

Evandro Dalbem Lopes

Utilização do modelo *skip-gram* para
representação distribuída de palavras no
projeto Media Cloud Brasil

Rio de Janeiro
2015

Evandro Dalbem Lopes

Utilização do modelo *skip-gram* para
representação distribuída de palavras no
projeto Media Cloud Brasil

Dissertação apresentada a Escola de Matemática Aplicada da Fundação Getulio Vargas, para a obtenção de Título de Mestre em Ciências, na Área de Modelagem Matemática da Informação.

Orientador: Flávio Codeço Coelho

**Rio de Janeiro
2015**

Lopes, Evandro Dalbem

Utilização do modelo skip-gram para representação distribuída de palavras no projeto Media Cloud Brasil / Evandro Dalbem Lopes. - 2015.
62 f.

Dissertação (mestrado) – Fundação Getulio Vargas, Escola de Matemática Aplicada.

Orientador: Flávio Codeço Coelho.

Inclui bibliografia.

1. Processamento da linguagem natural (Computação). 2. Media Cloud Brasil. 3. Redes neurais (Computação). I. Coelho, Flávio Codeço. II. Fundação Getulio Vargas. Escola de Matemática Aplicada. III. Título.

CDD – 006.35

EVANDRO DALBEM LOPES

**UTILIZAÇÃO DO MODELO *SKIP-GRAM* PARA REPRESENTAÇÃO DISTRIBUÍDA
DE PALAVRAS NO PROJETO MEDIA CLOUD BRASIL.**


Dissertação apresentada ao Curso de Mestrado em Modelagem Matemática da Informação da Escola de Matemática Aplicada da Fundação Getúlio Vargas para obtenção do grau de Mestre em Modelagem Matemática da Informação.

Data da defesa: 30/06/2015.

ASSINATURA DOS MEMBROS DA BANCA EXAMINADORA

A handwritten signature in blue ink, appearing to read 'Flávio Codeço Coelho', is written over a horizontal line.

Flávio Codeço Coelho
Orientador (a)

A handwritten signature in blue ink, appearing to read 'Renato Rocha Souza', is written over a horizontal line.

Renato Rocha Souza

A handwritten signature in blue ink, appearing to read 'Sabrina Camargo', is written over a horizontal line.

Sabrina Camargo

A handwritten signature in blue ink, appearing to read 'Heliana Ribeiro de Mello', is written over a horizontal line.

Heliana Ribeiro de Mello

Aos meus pais, Manoel e Marcia

Agradecimentos

À CAPES e a Escola de Matemática Aplicada da FGV pelo suporte ao longo desses 2 anos.

Ao meu orientador Flávio Codeço pela confiança, paciência e aprendizado passado ao longo deste tempo. Ao professor e coordenador Renato Rocha por ter acreditado no meu potencial e me aceitado como aluno. A todos os professores e funcionários da EMAp que contribuíram para minha formação.

Aos meus pais, Manoel e Marcia, ao meu irmão Manoel, aos meus tios, tias e toda família por ter acreditado e apoiado toda esta loucura que é fazer um mestrado. A Carolina pela companhia, atenção e carinho que certamente contribuíram para este trabalho.

Gostaria de agradecer também a todos os meus amigos que estiveram presentes ao longo destes anos.

Resumo

Existe um problema de representação em processamento de linguagem natural, pois uma vez que o modelo tradicional de *bag-of-words* representa os documentos e as palavras em uma única matriz, esta tende a ser completamente esparsa. Para lidar com este problema, surgiram alguns métodos que são capazes de representar as palavras utilizando uma representação distribuída, em um espaço de dimensão menor e mais compacto, inclusive tendo a propriedade de relacionar palavras de forma semântica. Este trabalho tem como objetivo utilizar um conjunto de documentos obtido através do projeto *Media Cloud Brasil* para aplicar o modelo *skip-gram* em busca de explorar relações e encontrar padrões que facilitem na compreensão do conteúdo.

Palavras-chave: *Skip-gram*, Processamento de Linguagem Natural, *Media Cloud Brasil*, Redes Neurais

Abstract

There is a representation problem when working with natural language processing because once the traditional model of bag-of-words represents the documents and words as single matrix, this one tends to be completely sparse. In order to deal with this problem, there are some methods capable of represent the words using a distributed representation, with a smaller dimension and more compact, including some properties that allow to relate words on the semantic form. The aim of this work is to use a dataset obtained by the Media Cloud Brasil project and apply the skip-gram model to explore relations and search for pattern that helps to understand the content.

Keywords: *Skip-gram*, Natural Language Processing, *Media Cloud Brasil*, Neural Networks

Lista de Figuras

2.1	Descrição conceitual da Decomposição em Valores Singulares - SVD . . .	8
2.2	Descrição conceitual da Alocação Latente de Dirichlet mostrando como um documento é composto por uma mistura de tópicos (Blei et al., 2003)	9
2.3	Posição das palavras utilizando a representação distribuída em um espaço bi-dimensional criado pelo PCA. (Mikolov et al., 2013c)	13
3.1	O modelo <i>skip-gram</i> . (Rong, 2014)	22
3.2	Exemplo de árvore binária utilizando <i>softmax</i> hierárquico. Os nós brancos são palavras no vocabulário e os escuros não os nós internos. Um exemplo de caminho até a palavra w_2 é destacado e o tamanho do caminho $L(w_2) = 4$. $n(w, j)$ indica o j -ésimo nó no caminho da raiz até a palavra w . (Rong, 2014)	24
3.3	Ilustração gráfica do algoritmo <i>k-means</i> . (Wikipedia, 2014)	27
4.1	Palavras associadas a palavra central “violência”, com profundidade máxima de 2 palavras.	32

Lista de Tabelas

3.1	Lista com alguns exemplos de <i>stop words</i>	19
4.1	Exemplo de frases criadas com a utilização de até trigramas	29

Sumário

1	Introdução	1
1.1	A mídia brasileira	1
1.2	O Projeto Media Cloud	2
1.3	O projeto Media Cloud Brasil	3
1.4	Objetivos deste trabalho	4
2	Referencial Teórico	5
2.1	Processamento de Linguagem Natural	5
2.1.1	Um pouco de história	5
2.2	Análise Semântica Latente - LSA	7
2.3	Alocação Latente de Dirichlet	9
2.4	Modelos de Linguagem de Redes Neurais	10
2.4.1	Exploração de relações semânticas entre as palavras	11
3	Metodologia	15
3.1	O Processo de captura do Media Cloud	15
3.2	Processamento de Linguagem Natural	17
3.2.1	O modelo “ <i>Bag-of-words</i> ”	17
3.2.2	Pré-processamento do texto	18
3.3	O Modelo N-Grama	18

3.3.1	Criação de frases com o modelo N-Gram	19
3.4	O Modelo <i>Skip-Gram</i>	21
3.4.1	Representação distribuída de palavras	21
3.4.2	<i>Skip-Gram</i>	21
3.4.3	<i>Softmax</i> Hierárquico	23
3.5	Clusterização - <i>K-means</i>	26
3.6	Estratégia para a identificação não-supervisionada de grupos	28
3.7	Softwares Utilizados	28
4	Resultados	29
4.1	Descrição dos dados	29
4.2	Aplicando o modelo <i>skip-gram</i>	30
4.2.1	Explorando relações entre as palavras	30
4.2.2	Limpeza e classificação do conteúdo através do modelo <i>skip-gram</i>	31
5	Conclusão e considerações finais	43
5.1	Trabalhos Futuros	44
	Referências Bibliográficas	46

Capítulo 1

Introdução

1.1 A mídia brasileira

Vivemos na era da informação e selecionando-se uma quantidade de tempo qualquer, podemos ver que jamais tanta informação foi produzida em dois intervalos distintos de igual tamanho. Se pararmos para refletir sobre o assunto, perceberemos que uma única pessoa produz uma quantidade extrema de dados diários. Sua movimentação bancária, seus acessos a e-mails, qualquer coisa feita pelo ser humano no mundo moderno é insumo para a geração de dados. Se para uma única pessoa esta quantidade é suficientemente grande, o que pode ser dito sobre a produção de dados relacionada aos grandes veículos de publicação? Uma coisa é certa: Todos os dias são geradas milhares de notícias com conteúdo completamente diferente ao redor do globo.

Observou-se nos últimos anos um grande avanço quanto a acessibilidade digital, seja por meio de computadores pessoais ou através da utilização de outros dispositivos, como por exemplo smartphones ou tablets. Isso fez com que fosse criada uma enorme demanda de conteúdo online que até então era inexistente - já que não havia a necessidade de publicações digitais - que é a de publicação editorial no formato digital.

Em aproximadamente uma década, pode-se observar uma migração quase que com-

pleta para esta nova forma de publicação. O que vemos hoje é um reflexo deste movimento e praticamente todos os grandes veículos editoriais migraram também para suportar esta nova plataforma de geração de conteúdo, que em algum sentido é mais interessante e eficiente que a publicação tradicional, já que até a década de 90 os jornais eram impressos podendo atingir as pessoas com uma baixa granularidade, a entrega de conteúdo jornalístico impresso era de no máximo uma vez ao dia, diferente do conteúdo digital, que pode ser entregue a qualquer momento.

Assim como as notícias televisivas, o conteúdo digital também tem um sério problema para a ciência, que é a ausência de dados estruturados. Isto é uma implicação forte que durante muito tempo inviabilizou a aplicação de modelos matemáticos e estatísticos e o desenvolvimento de novos métodos.

Um fato que não pode ser ignorado é que nos últimos anos, com o avanço e o fácil acesso a computadores pessoais e a smartphones fez com que fosse criado um novo segmento de publicação editorial, que são as mídias digitais. Todos os dias inúmeros eventos acontecem ao redor do globo: crise hídrica, análise esportiva, comentários políticos. Tudo isso é coberto pela mídia tradicional e esta quantidade de informação gerada é maior do que jamais foi e se considerarmos que existe também a mídia alternativa este número pode ser ainda maior.

1.2 O Projeto Media Cloud

O Media Cloud¹ é um projeto *open source* - feito pelo *Berkman Center for Internet & Society* da Universidade de Harvard em conjunto com o *Center for Civic Media* do Instituto de Tecnologia de Massachussets (MIT) - que permite localizar e seguir centenas de milhares de jornais, revistas e blogs, armazenando as informações coletadas de forma estruturada, possibilitando assim a recuperação do conteúdo publicado. O banco de

¹<http://mediacloud.org/>

dados coletado e indexado pelo *Media Cloud* permite a pesquisadores buscarem por pessoas, lugares e eventos (desde Michal Jackson até as eleições Iranianas). E não apenas a busca é possível, como também dizer quando, onde e quão frequente foi a cobertura de cada um destes tópicos. Portanto, o *Media Cloud* permite responder perguntas como: Dado uma notícia, onde foi sua origem? Existe alguma diferença de opinião entre a mídia popular e a mídia alternativa? É possível caracterizar o conteúdo de um website? Existem ciclos de notícias?

Em 2013 um artigo publicado por Benkler et al. (2015) ganhou notoriedade após mapear e analisar um conjunto de 9.757 histórias relacionadas ao COICA (*Combating Online Infringement and Counterfeits Act*)/SOPA (*Stop Online Piracy Act*)/PIPA (*Protect IP Act*) debate, desde Setembro de 2010 até o fim de janeiro de 2012 com dados provenientes do *Media Cloud*.

1.3 O projeto Media Cloud Brasil

O projeto *Media Cloud Brasil*² é uma replicação do projeto *Media Cloud* especializado na coleta de informações veiculadas pela mídia brasileira. É um projeto recente, tendo sido iniciado no começo de 2013, com uma parceria entre a Escola de Matemática Aplicada da Fundação Getulio Vargas - Rio de Janeiro e o MIT. Apesar de ter uma proposta inicial idêntica à versão original, o projeto acabou tomando um rumo bem distinto, sendo esta uma implementação totalmente independente e alternativa ao projeto original. Pode-se descrever o processo de captura em 3 etapas principais: Definição da mídia, *crawling* e extração de texto.

Devido às fontes online mostrarem-se heterogêneas, do ponto de vista de qualidade do conteúdo, percebeu-se uma necessidade que antes de olhar para estes dados com fins analíticos, deve-se fazer uma curadoria das fontes e conteúdo coletado. Problemas

²https://github.com/NAMD/mediacloud_backend

de codificação, conteúdo capturado de várias línguas e coleta de *spam* são exemplos de conteúdos desqualificados que já foram identificados.

1.4 Objetivos deste trabalho

O objetivo deste trabalho é utilizar metodologia de aprendizado de máquina e processamento de linguagem natural para identificar possíveis problemas de conteúdo capturado pelo *Media Cloud Brasil* e fazer inferência sobre possíveis assuntos contidos no conjunto de dados disponíveis.

Este trabalho está organizado da seguinte forma: no capítulo 2 apresentamos o referencial teórico em que se baseia este trabalho, no capítulo 3 abordamos a metodologia utilizada para a execução deste trabalho e no capítulo 4 apresentamos os resultados obtidos com o modelo *skip-gram* nos dados do *Media Cloud Brasil*.

Capítulo 2

Referencial Teórico

Neste capítulo apresentaremos o referencial teórico matemático em que se baseia este trabalho.

2.1 Processamento de Linguagem Natural

2.1.1 Um pouco de história

Processamento de Linguagem Natural (*Natural Language Processing* - *NLP*) é o desenvolvimento de modelos computacionais para a realização de tarefas que dependem de informações expressas em uma língua natural. A história do processamento de linguagem natural começou na década de 1950, embora alguns trabalhos possam ser encontrados em períodos anteriores. Neste ano, Turing ([Turing, 1950](#)) publicou seu mais famoso artigo “*Computing Machinery and Intelligence*”, no qual propôs o teste de Turing, onde o objetivo era determinar se máquinas poderiam pensar como seres humanos.

No exemplo original um juiz humano conversa em uma língua natural com um humano e uma máquina, sem saber qual dos dois é a máquina e o ser humano. Se o juiz não puder diferenciar com segurança a máquina do humano, então é dito que a

máquina passou no teste. A conversa está limitada a um canal contendo apenas texto (por exemplo, um teclado e um monitor de vídeo). Se no final do teste o interpretador não conseguir distinguir quem é o humano, então pode-se concluir que o computador pode pensar, segundo o teste de Turing.

Em 1954 o experimento de Georgetown envolveu a tradução automática de mais de sessenta sentenças do russo para o inglês. O experimento foi um sucesso e os autores afirmaram que dentro de três ou cinco anos a tradução automática seria um problema resolvido. No entanto o progresso real era muito mais lento do que imaginavam e o financiamento para a tradução automática foi reduzido drasticamente. Desde finais da década de 1980, à medida que o poder computacional foi aumentando e tornando-se menos custoso o interesse nos modelos estatísticos para a tradução automática foi também aumentando.

Em 1960 surgiram alguns sistemas de processamento de linguagem natural notavelmente bem sucedidos. Foram eles:

- SHRDLU: foi um programa de computador desenvolvido pelo norte-americano Terry Winograd no Instituto Tecnológico de Massachusetts (MIT) entre 1968 e 1970 para contextualizar partes de uma língua natural. O SHRDLU foi primariamente um analisador de linguagem que permitia a integração com seu usuário utilizando termos da língua inglesa. O usuário podia instruir o SHRDLU a mover vários objetos, como por exemplo blocos, cones e esferas apenas dando ordens, como por exemplo “mova o bloco verde para dentro da caixa”. O que o SHRDLU fez foi unicamente uma combinação de ideias simples que se somaram e fizeram a simulação de entendimento algo bem convincente.
- ELIZA: foi um programa de computador e exemplo de uma primeira aplicação de processamento de linguagem natural. ELIZA era operado por resposta à scripts e o mais famoso deles era um médico, baseado em uma simulação de uma psicone-

rapeuta. Utilizando quase nenhuma informação sobre o pensamento humano ou emoções este software as vezes proporcionava uma surpreendente interação que se assemelhava a um ser humano. ELIZA foi desenvolvido por Joseph Weizenbaum no MIT entre 1964 e 1966. Quando o “paciente” ultrapassa a minúscula base de dados, a resposta dada tornava-se genérica. Por exemplo, reponder a afirmação “Minha cabeça dói” com “Porquê você diz isso?”.

Durante a década de 1980 a maioria dos sistemas de processamento de linguagem natural eram baseados em um conjunto de regras escritas a mão. Porém no final da década de 1980 houve uma revolução no sentido em que foram introduzidas técnicas de aprendizado de máquina para o processamento de linguagem. Esta revolução deu-se em parte pelo gradual desenvolvimento das teorias linguísticas de Chomsky. Com isso, os primeiros algoritmos de aprendizado de máquina utilizados, como as árvores de decisão, eram capazes de produzir um sistema com regras “se-então” que eram similares as regras que inicialmente eram escritas a mão.

2.2 Análise Semântica Latente - LSA

A Análise Semântica Latente (*Latent Semantic Analysis - LSA*), proposta por [Deerwester et al. \(1990\)](#) é uma técnica matemática que serve para extrair e inferir relações de uso contextual diferente dos tradicionais métodos de processamento de linguagem natural. Ao contrário destes, não utiliza nenhuma base de conhecimento, redes semânticas, analisadores sintáticos ou morfológicos de nenhuma natureza, utilizando como entrada apenas texto formado por cadeias de caracteres. O que a Análise Semântica Latente faz é uma decomposição de matrizes utilizando Decomposição em Valores Singulares (*Singular Values Decomposition - SVD*), de forma a fazer uma aproximação de ordem k da matriz original, onde $A_k = U_k \cdot \Sigma_k \cdot V_k^T$. A imagem [2.1](#), mostra uma descrição conceitual de como é este processo.

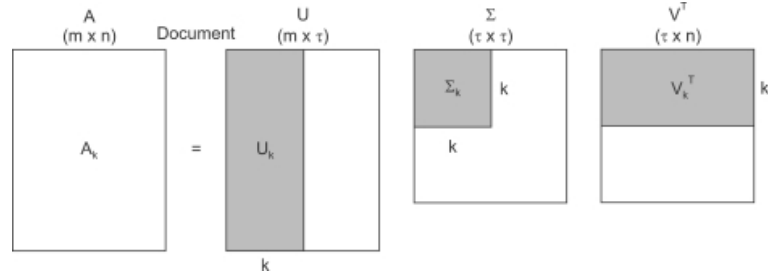


Figura 2.1: Descrição conceitual da Decomposição em Valores Singulares - SVD

É importante notar que a similaridade estimada pela Análise Semântica Latente não é apenas contagem de frequências, co-ocorrências ou correlações, e sim uma poderosa fatorização que torna capaz inferir corretamente sobre relações mais profundas - daí o termo “semântica latente”. Como consequência os resultados obtidos pela Análise Semântica Latente podem ser melhores que preditores baseados em julgamentos humanos.

Um artigo publicado por [Osiński et al. \(2004\)](#) propôs uma forma inovadora de indução de temas em uma coleção de documentos utilizando a Análise Semântica Latente. No trabalho proposto, identificam-se frases (que serão posteriormente candidatas a nomes de grupos) caracterizando-as como *pseudo* documentos. Em seguida aplica-se a decomposição dos valores singulares e aplica-se uma projeção da matriz aproximada para o espaço dos novos documentos. Desta forma torna-se possível associar um documento a cada um dos candidatos a nomes de grupos (que é o conjunto formado pela união das frases identificadas com as palavras).

Embora o trabalho inicial proposto por [Deerwester et al. \(1990\)](#) tenha sido uma forma de realizar consultas e a recuperação de informações de forma que não fosse binária e sim probabilística, logo esta técnica tornou-se muito popular em análises de processamento de linguagem natural.

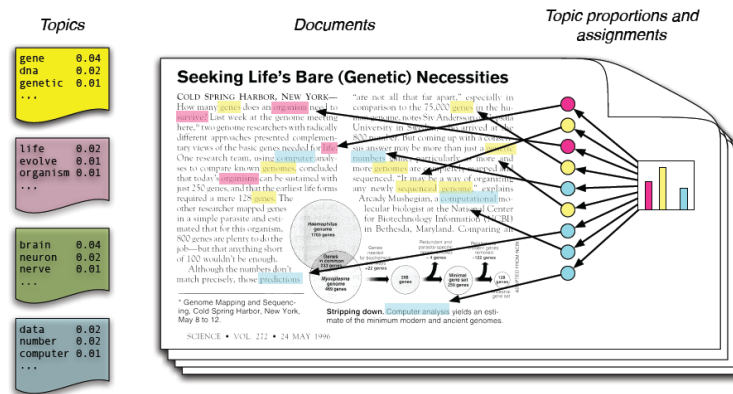


Figura 2.2: Descrição conceitual da Alocação Latente de Dirichlet mostrando como um documento é composto por uma mistura de tópicos (Blei et al., 2003)

2.3 Alocação Latente de Dirichlet

A Alocação Latente de Dirichlet (*Latent Dirichlet Allocation - LDA*) pertence a uma classe de modelos chamados modelos de tópicos (*topic models*). Seu desenvolvimento teve início com Blei et al. (2003), e tem como objetivo oferecer uma metodologia de aprendizado não-supervisionado que permita resumir grandes quantidades de informações textuais. A Alocação Latente de Dirichlet é um modelo Bayesiano composto por três níveis, no qual cada item de uma coleção de textos é modelado como uma mistura finita sobre um conjunto de tópicos. No contexto de modelagem de textos, a probabilidade dos tópicos oferece uma representação para os documentos. A figura 2.2 ilustra este conceito.

Este modelo surgiu do pensamento em que ao escrever um texto o redator possui vários tópicos em mente, e o ato de escrever significa permutar as palavras dos tópicos continuamente. Imagine a situação em que é publicado um artigo de jornal sobre um determinado tema. Este artigo pode fazer referência a vários tópicos, cada um destes associados com palavras específicas. Por exemplo, se estivermos falando de um conjunto de documentos sobre bioestatística, poderão existir dois tópicos distintos, um sobre

estatística - em que palavras como “probabilidade” e “distribuição” possuem altas probabilidades - e outro sobre biologia - onde palavras como “cérebro” e “plantas” possuem alta probabilidade.

Portanto o objetivo da Alocação Latente de Dirichlet é tratar cada um destes tópicos (que são desconhecidos) como uma mistura de distribuições de probabilidade sobre as palavras, tornando assim um documento uma mistura de tópicos e esta inferência se dá utilizando distribuições de *Dirichlet*. O grande problema deste método é o alto custo computacional, embora existam diversas implementações eficientes para melhorar a performance, e a não garantia de convergência pela aproximação que é utilizada.

2.4 Modelos de Linguagem de Redes Neurais

Utilizando o modelo tradicional de *bag-of-words* existe o problema de representação esparsa, sendo cada documento representado por um vetor com todas as palavras do vocabulário (sendo também referenciado na literatura como a maldição da dimensionalidade - *the curse of dimensionality* - que afirma que o número de dados de entrada deve crescer exponencialmente de acordo com o número de atributos que estes possuem [Härdle et al. \(2012\)](#)). E em processamento de linguagem natural este problema é bem evidente pois o tamanho do vocabulário pode facilmente chegar a algumas dezenas de milhões de palavras.

Alguns modelos já foram propostos para estimação da representação distribuída de palavras, dentre eles os citados anteriormente Análise Semântica Latente e a Alocação Latente de Dirichlet.

Representação de palavras como vetores com valores contínuos é uma metodologia recorrente. Uma arquitetura de modelo popular para estimar um modelo de linguagem de rede neural (*Neural Network Language Model* - NNLM) foi proposto por [Bengio et al. \(2003\)](#), onde uma rede neural com uma camada de projeção linear e uma camada oculta

não-linear foi utilizada para aprender conjuntamente a representação das palavras e criar uma representação distribuída.

Uma outra arquitetura importante de NNLM foi apresentada por Mikolov (2007), onde os vetores representando palavras foram primeiro aprendidos utilizando uma rede neural com uma única camada oculta. Os vetores de palavras foram então utilizados para treinar a NNLM. Entretanto, os vetores de palavras são aprendidos mesmo sem construir a NNLM completa. Neste trabalho, utilizamos uma extensão desta arquitetura, e focamos apenas no primeiro passo, onde os vetores de palavras são aprendidos utilizando um modelo simples.

O modelo escolhido para representar a arquitetura aqui proposta foi o modelo *skip-gram*. O *skip-gram* é um modelo que, dada uma palavra tenta prever quais são as palavras que aparecem no mesmo contexto que ela. Como em um texto evita-se a repetição massiva das mesmas palavras este método é capaz de captar relações profundas através da inferência sobre o contexto.

Alguns trabalhos anteriores já mostraram que a representação distribuída pode melhorar a qualidade do modelo significativamente em muitas tarefas de processamento de linguagem natural (Turian et al., 2010). Porém muitas das arquiteturas de redes neurais implementadas possuem problemas relacionados a custo computacional e uma solução apresentada por Mikolov (2007) conseguiu reduzir este custo computacional de forma logarítmica.

2.4.1 Exploração de relações semânticas entre as palavras

Uma propriedade interessante que a representação distribuída das palavras neste novo espaço vetorial possui é representar relações sintáticas e semânticas entre as palavras através de operações vetoriais simples. Por exemplo, ao somar dois vetores (que são as representações distribuídas para as palavras) e achar qual é o vetor-palavra mais próximo deste vetor soma, o resultado é uma palavra com uma relação de similaridade/semântica

muito forte.

Recentemente, um trabalho publicado por Wang et al. (2015) mostrou que um modelo treinado para resolver um determinado tipo de questão (envolvendo por exemplo, analogias, antônimos, sinônimos, etc) utilizando representações distribuídas não apenas consegue resultados superiores aos métodos existentes mas também supera a pontuação média de voluntários que respondem através da ferramenta *Amazon Mechanical Turk*¹.

Em um modelo treinando para a língua inglesa, Mikolov et al. (2013c) mostrou que ao relizar a operação “King” + “Woman” - “Man” o vetor-palavra mais próxima desta é a palavra “Queen”, que é a figura feminina do rei em uma monarquia. Realizar esta operação pode ser visto como “King está para Man assim como Queen está para Woman”. É importante lembrar que em nenhum momento é utilizado uma base de conhecimento, além dos próprios textos que são treinados o modelo.

É possível visualizar a distribuição destes vetores geometricamente em um espaço bi-dimensional realizando uma redução de dimensionalidade através da Análise de Componentes Principais (*Principal Component Analysis* - PCA). A figura 2.3 mostra a posição de algumas palavras - e algumas palavras correlatas - na sua representação distribuída bi-dimensional. E como pode-se observar, existe um padrão evidente entre as palavras e a semântica associada a esta palavra. Algumas aplicações envolvendo representação distribuída já mostraram que independente do idioma utilizado, as palavras e os conceitos representados são agnósticos quanto a linguagem, inclusive sendo utilizado em modelos para traduções entre línguas (Mikolov et al., 2013b).

¹<https://www.mturk.com/mturk/welcome>

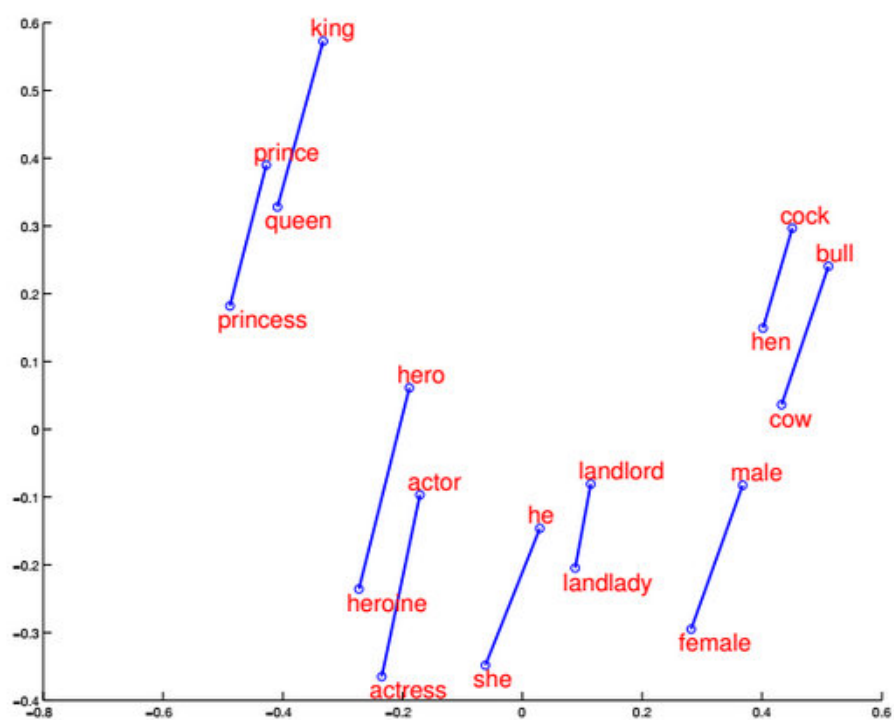


Figura 2.3: Posição das palavras utilizando a representação distribuída em um espaço bi-dimensional criado pelo PCA. (Mikolov et al., 2013c)

Capítulo 3

Metodologia

3.1 O Processo de captura do Media Cloud

A captura dos textos que alimentam o *Media Cloud Brasil* é um processo contínuo de varredura da web, de forma a estar sempre atualizado e com a maior cobertura possível das notícias que são publicadas. O processo divide-se em 3 etapas principais, descritas abaixo.

- Definição da mídia

Dado um conjunto de mídias, precisa-se descobrir feeds de notícias relacionado a cada uma das mídias selecionadas. Para ter a maior cobertura possível, o banco de dados é alimentado continuamente através de buscas no google por palavras chaves, capturando e armazenando então os *feeds RSS* das páginas mais populares. Atualmente são monitorados cerca de 150.000 endereços de feeds RSS.

- Crawling

Uma vez com os endereços de *feeds* relacionados às diferentes fontes de notícias, um *crawler* (robô) acessa esses feeds 6 vezes ao longo do dia coletando as notícias publicadas e armazenando o código fonte (código HTML) em um banco de dados.

Dentre as informações que são capturadas estão título da matéria, fonte que foi coletada, data de publicação e link.

- Extração de texto

Após a coleta dos dados, deve-se extrair o conteúdo real dos textos, isto significa não apenas a remoção do código HTML da páginas, mas também filtragem de anúncios, layout, *copyrights* e termos de uso contidos como texto no código HTML da página.

Porém nem tudo que é adquirido deve ser utilizado uma vez que não existe nenhum controle sobre a qualidade do conteúdo que é capturado. Segue abaixo algumas das dificuldades envolvidas no processo de captura.

Falha no processo de limpeza do HTML: O processo de extração de conteúdo necessita detectar dentro de um HTML genérico, o que de fato é conteúdo e o que é código HTML. No processo de captura, este processo é fundamental, já que caso a limpeza não seja feita da forma esperada, tudo que restará para a análise posterior é código HTML, que não possui nenhum valor semântico. Nesta etapa deve-se remover os cabeçalhos e rodapés das páginas, propagandas e colunas adicionais, restando assim apenas o conteúdo. No projeto *Media Cloud Brasil* esta etapa é feita pela biblioteca *Goose*¹, que faz uma remoção do cabeçalho/rodapé e em seguida tenta inferir, via contexto, onde está o conteúdo da página.

Problemas de codificação: Como padrão, a codificação das páginas deveria ser UTF-8, mas nem todas as páginas seguem o padrão esperado. Assim, existe uma quantidade de artigos que não podem ser utilizados para a análise já que possuem problemas de codificação. Palavras que contenham acentos e/ou caracteres especiais tornam a análise inviável neste caso.

¹<https://github.com/grangier/python-goose>

Falta de controle sobre o conteúdo obtido: Aqui podemos citar três subproblemas recorrentes. O primeiro é pelo fato de alguns sites tratarem os comentários como um outro *feed* de notícias e isso faz com que sejam armazenados comentários fora do contexto em que foi publicado, tornando inviável fazer inferências/análises sobre este conteúdo. O segundo problema é sobre a falta de controle sobre os feeds que são capturados, implicando na captura de notícias provenientes de outras línguas (que não o português). E o terceiro problema refere-se a captura de *spam*, que no caso do *Media Cloud Brasil* revelou ter uma grande concentração de anúncios vendendo remédios controlados.

Estima-se que aproximadamente 10% do conteúdo capturado pelo *Media Cloud Brasil* esteja dentro de um destes problemas mencionados acima. Posteriormente propomos um método para a identificação e remoção deste conteúdo.

3.2 Processamento de Linguagem Natural

3.2.1 O modelo “*Bag-of-words*”

Uma das implicações dos modelos tradicionais é a representação dos dados com o modelo *bag-of-words*. Utilizando esta representação a ordem das palavras torna-se irrelevante. É importante ressaltar que com o modelo *bag-of-words* estamos supondo o pressuposto de permutabilidade. Utilizando este modelo as frases abaixo teriam o mesmo significado.

...fui a padaria e comprei pão...

...e comprei fui padaria pão a...

...a pão comprei padaria e fui...

3.2.2 Pré-processamento do texto

Uma pequena parcela das palavras contidas em um texto realmente consegue refletir as informações contidas no mesmo. Analisando a língua portuguesa, podemos dizer que palavras como “e”, “de” e “seus” possuem pouco ou nenhum valor semântico associado e são conhecidas como *stop words*. A remoção das *stop words* é uma tarefa trivial em qualquer método com objetivo de analisar textos pois ninguém deseja achar relações entre palavras como “de” e “uma”, pois estas palavras não carregam consigo nenhum tipo de valor semântico. De forma análoga, palavras como “estudante”, “estudo” e “estudei” possuem em comum o fato de representarem de forma genérica o significado de “estudar”. Além disso, elas são diferenciadas apenas por variações afixais (prefixo e/ou sufixo). Existem algoritmos que buscam tratar estas palavras de forma a reduzi-las apenas a seu radical, conhecidos como algoritmos de radicalização.

Stop Words

Esta é uma técnica que visa remover termos pouco significativos para melhorar a análise textual. Entretanto, estes representam a maioria dos termos nos documentos. É importante lembrar que as *stop words* não são um problema tipicamente português-brasileiro e sim um problema geral envolvendo as diferentes línguas, por exemplo, no caso da língua inglesa podemos pensar em palavras como “of” e “the”. Pode-se obter uma lista de palavras consideradas *stop words* analisando-se os textos que serão utilizados ou fazendo um estudo do idioma a ser considerado. No quadro abaixo é exibida uma lista de *stop words* para o vocabulário português-brasileiro.

3.3 O Modelo N-Grama

N-gramas podem ser definidos como uma subsequência de n itens construídos a partir de uma sequência de itens (Brown et al., 1992). Este é um modelo baseado em

Tabela 3.1: Lista com alguns exemplos de *stop words*

de	o	minhas	tenho	deles	no	na	já	seu
os	se	às	suas	isto	teus	nosso	até	ela
tua	tuas	nos	está	um	à	te	seja	elas
tem	tu	pela	isso	eles	você	aos	esse	estas
estão	por	havia	pelos	do	em	Dos	mas	nós
da		me	estes	vos	meus	ou	quando	sem
lhes	As	como	tinha	mais	esses	aquela	aquelas	essa
essas	sua	ser	depois	mesmo	pelas	era	nossos	fosse
e	aquele	aqueles	foram	num	com	uma	têm	qual
é	entre	nossa	este	dele	Ao	das	também	pelo
Foi	não	vocês	para	será	dela	esta	seus	nas
nossas	Ele	eu	só	minha	há	teu	lhe	numa
muito	delas	ter	quem	a	que	nem	meu	aquilo

uma *cadeia de Markov* onde a probabilidade de seleção de uma palavra é condicionada as palavras anteriores. O tamanho das sequências pode ser arbitrário, quando, por exemplo bigrama refere-se a n-gramas de tamanho 2, trigramas referem-se a n-gramas de tamanho 3 e assim consecutivamente.

Os modelos que utilizam n-gramas são muito populares devido a sua simplicidade e geralmente apresentam boa performance, inclusive por muito tempo foram utilizados como os principais modelos de linguagens, porém a criação de n-gramas para a utilização posterior no vocabulário faz com que este aumente exponencialmente, o que implica em um custo computacional de ordem maior e que por vezes pode ser inviável. Neste trabalho optou-se por utilizar n-gramas para a criação de frases - conjunto de palavras que co-ocorrem juntas - e desta forma evitar o crescimento exponencial do vocabulário.

3.3.1 Criação de frases com o modelo N-Gram

Representações tradicionais de palavras possuem a limitação de representar apenas composições individuais de palavras, não sendo possível a representação de frases. Por exemplo “Rio de Janeiro” é um estado do Brasil, sendo seu significado completamente

diferente da combinação das palavras rio (do verbo rir) e janeiro (primeiro mês do calendário).

A extensão do modelo baseado em palavras para o modelo baseado em frases é relativamente simples. Primeiro são identificados um grande número de palavras que ocorrem conjuntamente em um dado contexto, mas que não são frequentes em outros contextos. Por exemplo as frases “Rio de janeiro” e “Polícia Militar” devem representar uma “única palavra” no conjunto de documentos, enquanto que “de acordo” continuará sem sofrer modificações, ou seja, será representado por duas palavras, “de” e “acordo”.

Desta forma pode-se formar uma quantidade significativa de frases sem aumentar muito o tamanho do vocabulário. Em teoria, o modelo proposto poderia ser treinado utilizando-se todos os possíveis n-gramas, porém a utilização dos recursos computacionais neste caso seria exponencialmente maior. Muitas técnicas já foram desenvolvidas anteriormente para a identificação de frases no texto, porém a proposta deste trabalho não é comparar estas diferentes técnicas. Este trabalho utiliza uma técnica direcionada unicamente aos dados, onde frases são formadas baseadas nas frequências de até trigramas, utilizando

$$score(w_i, w_j) = \frac{count(w_i w_j) - \delta}{count(w_i) \times count(w_j)} \quad (3.1)$$

o termo δ é usado como um coeficiente de penalização e previne que muitas frases contendo poucas palavras sejam formadas. Os n-gramas com um score acima de um determinado valor estabelecido são então utilizados como frases e este processo faz com que o número de frases adicionadas ao vocabulário não cresça tanto, diferente do caso em que seriam gerados todos os n-gramas possíveis.

3.4 O Modelo *Skip-Gram*

3.4.1 Representação distribuída de palavras

A ideia de utilizar relações distribuídas entre as palavras foi um dos pontos altos como reutilização de redes neurais artificiais na década de 1980. A ideia por trás da representação distribuída foi introduzida utilizando como referência a representação cognitiva: Um objeto mental pode ser representado eficientemente caracterizando este objeto com poucos atributos, sendo cada um destes representados por dois estados: ativos e inativos.

3.4.2 *Skip-Gram*

O *skip-gram* é um modelo introduzido por Mikolov et al. (2013a) com o objetivo de achar a representação de palavras que seja útil para prever palavras próximas a esta - que pode ser chamado de contexto. Basicamente este modelo é uma rede neural com uma camada oculta W e uma camada de saída, que chamaremos de W' . É importante notar que neste caso, W' não é a matriz W transposta e sim uma outra matriz. A figura 3.1 exibe o modelo para um contexto de tamanho C , com um vocabulário de tamanho V e uma camada oculta de tamanho N . O vetor de entrada é codificado de forma que as palavras que estão dentro daquele contexto definido assumirão o valor 1 e todos os outros valores serão 0. O objetivo do modelo é minimizar a log verossimilhança negativa, escrita como

$$E = -\log p(w_{O,1}, w_{O,2}, \dots, w_{O,C} | w_I) \quad (3.2)$$

$$\begin{aligned} &= -\log \prod_{c=1}^C \frac{\exp(u_{c,j_c^*})}{\sum_{j'=1}^V \exp(u_{j'})} \\ &= -\sum_{c=1}^C u_{j_c^*} + C \cdot \log \sum_{j'=1}^V \exp(u_{j'}) \end{aligned} \quad (3.3)$$

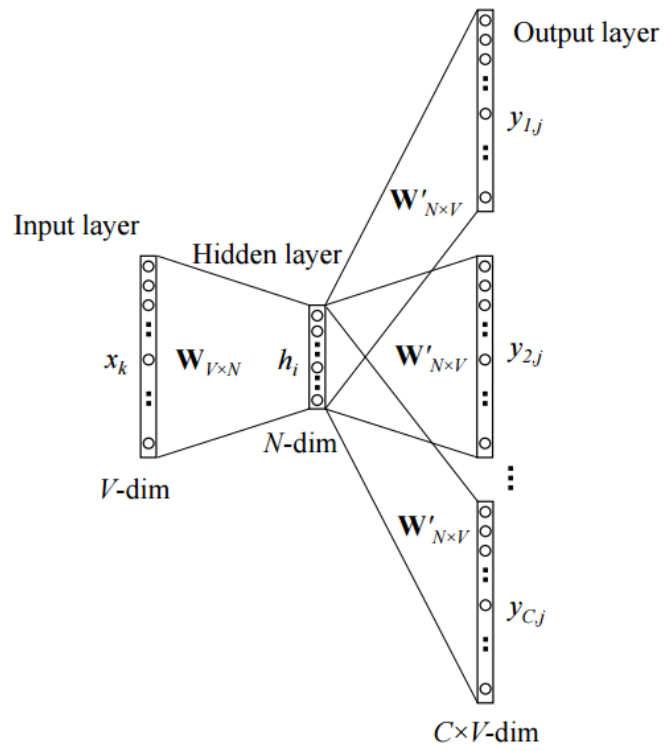


Figura 3.1: O modelo *skip-gram*. (Rong, 2014)

onde C é o tamanho do contexto escolhido e V são as palavras do vocabulário. A escolha de um C maior requer uma quantidade maior de dados de treino e pode levar a uma maior precisão, ao custo de um maior tempo de treinamento. A formulação básica do modelo *Skip-gram* define $p(w_O|w_I)$ utilizando a função softmax:

$$p(w_O|w_I) = \frac{\exp(v_{w_O}^T v_{w_I})}{\sum_{j'=1}^V \exp(v_{j'}^T v_{w_I})} \quad (3.4)$$

onde v_w e v'_w são as representações vetoriais do “input” e “output”, respectivamente e V é o número de palavras no vocabulário. Esta formulação é impraticável por conta do custo de calcular $\nabla \log p(w_O|w_I)$ é proporcional a V , que costuma ser grande (entre 10^5 e 10^7 termos). Para resolver este problema uma alternativa é optar por aproximações ao *softmax* que tornem o processo de cálculo mais simples e neste caso vamos utilizar o *softmax* hierárquico.

3.4.3 *Softmax* Hierárquico

Uma vez que utilizar o modelo softmax completo é inviável devido ao alto custo computacional, deve-se utilizar técnicas de aproximação para o cálculo das probabilidades de seleção de cada uma das palavras e uma destas técnicas que é eficiente do ponto de vista computacional é o *softmax* hierárquico. No contexto de redes neurais de modelos de linguagens, este modelo foi proposto inicialmente por [Morin e Bengio \(2005\)](#). A maior vantagem deste método é que ao invés de avaliar V nós de outputs para obter a distribuição de probabilidade, precisa avaliar apenas $\log_2(V)$.

Este modelo hierárquico utiliza uma representação de árvore binária na camada de output com V palavras como suas folhas e, para cada nó, representa explicitamente as probabilidades relativas dos nós filhos. Isto define um caminho aleatório que atribui probabilidades a palavras.

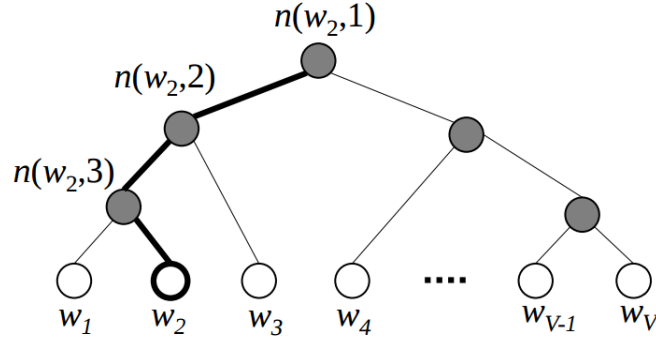


Figura 3.2: Exemplo de árvore binária utilizando *softmax* hierárquico. Os nós brancos são palavras no vocabulário e os escuros não os nós internos. Um exemplo de caminho até a palavra w_2 é destacado e o tamanho do caminho $L(w_2) = 4$. $n(w, j)$ indica o j -ésimo nó no caminho da raiz até a palavra w . (Rong, 2014)

$$p(w_O | w_I) = \prod_{j=1}^{L(w)-1} \sigma \left(\llbracket n(w, j+1) = ch(n(w, j)) \rrbracket \right) \cdot v_{n(w, j)}^T v_{w_I} \quad (3.5)$$

onde $ch(n)$ é o nó filho a esquerda do nó n ; $v'_{n(w, j)}$ é a representação (“vetor de saída”) do nó interno $n(w, j)$; v_{w_I} é o valor de saída da camada oculta; $\sigma(x)$ é a função sigmóide e $\llbracket x \rrbracket$ é uma função especial, definida como

$$\llbracket x \rrbracket = \begin{cases} 1 & \text{se } x \text{ é verdadeiro;} \\ -1 & \text{caso contrário.} \end{cases} \quad (3.6)$$

Pode-se entender a equação (3.5) através de um exemplo. Suponha - olhando para a figura 3.2 - que deseja-se calcular a probabilidade de w_2 seja a palavra de saída. Pode-se definir esta probabilidade como a probabilidade de um caminho aleatório começando da raiz terminar na folha em questão. A cada nó que é percorrido (incluindo o nó raiz), precisa-se atribuir probabilidades de ir para a esquerda ou direita. A probabilidade de

ir para a esquerda em um nó interno n é

$$p(m, left) = \sigma(v_n'^T \cdot v_{w_I}) \quad (3.7)$$

que é determinado tanto pela representação vetorial do nó interno $v_n'^T$ e o valor camada oculta v_{w_I} (que é determinado pela representação vetorial da palavra de entrada). Já a probabilidade de ir para o nó da direita é

$$p(m, right) = 1 - \sigma(v_n'^T \cdot v_{w_I}) = \sigma(-v_n'^T \cdot v_{w_I}) \quad (3.8)$$

Seguindo o caminho da raiz até a palavra w_2 , como na figura 3.2, pode-se calcular a probabilidade da palavra w_2 ser a saída como

$$p(w_2 = w_O) = p(n(w_2, 1), left) \cdot p(n(w_2, 2), left) \cdot p(n(w_2, 3), right) \quad (3.9)$$

$$= \sigma(v_{w_2,1}'^T v_{w_I}) \cdot \sigma(v_{w_2,2}'^T v_{w_I}) \cdot \sigma(-v_{w_2,3}'^T v_{w_I}) \quad (3.10)$$

que é exatamente o mesmo resultado da equação (3.5), também é possível verificar que

$$\sum_{i=1}^V p(w_i = w_O) = 1 \quad (3.11)$$

fazendo o *softmax* hierárquico uma distribuição multinomial bem definida sobre o vocabulário.

A partir das equações de atualização, pode-se ver que a complexidade para treinar o modelo cai de $O(V)$ para $O(\log(V))$, que é uma melhora considerável em termos de número de iterações. A estrutura da árvore utilizada pelo *softmax* hierárquico possui um efeito considerável em sua performance. Mnih e Hinton (2009) exploraram inúmeros métodos para a construção da estrutura da árvore e contabilizaram os efeitos tanto no tempo gasto no treino como na respectiva precisão de cada um dos métodos.

3.5 Clusterização - *K-means*

O objetivo da clusterização de dados, também conhecido como análise de cluster é descobrir o agrupamento natural de um conjunto de objetos. Este é um dos algoritmos mais utilizados de clusterização, devido a sua extrema simplicidade e facilidade de implementação, mesmo tendo sido descoberto independente por várias áreas do conhecimento (Steinhaus (1956), Lloyd (1982), Ball e Hall (1965) e MacQueen et al. (1967)) há mais de 50 anos.

Seja $X = x_i, i = 1, 2, \dots, n$ um conjunto de n pontos d -dimensionais que serão clusterizados em um conjunto de K clusters. O algoritmo *K-means* particiona os dados de forma que o erro quadrático entre a média sadasdasd seja minimizada. Seja μ_k a média do cluster c_k . E erro quadrático entre μ_k e os pontos pertencentes ao cluster c_k pode ser definido como

$$J(c_k) = \sum_{x_i \in c_k} \|x_i - \mu_k\|^2. \quad (3.12)$$

O objetivo do algoritmo é minimizar a soma do erro quadrático sobre todos os K clusters

$$J(C) = \sum_{k=1}^K \sum_{x_i \in c_k} \|x_i - \mu_k\|^2. \quad (3.13)$$

O método inicia com centróides geradas aleatoriamente no espaço d -dimensional. Uma vez que o erro quadrático tende a diminuir com o aumento no número de clusters K (assumindo o valor zero quando o número de clusters é igual ao número de observações no conjunto de dados) isto pode ser feito apenas para um número fixo de número de clusters. Os principais passos do algoritmos são:

1. Geram-se K centróides.
2. Calcula-se a distância euclidiana a atribui-se cada observação ao cluster com a menor distância.

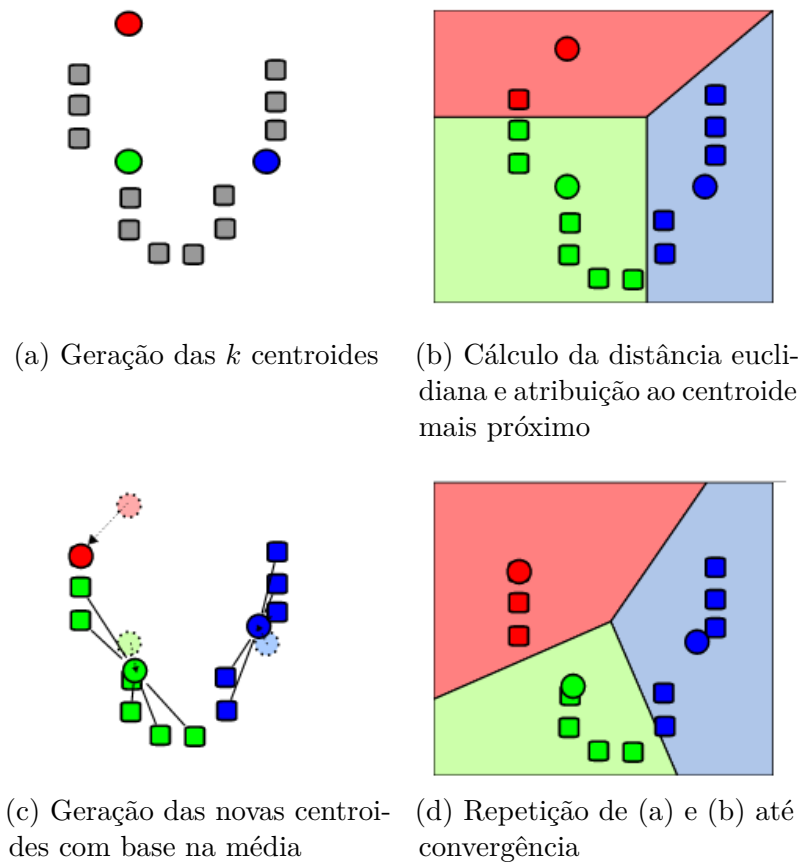


Figura 3.3: Ilustração gráfica do algoritmo *k-means*. ([Wikipedia, 2014](#))

3. Calculam-se as novas centróides com base nas médias das observações que formam aquele cluster.
4. Repetem-se os passos anteriores até convergência.

A figura 3.3 exhibe como funciona este procedimento.

3.6 Estratégia para a identificação não-supervisionada de grupos

Para a identificação de grupos de maneira não-supervisionada utilizamos a seguinte metodologia: Primeiro estima-se o modelo *skip-gram* com uma camada oculta de 300 atributos (representação distribuída das palavras com alto valor semântico) e em seguida aplica-se um algoritmo de clusterização para que as palavras com valor semântico parecido fiquem no mesmo grupo. Feito isso, para cada documento calcula-se a proporção de cada um dos *clusters* de palavras, ponderando o peso inversamente proporcional à quantidade de palavras naquele grupo.

3.7 Softwares Utilizados

Este trabalho foi implementado em Python e utilizou a biblioteca Gensim ([Řehůřek e Sojka, 2010](#)).

Capítulo 4

Resultados

4.1 Descrição dos dados

Para este trabalho foram selecionados todos os artigos capturados pelo *Media Cloud Brasil* entre 05/2013 e 10/2015. O número total de artigos utilizados foi 1.518.626 .

Como existe o projeto de qualidade do conteúdo capturado o resultado será dividido em duas partes: A primeira tem como objetivo a identificação e limpeza do conteúdo que não seja apropriada para a análise e inclui uma curadoria nos dados fornecidos pelo *Media Cloud Brasil* e a segunda tem como objetivo criar os grupos onde palavras tenham uma semântica similar, tornando possível a classificação dos documentos com base na proporção de cada grupo.

Para a criação das frases foram utilizado até trigramas. A tabela 4.1 exibe uma lista de frases que foram identificadas utilizando esta metodologia.

flappy bird	capitais brasileiras	recôncavo baiano	cartão pré pago
operação lei seca	azul linhas aéreas	shopping center	smartphones android
santa missa	rede globo	centro oeste	câmbio manual
campeonato pan americano	projeto tamar	hospital regional	cadastro gratuito
grupo rbs	região amazônica	lençol freático	inteligência artificial

Tabela 4.1: Exemplo de frases criadas com a utilização de até trigramas

4.2 Aplicando o modelo *skip-gram*

A primeira iteração do modelo teve o objetivo de identificar artigos com algum tipo de problema (descritos na seção 3.3.1). Para isto, treinou-se um modelo *skip-gram* com uma camada oculta composta de 300 unidades e um tamanho de contexto igual a 5. As palavras que apareceram menos de 100 vezes foram excluídas da análise, restando no final um vocabulário com 101.680 palavras/frases diferentes.

4.2.1 Explorando relações entre as palavras

Como mencionado anteriormente, a representação distribuída permite encontrar relações semânticas entre as palavras. Utilizando os textos capturados pelo *Media Cloud Brasil* pode-se encontrar relações entre as palavras que seguem.

1. Pelé + Argentina - Brasil = Maradona
2. Neymar + Argentina - Brasil = Messi
3. São Paulo + Argentina - Brasil = Buenos Aires
4. Inglaterra + Paris - França = Londres
5. Itália + Paris - França = Roma
6. Fox News + Brasil - Estados Unidos = Rede Globo
7. Harvard + Brasil - Estados Unidos = UFRJ
8. Areia + Montanha - Praia = Neve
9. Rei + Mulher - Homem = Rainha
10. Maior + Pequeno - Grande = Menor
11. Touro + Cadela - Vaca = Cão

12. Ele + Atriz - Ela = Ator
13. Ela + Homem - Ele = Mulher
14. Bonito + Sujo - Feio = Limpo
15. Vencer + Perdeu - Venceu = Perder

Observou-se que o modelo é sensível a detecção de sinônimos (“Ela está para atriz assim como ele está para ator.”), antônimos (“Bonito está para feio assim como limpo está para sujo.”) e flexões verbais (“Vencer está para venceu assim como perder está para perdeu.”).

Aproveitando esta característica do modelo de inferir relações semânticas entre palavras, podemos a partir de uma palavra criar um grafo com as palavras as quais são também relacionadas a esta, seja diretamente ou indiretamente - através de palavras que diretamente possuem um alto grau de associação. A figura 4.1 mostra a relação da palavra “violência” com palavras similares e como resultado temos as palavras “violência sexual”, “violência policial”, “discriminação”, “criminalidade”, “violência doméstica”, “intolerância”, “xenofobia”, “violência física” e “barbárie”.

4.2.2 Limpeza e classificação do conteúdo através do modelo *skip-gram*

Após a modelagem, utilizou-se o algoritmo *k-means* para agrupar as palavras que tinham significado semântico parecido. O número de grupos de palavras selecionado foi $k = 30$.

Seguem alguns exemplos de textos que tiveram um alto percentual de palavras dos referidos grupos. Quanto menor a quantidade de palavras em um cluster, mais particular ele é e conseqüentemente mais evidente identificar o assunto do qual é abordado.

cluster 0: Economia (3.302 palavras)

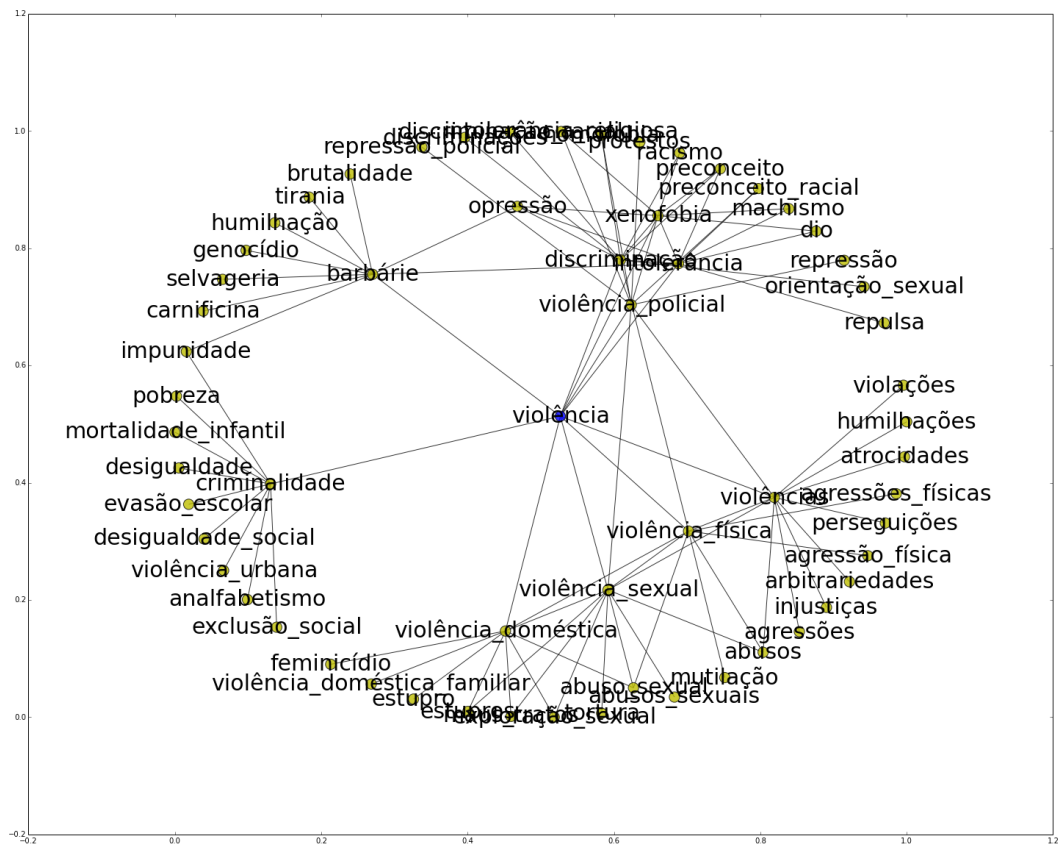


Figura 4.1: Palavras associadas a palavra central “violência”, com profundidade máxima de 2 palavras.

Exemplo de palavras: atingiu, financiamentos imobiliários, muito aquém, média histórica, mede desempenho, juros impostos amortização depreciação, crédito rotativo, fii, compra pesos, volume produzido, alugueis, reduzirá, vgv, divulgado nesta sexta feira, medido, houve piora, dois dígitos, atingindo marca, ipc c, principais influências, orçamento doméstico, números fracos, longo prazo tjlp, litro, ter aumentado, intradia, crédito, dólar vista, setor automotivo, us milhões corrente, ficou estável, consolidação fiscal, dólares pelas atuações, analistas esperavam, previsão inicial era, hortaliças legumes, batendo recordes, pressões inflacionárias, pressão inflacionária, gwh, subiu ponto percentual, equivalente us, ao consumidor amplo, us galão, dólar recuou, levantamento anterior, ajudadas, vendas totais, vem diminuindo, expansão

Exemplo de artigo classificado:

A confiança do empresário do comércio evoluiu em um ritmo desfavorável no trimestre encerrado em novembro.
O Índice de Confiança do Comércio (Icom), divulgado nesta sexta-feira, 29, pela Fundação Getulio Vargas (FGV)...

cluster 2: Línguas estrangeiras (1.430 palavras)

Exemplo de palavras: fuera, fin, fil, hiyo, fra, unos, hai, hal, han, hao, hay, información, slideshare, tienen, rosso, sang, sana, en el, ministerio, video, multimedia, cuatro, manos, tapi, aec, aee, indice, ouml, ts, cms, eri, ogni, bahwa, huang, har, mio, gigi, gut, iba, ibu, bajo, ha sido, border src http s, deben, order by, ade, ada, gq, sich, anatomy game of thrones, abdallah, todas las, class attachment article, rc htm rel nofollow, través, puede, amp, auf, aus, med, mem, mei, mes, creo, comunicación, sebuah, amp link http fct, bon, sni, auch, ger, gen, punto, tout, gv, ge, gc, gn, entonces, gif, valet, class ot hashtag href, program files

Exemplo de artigo classificado:

Ma, dopo le dimissioni di ieri, Miliband stato costretto a un veloce rimpasto del suo Gabinetto ombra, installando l ministro dell ombra Ed Balls al posto di Johnson e la moglie di Balls, Yvette Cooper, alla poltrona occupata fino a ieri dal marito

cluster 3 e 26: Spam de farmácia (1.196 e 1.102 palavras respectivamente)

Exemplo de palavras: the brand name, caffeine and, style background cfc color, smiletemplates, around the clock pain, gt div td tr, gt div td td, disease interactions with, professional monograph drugs, drugs containing doxycycline vibramycin, sleep latency and improved, chemical name dihydro, procedures that may be, answers blog group members, adhd meds to undiagnosed, generator, august strength mg ab, ffa color black font, dioxo naphthacenecarboxamide monohydrate molecular, with symptoms such, oxycodone major drug interactions, severe pain it belongs, sob exame concedo, mg oxycodone extended, name eszopiclone dosage form, system cns to relieve, moderate drug doxycycline calcium, cholesterol or side effectspregnancy, false, drug interactions minor lunesta

Exemplo de artigo classificado:

amlodipine besylate picture, amlodipine 5 mg cost,
amlodipine benazepril hydrochloride, amlodipine besylate
maximum dosage, amlodipine dosage side effects,
amlodipine dose related side effects,

The Most Trusted online Drug Store for Amlodipine

cluster 4 e 23: Entretenimento (7.486 e 1.472 palavras respectivamente)

Exemplo de palavras: moderno, gênero musical, castelo rá tim bum, diurno, palma, ballet, truques, moda praia, sanfoneiro, sob batuta, cantor compositor, superpoderes, sesc belenzinho, charles chaplin, atração musical, lirismo, glauber rocha, tule, jia, ernest hemingway, didi, narradora, especiarias, chico xavier, mescla, zoo, gilberto gil, bradley cooper, bolshoi, imagens sons, brisa, cordas, carimbó, bruce springsteen, seguidor, vendeu, recordações, néctar, american horror story, inseparável, obra prima

Exemplo de artigo classificado:

Para Chanel atemporalidade é a palavra certa e para quem não assistiu ao vídeo "Chanel Particuliere" de outono-inverno 2010, não sabe o que perdeu. Animado e com estilo stop-motion, o vídeo foi dirigido por Quentin Jones. É simplesmente encantador, assistam!

cluster 5: Política (2.266 palavras)

Exemplo de palavras: discursou, senador lindbergh farias pt, ataques pessoais, paz eu odeio, acm neto, petista fernando pimentel, comícios, decreto assinado, peter siemens, mário covas psdb, deputado socialista, jorge viana, entrevista rádio metrópole, pfl, senador lindbergh farias, núcleo duro, prefeitos, ungido, márcio thomaz bastos, marta, governo wagner futura, tenho apreço, governar, peemedebistas, prefeito eduardo paes, assumir presidência, rosalba, presidida, tarso genro, governo eles construíram, presidencialável aécio neves, sufrágios, principais articuladores, arrecadou milhões, gestões, acompanhar apuração, deputado federal beto albuquerque, josé sérgio gabrielli, henrique capriles, presidencialável tucano, senador cícerio lucena psdb, deputado carlos sampaio, prefeito marcelo rangel, púlpito, subiu tribuna

Exemplo de artigo classificado:

De acordo com pesquisa Datafolha, a presidente Dilma Rousseff seria reeleita no primeiro turno se disputasse a eleição contra os dois candidatos mais prováveis do PSDB e do PSB, Aécio Neves e Eduardo Campos.

cluster 6: Nomes de pessoas (3.497 palavras)

Exemplo de palavras: silva, mori, mora, patriarca, schettino, hugo, viana, isaac, susana, lourival, eugenia, gilberto, tamires, sebastiana, banco central carlos hamilton, katia, erico, erick, erica, sávio, bianca, bianco, ex administrador, alejandro, alejandra, jurista, biondi, alessandra, araujo, cruz, schmitt, sobrinha, abade, neto pb, defensor público, janaina, virgilio, colégio estadual, arcebispo, petry, lessa, bandido, bra, heloisa, padre, schneider, serra branca, louzada, daniel, lidiane, reitora, fogaça, publicitária, marinho, dr josé, félix, luiz fernando, babau, toni, aquino, priscilla, edmar, abordará tema, francisco soares, jacques, heller

Exemplo de artigo classificado:

Antônio desconfia de Palhares. Hernandez cobra explicações de Maura sobre o bilhete de Tita. Palhares afirma a Antônio que o menino é ideal para uma missão. Luciana distribui panfletos que anunciam sua creche no casarão, e Omar prevê problemas.

cluster 9, 28 e 29: Direito (5.266, 414 e 556 palavras respectivamente)

Exemplo de palavras: determinou suspensão, danos decorrentes, interrogatório, pessoa física irpf, testemunhas arroladas, sergio moro, dias estava analisando, evitar fraudes, desconstituição, justiça, decretos, medidas alternativas, ordenamento

jurídico brasileiro, cassados, reformada, obedecendo, documentação exigida, livre concorrência, imposto sobre serviços iss, caráter permanente, comissao, cláusulas contratuais, benesse, inquéritos, concordância, rgãos, parâmetros, política criminal penitenciária, irregularidades cometidas, processo administrativo n^o, techint, reeducando, sujeito passivo, leis, práticas criminosas, representantes eleitos, exigido, estará sujeito, corregedoria regional, sentido estrito, amparo, firmar contrato, hipossuficiência, resguardar, bahia mp ba, houve irregularidade

Exemplo de artigo classificado:

A pensão alimentícia judicial ou homologada em cartório
está sujeita ao recolhimento mensal pelo carnê-leão
(desde que superior ao limite de isenção, de R\$ 1.787,77
neste ano) e à tributação na declaração anual...

cluster 10: Código HTML que não conseguiu extrair o conteúdo (1.411 palavras)

Exemplo de palavras: px font weight normal, baseline h, div class attachments
class, style background image none, http static ow ly, family inherit font size, here
br br, co uk, px text decoration none, margin top px padding, epessoas, afredes,
px padding px vertical, jpg title file in, div class msonormal style, color ce, font
family helvetica

Exemplo de artigo classificado:

```
<table> <tr> <td rowspan='2' valign='top' width='150'>
<a href='http://virgula.uol.com.br/audio/panico/leo-linswfgs'
target='_self'>
```

cluster 11: Notícias internacionais (2.621 palavras)

Exemplo de palavras: china continental, pós troika, França François Hollande, união africana, delegações, niato, negociações nucleares, São Petersburgo, Alexei, BBC Brasil, México Chile, Durban, nuclear, subdiretor, Saleh, assuntos políticos, France Presse, Abadi, Thomas Bach, dupla cidadania, Zhou, Unido Rússia, Petro, Nicolas Sarkozy, outras nações, fonte próxima, Moçambique, Maduro, mudanças climáticas IPCC, Reino Unido David Cameron, jornal francês Le, integração europeia, secretário geral, Minsk, Timor Leste, libertação duas edições, comissário, comissária, Mujica, Marroquino, fazenda Guido Mantega, relações externas, segurança cibernética

Exemplos:

O governo sul-africano está ciente de que o falso intérprete de sinais que participou da missa campal em homenagem a Nelson Mandela é acusado de homicídio e informou nesta sexta-feira (13) que ele está sendo investigado.

cluster 12: Violência internacional (2.697 palavras)

Exemplo de palavras: líderes ocidentais, exército israelita, destruiu, soldados norte americanos, combatendo, protestos nas ruas, explosão ocorreu, Irã, crianças mortas, meridional, combatentes rebeldes, abandonaram, grupos separatistas, deflagrou, dignitários, reforçar segurança, violência sectária, Falluja, entre Ucrânia Rússia, Honduras, desaparecimento, hospital presbiteriano, veículo blindado, foram assassinadas, matando, armas pesadas, bases militares, exército nigeriano, defesa aérea, sharia lei islâmica, iraquianas, se suicidou.

Exemplos:

Um ataque atribuído ao grupo islâmico Boko Haram matou oito pessoas e feriu várias outras neste domingo, durante uma festa de despedida de solteiro na aldeia de Tashan-Alede, na Nigéria.

cluster 14: Negócios (1.054 palavras)

Exemplo de palavras: mmx mineração mmxm, natura, grupo ebx, subsidiárias, tóquio fechou, pilgrim pride, casas bahia, prumo, figurou entre, cia hering, rodobens, leves ganhos, inglaterra boe, dow jones, disse estrategista, balanço trimestral, repercutindo, moeda norte americana, cvc, bolsas europeias fecharam, debentures, jp morgan, brf, altice, blackrock, investidor aconselhável buscar, penalizadas, itaú unibanco, teve prejuízo líquido, entre ganhos perdas, qgep, principais bolsas, mercados norte americanos.

Exemplos:

O Conselho de Administração da MMX aprovou nesta
segunda-feira a proposta de grupamento de ações da
companhia à razão de seis para uma.

cluster 15 e 21: Esporte (2.765 e 3.571 palavras)

Exemplo de palavras: arena pe, cumprir suspensão automática, bola parada, fig, mário bittencourt, chapinha, torrico, contra penapolense, técnico alexandre gallo, tomou iniciativa, driblou goleiro, contusão muscular, gol defendido, peça fundamental, treinador alvinegro, diretoria alvinegra, atlético mg victor marcos, rasgou elogios, num chute, esquerdinha, ananias, zagueiro vitor hugo, anotado, jogadas individuais, audax, william alves, deixou desejar, errava, palestra itália, rhayner, vacilos, vacilou, havia sofrido, goleador

Exemplos:

O Santos guerreiro enfrentou o São Paulo apático e venceu
sem dificuldade. Deve ter sido um dos jogos mais fáceis

para o Peixe, após a paralisação do campeonato brasileiro por causa da Copa das Confederações, dentro de sua casa.

cluster 16, 20 e 27: Língua inglesa (2.317, 2.163 e 5.911 palavras respectivamente)

Exemplo de palavras: kids, stern, the mirror, yahoo, fip, master, genesis, wal mart, tech, bielsa, nature, fry, palazzo, crowe, idol, john kennedy, chair, ufc, benson, doherty, hyatt, jimmy, jin, kang, turner, zoe, dexter, fort lauderdale, hashtag, traffic, weidman, cream, natalie, sand, one direction, muhammad ali, paper, scott, saint.

One of the best restaurants I have ever been to, worth the price.
Food is awesome, service is friendly ...
remember to reserve though or you could wait for over an hour.

cluster 17: Educação (1.793 palavras)

Exemplo de palavras: cargo estágio, escolas públicas particulares, recreação, quatro horas diárias, vale transporte formação acadêmica, prova objetiva, concorrer vagas candidato, livre docente, pedagogia, pibid, empregabilidade, graduação pós graduação, fisioterapeutas, ciências humanas suas, currículo lattes, rede pública privada, recrutamento seleção, período integral, mestrando, contraturno, produção audiovisual, ufpr, matrícula, original cópia, graduandos, edital nº, atividades lúdicas, educação continuada, rondônia unir, graduação prograd, prova discursiva, superior cursando experiência salá, escolas públicas obmep

Encontram-se abertas as inscrições para os cursos do Núcleo de Estudos da Terceira Idade NETI. São mais de dez cursos que acontecerão no segundo semestre de 2013, todos destinados a pessoas com mais de 50 anos.

cluster 18: Violência (5.440 palavras)

Exemplo de palavras: jazigo, porte ilegal, menino bernardo boldrini, conforme relato, bauru sp, praça mauá, foram enterrados, sitio, teve traumatismo craniano, chuva trovoadas isoladas, bandidos fugiram, polícia investiga, presídio estadual, abriu inquérito, quartel, ultrapassagem, reforçar policiamento, perfurações, furto roubo, presos fugiram, ônibus foi incendiado, tubarão, agentes penitenciários, polícia militar bpm.

Um menor de idade perdeu o controle do veículo e colidiu
com um poste na madrugada deste domingo (13) no
Lago Por do Sol em Iporá.

cluster 19: Órgãos governamentais (3.185 palavras)

Exemplo de palavras: patrimônio histórico, marketing esportivo, projetos voltados, transportes rodoviários, scania, firmar parcerias, secretaria extraordinária, apls, turismo vinicius lages, pedra fundamental, universidade tecnológica, bolsa pódio, aes, aeb, medicina crm, cvs, cbtu, secretários estaduais, selo combustível social, pgm, foram investidos milhões, pesquisa agropecuária embrapa, instituto akatu, policiamento comunitário, pecém, uern, ibp, firmou contrato, grupo odebrecht, abinee, petróleo gás natural biocombustíveis, justiça itinerante, coordenador geral, mt gov br, prefeito alfredo, concessionária responsável pela

Ministério da Cultura celebrou, nessa quinta-feira (30),
convênio para a formação de uma Rede Intermunicipal de Pontos
de Cultura em Araçatuba, interior de São Paulo.
Ao todo, serão 40 pontos conveniados na região, com 35 já atuando.

cluster 24: Problemas de codificação (2.873 palavras)

Exemplo de palavra: vã rios, satisfaã, emc, tâ xi, wtm latin america, emâ, verã, suas prã³prias, fusã, logã stica, caã ram, especã ficos, informaã es confidenciais, prã³ximos meses, nã oã, mandic, emissã, histãrias, polã ticas, concepã, concurso pãºblico, partã culas, famã lia, preã mã dio, nã's nã, amazã nia, combustã vel

TIM e Claro jã realizam prã©-cadastro para clientes interessados nos aparelhos, lanã§ados em setembro nos EUA.

Vivo e Oi ainda nãõ se manifestaram.

As operadoras TIM e Claro anunciaram nesta sexta-feira, 8/11, que vãõ comeã§ar a vender os novos iPhone 5S e 5C no prã³ximo dia 22 de novembro.

cluster 25 Tecnologia (3.057 palavras)

Exemplo de palavras: gameplay, megabytes, vga, ram gb, reconfigurar, patch, half life, utilitário, celular wipe cache dalvik, memorias, pc, polegar, tela touch, consoles, steam machine, playback, bootloader, reviews, call of duty advanced, abaixo estão, iwatch, media player, fabricante japonesa, imprima, hash, stylus, gadget, offline, você for seguir, multiplataforma, foco automático, lg watch, dissipador, swipe, idle, cartão microsd

Google lança projeto de software livre chamado Coder que pode transformar um dispositivo Raspberry Pi em um servidor web e proporcionar um ambiente para programadores iniciantes praticarem devolvimento de código na web.

Capítulo 5

Conclusão e considerações finais

A representação distribuída de palavras, fornecida pelo *skip-gram*, mostrou-se um método eficaz para representação das palavras em um domínio mais compacto e com alto valor semântico. O modelo utilizado para a detecção de frases permitiu a criação de frases sendo possível representar estas como uma “única palavra”, e através dos resultados observados, estes “pedaços de palavras” carregam uma informação muito forte.

Assim como já explicitado por [Mikolov et al. \(2013b\)](#), a representação distribuída é capaz de representar semântica e sintaticamente as palavras mesmo para a língua português-brasileiro, embora não tenham sido feito testes mais robustos para medir a acurácia das relações semânticas e sintáticas identificadas pelo método.

Quanto ao conteúdo do *Media Cloud Brasil*, o método mostrou-se sensível o suficiente para a detecção de *spam*, conteúdo em línguas estrangeiras, problemas de codificação e documentos onde o texto não foi efetivamente extraído. Com isso é possível fazer inferência quanto a temas de novos textos apenas calculando a proporção de cada cluster de palavras presente naquele documento. Foram identificados também diversos grupos de notícias, como esporte, entretenimento, política e internacional. Admitindo-se uma postura conservadora, podemos utilizar os documentos com maiores proporções

nos tópicos identificados para o treinamento de um classificador para fazer inferência nos documentos onde não é possível inferir o assunto através das proporções de cada um dos clusters citados.

O projeto *Media Cloud Brasil*, embora seja recente e apresente alguns problemas explicitados ao longo deste trabalho, já mostrou seu potencial ao mostrar que as relações entre palavras podem ser identificadas apenas a partir do conteúdo coletado pelo mesmo. É possível que com o passar do tempo, este projeto seja uma das principais fontes de dados disponível para possíveis consultas e análises de documentos.

5.1 Trabalhos Futuros

Podemos citar como trabalho futuro os seguintes pontos desenvolvidos durante esta pesquisa:

Utilização de um método de clusterização mais flexível

O modelo escolhido neste trabalho para clusterizar as palavras não é flexível. A polissemia existe quando trabalha-se com linguagem e este fato não pode ser ignorado. Ao optar por uma clusterização mais flexível, por exemplo os modelos de clusterização *fuzzy* uma palavra não pertencerá apenas a um único grupo, e sim a todos com pertinência diferente.

Forma melhor de validar os dados

Embora seja um projeto recente e promissor o conjunto de dados escolhido para a realização deste trabalho não possui dados marcados de nenhuma natureza que possam fornecer um ambiente de validação

Testes automatizados de semântica para o português-brasileiro

Assim como Mikolov aplicou testes automático para avaliação de expressões semânticas, pode-se aplicar a mesma lógica utilizada, isto é uma tabela com diversas relações entre as palavras para reproduzir e avaliar os resultados para a língua português-brasileira. Ainda que o método seja agnóstico quanto aos dados de entrada a representação do vocabulário - por exemplo flexões verbais - são diferentes, o que pode enfraquecer o método para determinada língua.

Utilizar grafos para exploração de comunidades

Uma alternativa aos métodos de clusterização tradicionais é a possibilidade da utilização de modelos de grafos para a identificação de comunidades entre palavras.

Referências Bibliográficas

- Ball, G. H. e Hall, D. J. (1965). Isodata, a novel method of data analysis and pattern classification. Technical report, DTIC Document.
- Bengio, Y., Ducharme, R., Vincent, P., e Janvin, C. (2003). A neural probabilistic language model. *The Journal of Machine Learning Research*, 3:1137–1155.
- Benkler, Y., Roberts, H., Faris, R., Solow-Niederman, A., e Etling, B. (2015). Social mobilization and the networked public sphere: Mapping the sopa-pipa debate. *Political Communication*, (ahead-of-print):1–31.
- Blei, D. M., Ng, A. Y., e Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.
- Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., e Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479.
- Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., e Harshman, R. A. (1990). Indexing by latent semantic analysis. *JAsIs*, 41(6):391–407.
- Härdle, W. K., Müller, M., Sperlich, S., e Werwatz, A. (2012). *Nonparametric and semiparametric models*. Springer Science & Business Media.
- Lloyd, S. P. (1982). Least squares quantization in pcm. *Information Theory, IEEE Transactions on*, 28(2):129–137.

- MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. Em *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pgs. 281–297. Oakland, CA, USA.
- Mikolov, T. (2007). *Language Modeling for Speech Recognition in Czech*. PhD thesis, Master’s thesis, Brno, FIT BUT.
- Mikolov, T., Chen, K., Corrado, G., e Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Mikolov, T., Le, Q. V., e Sutskever, I. (2013b). Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., e Dean, J. (2013c). Distributed representations of words and phrases and their compositionality. Em *Advances in neural information processing systems*, pgs. 3111–3119.
- Mnih, A. e Hinton, G. E. (2009). A scalable hierarchical distributed language model. Em *Advances in neural information processing systems*, pgs. 1081–1088.
- Morin, F. e Bengio, Y. (2005). Hierarchical probabilistic neural network language model. Em *Proceedings of the international workshop on artificial intelligence and statistics*, pgs. 246–252. Citeseer.
- Osiński, S., Stefanowski, J., e Weiss, D. (2004). Lingo: Search results clustering algorithm based on singular value decomposition. Em *Intelligent information processing and web mining*, pgs. 359–368. Springer.
- Řehůřek, R. e Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. Em *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pgs. 45–50, Valletta, Malta. ELRA. <http://is.muni.cz/publication/884893/en>.

- Rong, X. (2014). word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*.
- Steinhaus, H. (1956). Sur la division des corp materiels en parties. *Bull. Acad. Polon. Sci*, 1:801–804.
- Turian, J., Ratinov, L., e Bengio, Y. (2010). Word representations: a simple and general method for semi-supervised learning. Em *Proceedings of the 48th annual meeting of the association for computational linguistics*, pgs. 384–394. Association for Computational Linguistics.
- Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, pgs. 433–460.
- Wang, H., Gao, B., Bian, J., Tian, F., e Liu, T.-Y. (2015). Solving verbal comprehension questions in iq test by knowledge-powered word embedding. *arXiv preprint arXiv:1505.07909*.
- Wikipedia (2014). k-means clustering. [Online; accessed 02-June-2015].