19
10	67,16
11	66,96
12	67,25
13	67,34
14	67,42
16	67,77
17	67,60
19	67,45
21	67,82
23	68,23
25	68,20
28	67,85
30	67,65
33	67,68
36	67,82
40	67,31
44	67,22
48	66,87
52	66,56
58	66,44
63	66,35
69	66,50
76	65,96
83	66,10
91	65,89
100	65,67

APÊNDICE B – Resultado do processo de busca dos parâmetros *C* e *ε* do classificador SVM.

C	*ε*	acurácia (%)
2,00	0,00003	70,94
0,50	0,00700	70,14
0,50	0,00012	69,96
2,00	0,00200	69,79
0,50	0,00003	69,63
0,50	0,03125	69,63
32,00	0,00012	69,62
2,00	0,50000	69,36
8,00	0,12500	69,32
512,00	0,12500	69,31
128,00	0,00700	69,30
512,00	0,00003	69,17
2,00	0,00012	69,17
8,00	0,03125	69,15
128,00	0,00200	69,14
2,00	0,12500	69,13
32,00	0,00003	69,12
0,50	0,00200	68,98
2,00	0,00700	68,98
0,50	0,12500	68,80
128,00	0,00012	68,79
512,00	0,00700	68,79
2,00	0,03125	68,79
32,00	0,00200	68,66
8,00	0,50000	68,65
8,00	0,00003	68,64
0,13	0,00200	68,50
8,00	0,00012	68,49
8,00	0,00200	68,48
0,50	0,50000	68,48
128,00	0,00003	68,35
8,00	0,00700	68,32
32,00	0,03125	68,31
32,00	0,00700	68,30
512,00	0,00012	68,29
0,13	0,00003	68,14
512,00	0,00200	68,13
32,00	0,12500	68,00
32,00	0,50000	67,83
512,00	0,50000	67,65
128,00	0,12500	67,49
0,13	0,12500	67,31
128,00	0,50000	67,30
0,13	0,03125	67,16
128,00	0,03125	67,16
0,13	0,00700	67,16
0,13	0,50000	67,15
0,13	0,00012	66,68
512,00	0,03125	66,36
512,00	2,00000	65,71
2,00	2,00000	65,54
32,00	2,00000	65,53
8,00	2,00000	65,20
0,50	2,00000	65,18
128,00	2,00000	65,02
0,03	0,03125	64,54
0,03	0,00700	63,85
0,03	0,00003	63,19
0,03	0,12500	63,19
0,03	0,00012	63,04
0,03	0,00200	62,72
0,03	0,50000	62,54
0,13	2,00000	60,22
0,03	2,00000	56,06

APÊNDICE C – Resultado consolidado do processo de otimização e avaliação preliminar dos categorizadores.

Categorias	SVM			k-NN			*Naive Bayes*		
	p	***r***	***F*** $_1$	***p***	***r***	***F*** $_1$	***p***	***r***	***F*** $_1$
1 GESTÃO DA PRODUÇÃO	0,63	0,60	**0,62**	0,59	0,57	**0,58**	0,58	0,54	**0,56**
2 GESTÃO DA QUALIDADE	0,75	0,75	**0,75**	0,69	0,79	**0,74**	0,68	0,75	**0,72**
3 GESTÃO ECONÔMICA	0,68	0,71	**0,70**	0,69	0,66	**0,68**	0,68	0,66	**0,67**
4 ERGONOMIA E SEGURANÇA DO TRABALHO	0,88	0,90	**0,89**	0,86	0,88	**0,87**	0,89	0,84	**0,87**
5 GESTÃO DO PRODUTO	0,77	0,74	**0,75**	0,66	0,69	**0,68**	0,65	0,72	**0,68**
6 PESQUISA OPERACIONAL	0,72	0,78	**0,75**	0,77	0,68	**0,72**	0,75	0,64	**0,69**
7 GESTÃO ESTRATÉGICA E ORGANIZACIONAL	0,65	0,63	**0,64**	0,63	0,62	**0,62**	0,66	0,51	**0,58**
8 GESTÃO DO CONHECIMENTO ORGANIZACIONAL	0,68	0,73	**0,70**	0,61	0,75	**0,67**	0,51	0,80	**0,62**
9 GESTÃO AMBIENTAL	0,71	0,73	**0,72**	0,72	0,68	**0,70**	0,76	0,54	**0,64**
10 EDUCAÇÃO EM ENGENHARIA DE PRODUÇÃO	0,72	0,68	**0,70**	0,72	0,53	**0,61**	0,75	0,54	**0,63**
11 ENG. PROD., SUSTENTABILIDADE E RESPONSABILIDADE SOCIAL	0,58	0,38	**0,46**	0,70	0,27	**0,39**	0,48	0,43	**0,45**
Acurácia	**71,10% (±3,06%)**			**68,23%(±2,35%)**			**65,61%(±3,58%)**		

APÊNDICE D – Lista de *stopwords* utilizadas no trabalho (*Stop-list*).

a	da	diz	feita	muito	pelos	quanto	tido
à	daquele	dizem	feitas	muitos	pequena	quantos	tinha
agora	daqueles	do	feito	na	pequenas	que	tinham
ainda	das	dos	feitos	não	pequeno	quem	toda
alguém	de	e	foi	nas	pequenos	são	todas
algum	dela	é	for	nem	per	se	todavia
alguma	delas	e'	foram	nenhum	perante	seja	todo
algumas	dele	ela	fosse	nessa	pode	sejam	todos
alguns	deles	elas	fossem	nessas	pôde	sem	tu
ampla	depois	ele	grande	nesta	podendo	sempre	tua
amplas	dessa	eles	grandes	nestas	poder	sendo	tuas
amplo	dessas	em	há	ninguém	poderia	será	tudo
amplos	desse	enquanto	isso	no	poderiam	serão	última
ante	desses	entre	isto	nos	podia	seu	últimas
antes	desta	era	já	nós	podiam	seus	último
ao	destas	essa	la	nossa	pois	si	últimos
aos	deste	essas	la	nossas	por	sido	um
após	deste	esse	lá	nosso	porém	só	uma
aquela	destes	esses	lhe	nossos	porque	sob	umas
aquelas	deve	esta	lhes	num	posso	sobre	uns
aquele	devem	está	lo	numa	pouca	sua	vendo
aqueles	devendo	estamos	mas	nunca	poucas	suas	ver
aquilo	dever	estão	me	o	pouco	talvez	vez
as	deverá	estas	mesma	os	poucos	também	vindo
até	deverão	estava	mesmas	ou	primeiro	tampouco	vir
através	deveria	estavam	mesmo	outra	primeiros	te	vos
cada	deveriam	estávamos	mesmos	outras	própria	tem	vós
coisa	devia	este	meu	outro	próprias	tendo	
coisas	deviam	estes	meus	outros	próprio	tenha	
com	disse	estou	minha	para	próprios	ter	
como	disso	eu	minhas	pela	quais	teu	
contra	disto	fazendo	muita	pelas	qual	teus	
contudo	dito	fazer	muitas	pelo	quando	ti	