SASI: sumarizador automático de documentos baseado no problema do subconjunto independente de vértices

Elvys Linhares Pontes, Juan-Manuel Torres-Moreno, Andréa Carneiro Linhares

SASI: sumarizador automático de documentos baseado no problema do subconjunto independente de vértices

Elvys Linhares Pontes
Universidade Federal do Ceará
Campus Sobral, Fortaleza, CE, Brasil
elvyslpontes@gmail.com

Andréa Carneiro Linhares
Universidade Federal do Ceará
Campus Sobral, Fortaleza, CE, Brasil
andrea.linhares@ufc.br

Juan-Manuel Torres-Moreno
University of Avignon
Avignon 84000, França
juan-manuel.torres@univ-avignon.fr

RESUMO

Este artigo discute um sistema sumarizador de documentos denominado SASI. Esse sistema apresenta uma abordagem inovadora na produção de resumos automáticos com base na determinação do subconjunto independente máximo de vértices, modelando o problema como um grafo de frases (vértices) e relações entre as mesmas (arestas). São descritos os conceitos e funcionamento do sumarizador proposto e uma série de testes comparando os resultados fornecidos pelo SASI com outros sistemas sumarizadores. Os resultados iniciais são promissores, avaliando-se o quesito de informatividade dos resumos diante de parâmetros de tempo e complexidade algorítmica.

PALAVRAS CHAVE. Subconjunto independente de vértices, Sumarização automática, Grafo de frases.

Área Principal: Teoria e Algoritmos em Grafos, Outras aplicações em PO.

ABSTRACT

This article discusses a summarizer system of documents named SASI. This system features an innovative approach to provide automatic summaries, based on the determination of the maximum independent subset of vertices, modeling the problem a graph of phrases (vertices) and the relationships between them (edges). The concepts and operation of the proposed summarizer and a series of tests comparing the results provided by SASI with others summarizer systems are described. Initial results are promising, evaluating questions of informativeness of the produced summaries on the parameters of time and algorithmic complexity.

KEYWORDS. Independent vertex subset, Automatic summarization, Phrases graph.

Main Area: Theory and Algorithms in Graphs, Other applications in OR.

1. Introdução

Resumir consiste em condensar as informações mais importantes de um ou vários documentos, a fim de produzir uma versão reduzida do seu conteúdo com as partes importantes do texto original e que seja gramaticalmente correto [Mani and Maybury(1999), Das and Martins(2007), Filippova(2010)]. Os títulos de revistas, as propagandas e as sinopses são alguns exemplos de resumos que utilizamos cotidianamente. De modo geral, as pessoas são excelentes "resumidores" em termos de qualidade do resumo produzido. Baseando-se em estudos do comportamento de resumidores profissionais, os pesquisadores tentaram imitar o processo cognitivo de criação de um resumo [Pollock and Zamora(1975), Mihalcea(2004), Wann et al.(2009)].

Atualmente, há uma grande necessidade de se obter resumos de documentos nas mais diversas áreas do conhecimento, pois isso facilita a busca por textos de referência bem como acelera o processo de compreensão do mesmo. O trabalho aqui descrito objetivou a criação de um sumarizador automático denotado por SASI (Sumarizador Automático baseado em Subconjunto Independente), utilizando a teoria e algoritmos de grafos como ferramenta básica conceitual. Nesse intuito, modelou-se o problema como um grafo e implementou-se um algoritmo a partir de ideias clássicas utilizadas na obtenção do Subconjunto Independente Máximo de vértices de um grafo. Técnicas estatísticas foram igualmente consideradas para cálculos de similaridade e para a avaliação da qualidade dos resumos produzidos.

A seção 2 introduz formalmente o problema do Subconjunto Independente Máximo (SIM), enquanto a seção 3 fará um breve estado da arte sobre sumarização automática de textos. A seção 4 introduz a notação matemática adotada e a modelagem do problema. Em 5, o sistema SASI é apresentado, onde sua fundamentação e funcionamento são detalhados. Os resultados são discutidos na seção 6 e as conclusões apresentadas na seção 7.

2. Problema do Subconjunto Independente Máximo

Considere um grafo $G = (V, E)$, onde V é o conjunto de vértices e E o conjunto de arestas não orientadas de G. Conceitualmente, vértices adjacentes ou vizinhos são aqueles ligados diretamente por uma aresta. Desse modo, um Subconjunto Independente de Vértices (denotado como SIV neste trabalho) é formado por um subconjunto dos vértices do grafo que não são adjacentes [Garey and Johnson(1990)], ou seja, vértices que não possuem aresta ligando-os.

O SIM é o subconjunto independente de vértices com maior cardinalidade no grafo. Ele é um problema NP-Difícil. Uma forma simples de tentar calcular o SIM é através do método guloso abaixo, descrito no trabalho de [Halldon and Radhakrishnan(1997)]. Esse problema é complementar ao problema da clique máxima [Garey and Johnson(1990)].

Algoritmo
Entrada Grafo G.
Saída Subconjunto Independente de Vértices do grafo G.
1 SIV $\leftarrow null$
2 **enquanto** $G \neq \emptyset$ **faça**
3 v é vértice de G tal que $d(v) = \min_{u \in V(G)} d(u)$
4 $G \leftarrow G - (\{v\} \cup N(v))$
5 SIV$\leftarrow$ SIV $\cup \{v\}$

Para um grafo G, o algoritmo deverá determinar seu subconjunto independente de vértices. O algoritmo calcula o grau dos vértices de G e ordena-os de modo crescente em função do grau. Conforme o algoritmo, o vértice v de menor grau de G é removido e adicionado ao conjunto SIV. Se algum vizinho de v, denotado por N(v), for viável, ele será removido de G. O processo se repete enquanto o conjunto G possuir vértices. Ao final do algoritmo, o subconjunto encontrado do grafo G será o melhor possível em termos de cardinalidade, mas como o algoritmo não é exato, não é possível assegurar que ele seja um SIM.

Em [Butenko(2003)], o SIM é abordado com mais detalhes, explorando questões acerca da complexidade algorítmica e de possíveis soluções, adaptadas às características do grafo. O SIM está presente em várias aplicações, como coloração de grafos, agendamento de tarefas, atribuição de canais de rádio, entre outras. Em [Yutao et al.(2009)], o SIM é utilizado para alocação de canal em sistemas de rádio cognitivos.

3. Sumarizador Automático de Documentos

Nos primeiros trabalhos acerca da sumarização automática de documentos, o trabalho [Luhn(1958)] descreve uma técnica simples e específica aos artigos científicos que utiliza a distribuição das frequências das palavras no documento para ponderação de frases. Ele apresenta algumas das vantagens que os resumos produzidos de modo automático têm com relação aos manuais: custo de produção bem reduzido, inexistência de problemas de subjetividade e de variabilidade observadas nos "resumidores" profissionais, dentre outros. A grande maioria dos sistemas resumidores de hoje continuam baseados nessa mesma ideia.

[Edmundson(1969)] deu continuidade aos trabalhos de Luhn, adicionando ao processo de produção de resumos considerações sobre posição das frases e presença de palavras provenientes da estrutura do documento (por exemplo, títulos, sub-títulos, etc.). As pesquisas desenvolvidas em [Pollock and Zamora(1975)] no *Chemical Abstracts Service* (CAS) concernentes à produção de sumários a partir de artigos científicos de Química permitiram validar a viabilidade das abordagens de extração automática de frases. Uma "limpeza" das frases - uso de operações de eliminação, foi introduzida pela primeira vez. No intuito de adequar os resumos aos padrões impostos pela CAS, uma normalização do vocabulário era efetuada. A normalização inclui a substituição de palavras/frases por suas abreviações e uma padronização das variantes ortográficas. Em seguida, os estudos sobre sumarização automática de textos foram divididos em dois grupos, extração e abstração de texto. Na primeira, há a identificação das partes mais relevantes em um ou mais textos, através de técnicas de recuperação de informação estatística. O trabalho de [Wu et al.(1992)] descreve vários métodos estatísticos utilizados no processo de sumarização de textos. A abstração analisa o texto original de uma forma linguística profunda, interpreta o texto semanticamente em uma representação formal, encontra novos conceitos mais concisos para descrevê-lo e, em seguida, gera um novo texto mais curto com o mesmo sentido do texto original [Hovy and Lin(1998)].

Os trabalhos voltados para a extração de frases se baseiam na seguinte metodologia de produção de sumários:

- Pré-processamento do texto;
- Identificação das frases em destaque no documento;
- Construção do sumário por concatenação das frases extraídas.

Outro campo de estudo uniu os conceitos de Processamento de Linguagem Natural (PLN) com os conceitos de grafos, como em [Mihalcea(2004)], que desenvolveu algoritmos de classificação baseados em grafos tais como *PageRank*. Esses algoritmos foram utilizados com sucesso nas redes sociais, na análise do número de citações ou no estudo da estrutura da Web. Eles permitem tomar decisões acerca da importância de um vértice, baseando-se na informação global advinda da análise recursiva do grafo completo, e não na análise local do vértice. No âmbito da sumarização automática, observa-se que o documento é representado por um grafo de unidades textuais (frases) conectadas entre elas através de relações oriundas de cálculos de similaridade. As frases são em seguida selecionadas segundo critérios de centralidade ou de prestígio no grafo, e em seguida reunidas a fim de produzir os extratos do texto.

Outra metodologia utilizada com grafos e PLN é descrita em [Filippova(2010)]. Esse artigo descreve uma forma de obter o resumo do texto através do menor caminho num grafo de palavras juntamente com processos de sintática. Em [Laureano-Cruces and Ramírez-Rodríguez(2012)],

os autores modelam o texto através de um grafo e calculam o peso entre as frases. A partir dessa modelagem, um resumo é produzido.

Além de gerar resumos, tem-se que analisar a qualidade deles. Um bom resumo deve ser significativamente menor do que o documento original, transmitir o máximo de informação possível do documento original e ser gramaticalmente correto [Almeida et al.(2014)]. Para isso, são analisadas três características do resumo: concisão, informatividade e gramaticalidade. A concisão representa que o resumo deve ser significativamente menor do que o texto original. Normalmente utiliza-se uma quantidade máxima de palavras ou de frases para gerar um resumo. A informatividade descreve a quantidade de informações importantes que o resumo possui em relação ao texto original. A gramaticalidade verifica a corretude gramatical do resumo [Boudin(2008)].

4. Modelagem Matemática

Neste trabalho, objetivou-se a concepção de um sistema de sumarização automática genérico. Esse sistema utilizou algoritmos oriundos da Teoria dos Grafos (através dos trabalhos descritos na seção 2 para calcular o SIM), a fim de determinar no texto suas frases mais importantes. Foi utilizado tratamento estatístico no intuito de construir um sistema o mais independente possível da linguagem. Os métodos propostos têm como base um pré-tratamento específico das palavras e uma função de ponderação das frases baseada na estrutura do grafo e na relação de vizinhança das mesmas (frases).

Foi utilizado o modelo de saco-de-palavras para representar o texto. Esse modelo utiliza uma matriz $S_{[m \times n]}$ construída a partir do documento, em que m é o número de frases e n é a quantidade de palavras distintas do texto. A célula s_{ij} da matriz representa a frequência da palavra j na frase i (FP_{ij}).

$$S = \begin{pmatrix} s_{11} & s_{12} & \dots & s_{1n} \\ s_{21} & s_{22} & \dots & s_{2n} \\ \vdots & \vdots & & \vdots \\ s_{m1} & s_{m2} & \dots & s_{mn} \end{pmatrix}, s_{ij} = \begin{cases} FP_{ij}, & se \exists\, palavra\ j\ na\ frase\ i \\ 0, & caso\ contrário \end{cases} \tag{1}$$

Na modelagem do problema, W é conjunto das palavras contidas nos conjuntos P e Q. P e Q podem representar frases ou conjunto de frases, conforme o cálculo da divergência que se deseja efetuar: entre duas frases; entre uma frase e um documento; entre documentos. A divergência Kullback-Leibler (KL) calcula a informação perdida ao aproximar uma distribuição P a uma distribuição Q (equação 2) [Søgaard(2013)]. Esse cálculo permite calcular a divergência entre dois conjuntos de palavras distintos, entretanto essa divergência é assimétrica ($DKL(P||Q) \neq DKL(Q||P)$). Essa assimetria não garante uma análise correta entre duas distribuições, pois teriam valores diferentes dependendo da análise realizada. Entretanto, a divergência Jensen-Shannon (JS) é uma versão simétrica e suavizada da divergência KL fornecendo uma forma mais estável para mensurar a diferença entre duas distribuições. Portanto, utilizou-se a divergência Jensen-Shannon e variações da mesma, a fim de mensurar a divergência (inverso da similaridade) entre duas distribuições P e Q (equação 3)[Torres-Moreno et al.(2010), Saggion et al.(2010)].

$$DKL(P||Q) = \frac{1}{2} \sum_{w \in W} \left(P_w \log \frac{P_w}{Q_w} \right) \tag{2}$$

$$DJS(P||Q) = \frac{1}{2} \sum_{w \in W} \left(P_w \log \frac{2 * P_w}{P_w + Q_w} + Q_w \log \frac{2 * Q_w}{P_w + Q_w} \right) \tag{3}$$

Supondo-se P e Q duas frases do documento e R seja o conjunto formado pelas palavras das frases P e Q, P_w representa a distribuição do conjunto de palavras da frase P relacionadas a todas as palavras do conjunto R. Respectivamente, Q_w será a distribuição do conjunto de palavras

da frase Q relacionadas às palavras do conjunto R. A divergência JS calcula a divergência entre duas distribuições, onde seu valor varia de $[0, \infty+)$. Os valores da divergência JS são mais próximos a zero quando as distribuições são semelhantes e mais próximo do valor 1 quando elas diferem. Desse modo, para as frases P e Q, quanto maior o valor de $DJS(P||Q)$ mais uma frase diverge da outra.

A divergência JS será utilizada para calcular a divergência entre duas frases, analisando a frequência de cada palavra presente nelas. No caso de haver uma palavra em uma frase que seja inexistente na outra, será utilizado um *smooth* (ponderação diferenciada) para evitar valores nulos e ter uma distribuição mais suave [Hiemstra(2009)]. Caso uma palavra w não esteja presente na frase Q, então o *smooth* é dado abaixo:

$$Q_w = \left(\frac{P_w + \gamma}{N + \gamma * \beta} \right) \tag{4}$$

onde $\beta = 1.5 * Vocabulário$, $Vocabulário$ é a quantidade de palavras distintas de R, γ é a variável que controla a relevância da ausência de uma palavra na frase e N o número de palavras presente em R.

Os cálculos descritos em (3) e (4) são utilizados no processo de pré-seleção de frases no documento, bem como na criação de arestas do grafo resultante, as quais representarão a existência de similaridade entre duas frases.

5. Sistema SASI

A etapa de pré-processamento é efetivada pelo SASI através do processo *stemming* (segmentação), onde são removidas todas as partes assumidas desnecessárias (conjunções, artigos, pontuação). Em seguida, realiza-se o processo de lematização para analisar os radicais das palavras e evitar diferenças entre palavras derivadas. Por fim, verifica-se cada palavra do texto, analisando-se sua frequência no texto e em cada frase através da matriz S (matriz de saco de palavras).

Inicialmente P e Q referenciam, respectivamente, o texto completo e cada frase isolada do mesmo. Assim, P_w será o conjunto de distribuições de palavras do texto P e Q_w será o conjunto de distribuições de palavras da frase Q. Dessa forma, $DJS(P||Q)$ calculará a divergência entre cada frase e o texto, verificando se uma frase é importante para o documento por meio da estatística das palavras existentes:

- Se a divergência JS da frase for pequena (próximo a zero) em relação ao texto completo, a frase será relevante;
- Caso contrário, uma nova análise de similaridade é realizada com relação ao título do documento. Se a DJS entre a frase e o título for pequena, a frase será considerada relevante, senão, ela será descartada da análise e assim do grafo.

Após selecionarmos as frases relevantes para o documento, verificamos a divergência entre as mesmas. Nessa etapa, P e Q correspondem a cada par de frases selecionadas. Em seguida, identifica-se o conjunto de palavras pertencentes a cada frase e calcula-se a $DJS(P||Q)$.

O grafo do problema é construído a partir do conjunto de frases relevantes (vértices) e o valor da divergência JS entre duas frases será utilizado para a criação e ponderação de arestas. Para a inserção de arestas, analisa-se todas as combinações possíveis de duas frases. Caso a divergência entre elas for "pequena" (parâmetro do algoritmo), uma aresta interligando-as será inserida. Ao final desse processo, teremos um grafo onde as ligações (arestas) representam a similaridade entre as frases.

Como o processo de sumarização objetiva a produção de um resumo de tamanho menor que o conteúdo original e com as principais informações do texto, seria interessante escolher somente as frases que trazem informação adicional ao resumo. A escolha de vértices adjacentes implica na seleção de frases com conteúdo redundante, o que não é interessante para o resumo. Por

isso, a determinação do subconjunto independente de vértices do grafo fornecerá a solução para o problema. Assim, a ideia é que o sistema selecione as frases com conteúdo distinto entre si, descartando aquelas com conteúdo repetitivo ou que trazem pouco informação adicional ao resumo. No término do algoritmo, temos o subconjunto de vértices representando as frases do texto resumido.

O algoritmo abaixo descreve o funcionamento do sistema SASI. É importante destacar que neste trabalho tenciona-se a obtenção de um resumo de um documento, sem preocupar-se com um tamanho máximo ou mínimo de frases selecionadas.

Algoritmo
Entrada Texto.
Saída Resumo do texto.
1 Pré-processamento do $texto$
2 $Resumo \leftarrow \emptyset$
3 **para** Cada frase $f \in texto$ **faça**
4 **se** $DJS(f||texto) < DJSTexto$ **então**
5 $Resumo \leftarrow Resumo \cup f$
6 **senão se** $DJS(f||titulo) < DJSTitulo$ **então**
7 $Resumo \leftarrow Resumo \cup f$
8 Cria grafo $G[Resumo]$
9 **para** Cada par de vértices P e Q (frases) $\in G$ **faça**
10 **se** $DJS(P||Q) < DJSFrase$ **então**
11 $A(P,Q) \leftarrow A(Q,P) \leftarrow 1$
12 **senão**
13 $A(P,Q) \leftarrow A(Q,P) \leftarrow 0$
14 Calcula o $SIM(G)$

6. Resultados

O desempenho do sistema descrito na seção 5 foi analisado a partir de diversos valores dos parâmetros associados ao cálculo da divergência JS. Os testes foram realizados num computador com processador i5 2.6 GHz e 4 GB de memória RAM no sistema operacional Debian de 64 bits. Os algoritmos foram implementados utilizando a linguagem de programação Perl. As similaridades foram analisadas entre: uma frase e outra ($DJSFrase$); uma frase e o título ($DJSTitulo$); e uma frase e o documento ($DJSTexto$). A tabela 1 descreve o nível de similaridade relacionado ao valor da divergência JS e os valores selecionados nos experimentos aqui descritos. Analisou-se o γ variando entre $0,01$ e $0,20$ e, a partir dos resultados, utilizou-se $\gamma = 0,1$.

Parâmetros	Nível de similaridade			
	Fraca	Média	Forte	**Selecionado**
$DJSFrase$	$JS > 0,75$	$0,25 < JS < 0,75$	$JS < 0,25$	**0,32**
$DJSTitulo$	$JS > 0,8$	$0,4 < JS < 0,8$	$JS < 0,4$	**0,6**
$DJSTexto$	$JS > 0,95$	$0,65 < JS < 0,95$	$JS < 0,65$	**0,9**

Tabela 1: Nível de similaridade relacionado à divergência JS

Neste trabalho, o corpus, conjunto de documentos que serve como base de análise, é composto por textos de jornais e revistas científicas em inglês, francês e espanhol. Além dessas características, os resumos dos textos selecionados estão disponíveis na literatura para outros sistemas sumarizadores [Torres-Moreno et al.(2002), Fernandez et al.(2007)]. O corpus utilizado é composto por 13 textos de diferentes tamanhos e assuntos. A tabela 2 mostra 4 textos do corpus detalhando a quantidade de frases dos resumos para textos de diferentes tamanhos e valores da $DJSFrase$. Pode-se observar, a partir dos dados dessa tabela, que quanto maior o valor da DJS entre frases menor será o resumo (número de frases) fornecido pelo algoritmo. Portanto, esse valor

dependerá do tamanho do resumo que se deseja obter. Deve-se considerar que quanto menor o resumo mais informações serão omitidas e a informatividade poderá ser reduzida.

Textos	Quantidade de Frases					
	Texto Original	Resumos				
		Valor da $DJSFrase$				
		0,30	**0,32**	0,33	0,35	0,40
Mars	12	6	**4**	4	4	2
Puces	31	14	**9**	8	5	3
Lewinsky	32	13	**7**	7	5	3
Quebec	45	18	**15**	14	12	6

Tabela 2: Tamanho dos resumos obtidos para um conjunto de valores da DJS.

Foram usados na comparação dos nossos resultados os valores fornecidos pelos sistemas sumarizadores Cortex, Enertex e REG. O REG modela o documento como um grafo e atribui arestas ponderadas ao grafo para gerar o resumo do texto [Torres-Moreno and Ramírez(2010)]. O Cortex é um sistema baseado em métodos numéricos, que é independente do tema e do comprimento do texto [Torres-Moreno et al.(2002), Boudin and Torres-Moreno(2007)]. O Enertex é um sistema que utiliza a abordagem de redes neurais, inspirado na física estatística de sistemas magnéticos para sumarizar textos [Fernandez et al.(2007), Fernandez et al.(2008)]. Dessa forma, comparou-se a informatividade dos resumos fornecidos pelo SASI, Cortex, Enertex e REG em relação aos sumários dos resumidores profissionais (tabela 3).

A figura 1 descreve o desempenho obtido em segundos pelo sistema SASI na produção de resumos a partir dos textos da tabela 2 com a quantidade de palavras variando entre 11 e 214. Os sistemas obtiveram os seguintes tempo de execução para o corpus analisado: Cortex, 58,27 segundos; Enertex, 56,80 segundos; REG, 55,70 segundos; e SASI, 38,78 segundos.

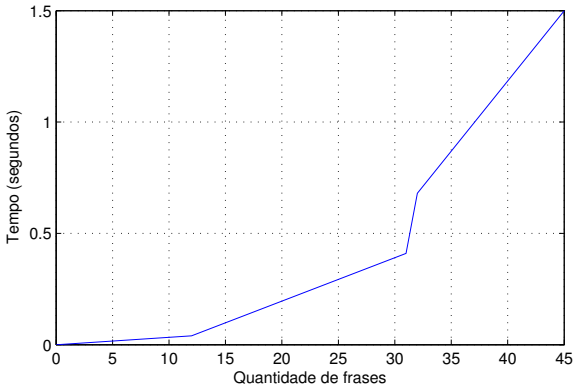

Figura 1: Desempenho do sistema SASI.

Analisou-se, ainda, a qualidade dos resumos através da informatividade, que foi calculada através da taxa de acerto das frases fornecidas pelos sistemas sumarizadores em relação aos resumos dos profissionais descritos na tabela 3.

O SASI obteve o melhor resultado no texto Quebec e um desempenho semelhante aos outros sistemas no texto Puces. Nos textos Lewinsky e Mars, o SASI teve uma taxa de acerto inferior aos demais sistemas, pois os SIV calculados não corresponderem aos SIM dos grafos analisados e a ponderação das frases não abordar outros elementos importantes nos textos. Nos demais textos do Corpus, os resumos tiveram a informatividade semelhante aos resultados obtidos nos resumos Puces. Assim, apesar do SASI ter obtido um desempenho inferior em 69% dos textos analisados,

Textos	Sistemas			
	SASI	Cortex	Enertex	REG
Mars	67%	100%	100%	100%
Puces	63%	75%	63%	63%
Lewinsky	43%	57%	57%	57%
Quebec	73%	55%	46%	55%

Tabela 3: Análise da precisão dos resumos gerados.

a informatividade dos documentos foi preservada, o que é extremamente importante na construção de um resumo de forma automática.

O SASI é uma ferramenta que tem por base uma heurística simples e que se apoia em cálculos estatísticos menos complexos do que outros sistemas utilizados no domínio do PLN, no âmbito da sumarização automática de documentos. A integração de novas regras sintáticas e semânticas para pré-tratamento dos textos proverá melhorias no desempenho do SASI. Além disso, com base nos testes realizados, o desenvolvimento de um método mais refinado para calculo do SIV que assegure resultados mais próximos de um SIM, impactará de maneira positiva na qualidade dos sumários produzidos, visto que serão escolhidas um numero maior de "frases importantes" do texto e com conteúdo distinto, o que evitaria que frases relevantes sejam descartadas no processo.

7. Conclusão

Analisando o trabalho desenvolvido e as abordagens existentes na literatura, pode-se concluir que a utilização do SIM concomitante ao uso da divergência JS para construção de um grafo de frases e relações entre as mesmas é inovadora.

O SASI conseguiu resultados satisfatórios comparado aos outros sistemas na literatura. Os resultados desse sistema foram semelhantes aos demais fornecendo sempre uma boa taxa de informatividade dos resumos. Em relação ao tempo de execução, o SASI conseguiu o menor tempo. Portanto, o sistema SASI é um sumarizador viável, pois forneceu maior parte das principais informações dos resumos em tempo hábil.

A divergência JS é simétrica, e esse recurso pode ter consequências negativas sobre a tarefa de resumo automático: a frase é sempre menor do que a original. Uma nova versão da divergência KL (não simétrica e ponderada pelo comprimento da frase) poderia ser usada para melhorar o resultado. No futuro, testaremos a divergência KL modificada como descrita em [Torres-Moreno(2014)].

Outras perspectivas futuras são melhorar a ponderação entre as frases analisando aspectos semânticos do texto, como também, verificar palavras sinônimas e fazer a abstração das principais frases do texto. Além disso, testar o SASI com um corpus em outros idiomas (Português, Espanhol, Inglês, etc.) para o qual estejam disponíveis resumos fornecidos por resumidores profissionais.

8. Agradecimentos

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUNCAP).

Referências

Almeida, M. B., Almeida, M. S. C., Martins, A. F. T., Figueira, H., Mendes, P., and Pinto, C. (2014). A new multi-document summarization corpus for european portuguese. *Language Resources and Evaluation Conference (LREC'14)*.

Boudin, F. (2008). *Exploration d'approches statistiques pour le résumé automatique de texte*. PhD thesis, Université D'Avignon et des pays de Vaucluse.

Boudin, F. and Torres-Moreno, J.-M. (2007). Neo-cortex: A performant user-oriented multi-document summarization system. In *Computational Linguistics and Intelligent Text Processing*, pages 551–562. Springer Berlin/Heidelberg.

Butenko, S. (2003). *Maximum independent set and related problems, with applications*. PhD thesis, University of Florida Gainesville, FL, USA.

Das, D. and Martins, A. F. (2007). A survey on automatic text summarization. *Literature Survey for the Language and Statistics II course at CMU*.

Edmundson, H. P. (1969). New methods in automatic extracting. *Journal of the ACM (JACM)*, pages 264–285.

Fernandez, S., SanJuan, E., and Torres-Moreno, J.-M. (2007). Textual energy of associative memories: Performant applications of enertex algorithm in text summarization and topic segmentation. *MICAI 2007: Advances in Artificial Intelligence*, pages 861–871.

Fernandez, S., SanJuan, E., and Torres-Moreno, J. M. (2008). Enertex: un système basé sur l'énergie textuelle. *Traitement Automatique des Langues Naturelles*, pages 99–108.

Filippova, K. (2010). Multi-sentence compression: finding shortest paths in word graphs. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 322–330. Association for Computational Linguistics.

Garey, M. R. and Johnson, D. S. (1990). *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA.

Halldon, M. M. and Radhakrishnan, J. (1997). Greed is good: Approximating independent sets in sparse and bounded-degree graphs. *Springer New York*, pages 145–163.

Hiemstra, D. (2009). Probability smoothing. In *Encyclopedia of Database Systems*, pages 2169–2170. Springer.

Hovy, E. and Lin, C.-Y. (1998). Automated text summarization and the summarist system. In *Proceedings of a Workshop on Held at Baltimore, Maryland: October 13-15, 1998*, pages 197–214. Association for Computational Linguistics.

Laureano-Cruces, A. and Ramírez-Rodríguez, J. (2012). A graph-based summarization system at qa@inex track 2011. In *Focused Retrieval of Content and Structure*, volume 7424 of *Lec. Notes in Computer Science*, pages 227–231. Springer Berlin Heidelberg.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, page 159.

Mani, I. and Maybury, M. T. (1999). *Advances in Automatic Text Summarization*. The MIT Press.

Mihalcea, R. (2004). Graph-based ranking algorithms for sentence extraction, applied to text summarization. *ACL 2004 on Interactive poster and demonstration sessions*, pages 181–184.

Pollock, J. J. and Zamora, A. (1975). Automatic abstracting research at chemical abstracts service. *J. of Chemical Information and C. Sciences*, pages 226–232.

Saggion, H., Torres-Moreno, J.-M., Cunha, I. d., and SanJuan, E. (2010). Multilingual summarization evaluation without human models. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 1059–1067. Association for Computational Linguistics.

Søgaard, A. (2013). *Semi-Supervised Learning and Domain Adaptation in Natural Language Processing*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Torres-Moreno, J.-M. (2014). *Automatic Text Summarization*. ISBN: 978-1-84821-668-6, Wiley-ISTE, London.

Torres-Moreno, J.-M. and Ramírez, J. (2010). Reg : un algorithme glouton appliqué au résumé automatique de texte. In *JADT*. JADT.

Torres-Moreno, J.-M., Saggion, H., da Cunha, I., Velázquez-Morales, P., and SanJuan, E. (2010). Evaluation automatique de résumés avec et sans référence. *Traitement Automatique des Langues Naturelles*.

Torres-Moreno, J.-M., Velázquez-Morales, P., and Meunier, J.-G. (2002). Condensés de textes par des méthodes numériques. In *JADT*, volume 2, pages 723–734. JADT.

Wann, S., Dras, M., Dale, R., and Paris, C. (2009). Improving grammaticality in statistical sentence generation: introducing a dependency spanning tree algorithm with an argument satisfaction model. In *Proceedings of the 12th EACL*, EACL '09, pages 852–860.

Wu, Z. B., Hsu, L. S., and Tan, C. L. (1992). A survey on statistical approaches to natural language processing.

Yutao, L., Mengxiong, J., Xuezhi, T., and Lu, F. (2009). Maximal independent set based channel allocation algorithm in cognitive radios. *Information, Computing and Telecommunication, 2009. YC-ICT'09. IEEE Youth Conference on*, pages 78 – 81.