Update Summarization

por

Vítor Costa

Tese de Mestrado em Modelação, Análise de Dados e Sistemas de Apoio à Decisão

Orientador

Professor Doutor Pavel Brazdil

2014

Breve Nota Bibliográfica

Vítor Manuel da Costa nasceu em 1976, em Vizela, terra onde morou até 2011, altura em que se mudou para Gondomar.

Em 1994 concluiu o ensino profissional, com o diploma de Técnico de Informática Fundamental.

Entre em 1999 e 2004 frequentou a Universidade do Minho, tendo-se licenciado em Informática de Gestão

Em 1994, juntamente com três sócios, fundou a *Característica Imagem & Comunicação, Lda.*, empresa onde se manteve durante dois anos a exercer diversas atividades relacionadas com a formação na área da informática.

É sócio fundador da empresa *Softideia – Informação Automática, Lda.*, criada em 1996, em Felgueiras, onde trabalha atualmente.

Agradecimentos

Começo por agradecer ao meu orientador, o Professor Doutor Pavel Brazdil. O seu conhecimento, apoio e orientação foram essenciais para conseguir desenvolver esta Tese de Mestrado. Acreditou sempre em mim e soube encaminhar-me para a direção correta. Espero que no futuro possamos voltar a colaborar.

Agradeço também a todos os professores do Mestrado em Modelação, Análise de Dados e Sistemas de Apoio à Decisão por todo o conhecimento que me transmitiram e pela disponibilidade que sempre tiveram para esclarecer as minhas dúvidas.

Deixo também um agradecimento ao Mohammadreza por ter disponibilizado os textos das conferências e por ter tido disponibilidade para esclarecer algumas das minhas dúvidas.

Os meus colegas de trabalho, na Softideia, em especial o António Macedo, foram inexcedíveis e tudo fizeram para compensar as minhas ausências, sempre que tive de me concentrar mais neste trabalho.

Quero também agradecer à minha família pelo apoio e por nunca deixarem de me encorajar.

Às três mulheres da minha vida

Ana, Susana e Ana

Resumo

Vivemos numa era em que há muito mais informação do que aquela que conseguimos assimilar e processar. Esta sobrecarga é bastante evidente no meio académico e científico onde existe uma grande e contínua produção de artigos. Perante este cenário impõe-se arranjar processos que facilitem o acesso e a leitura das informações. Um dos métodos mais utilizados é a sumarização de textos que consiste na produção de um pequeno texto que contém a informação relevante dos textos sumarizados. Quando há um conhecimento prévio de informações sobre um tópico, é necessário utilizar o *update summarization* que produz um sumário que apresenta apenas as informações relevantes e que ainda não surgiram nos textos anteriores, já conhecidos. Há diversas abordagens e métodos que podem ser utilizados tanto para a sumarização como para o *update summarization*. Neste trabalho apresentamos as várias etapas realizadas para implementar e avaliar em diversos cenários, sistemas que permitam fazer sumarização e *update summarization*. Os sistemas serão avaliados usando textos da *Document Understanding Conference 2007*, da *Textual Analysis Conference 2008* e da *Textual Analysis Conference 2009*.

Abstract

We are living in an era in which there is much more information than that we are able to assimilate and process. This overload is quite evident inside the academic and scientific community where there is a large and continuous production of articles. Before this scenario it is imperative to get processes that facilitate the access and the reading of the information. One of the most used methods is to summarize the texts that consists in producing a small text which includes the relevant information of the original texts. When there is a prior knowledge about a topic, it is necessary to use the update summarization that by itself produces a summary that only shows the relevant information that has not emerged in the previous texts that were already known. There are several approaches and methods that can be used for both summarization and update summarization. This work presents the several steps performed to implement and to evaluate in different scenarios, systems that allow us to summarize and to use the update summarization. The systems will be evaluated by using texts of *Document Understanding Conference 2007, the Textual Analysis Conference 2008 and Textual Analysis Conference 2009.*

Índice

Índice de Figuras

Índice de Gráficos

Índice de Tabelas

Siglas

DUC	Document Understanding Conference
IDF	Inverse Document Frequency
ISF	Inverse Sentence Frequency
MMR	Maximal Marginal Relevance
NLP	Natural Language Processing
PNR^2	Positive Reinforcement Negative Reinforcement
RNA	Redes Neuronais Artificiais
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
SCU	Summary Content Unit
TAC	Textual Analysis Conference
TF	Term Frequency
TREC	Text REtrieval Conference
SVM	Support Vector Machines
VSM	Vector Space Model

1. Introdução

1.1. Motivação

Vivemos numa época onde há um excesso de informação. Com a proliferação da internet a sobrecarga é de tal ordem que é de todo impensável conseguir assimilar toda a informação disponível.

Mesmo que alguém decida centrar-se apenas num determinado assunto, a não ser que seja realmente um caso muito específico, ainda assim dificilmente conseguirá obter, ler e assimilar todas as informações.

Pode dar-se ainda o caso de alguém gastar imenso tempo a ler muita informação, mas não estar a ler as melhores fontes, ou seja, estar a investir tempo e não tirar o devido benefício.

E numa sociedade que depende cada vez mais da informação, é essencial estar constantemente atualizado com as novidades que surgem nas várias áreas de interesse, sob pena de graves consequências.

É neste contexto que a sumarização de textos assume um papel de especial relevo, permitindo que os leitores acedam à informação relevante, de uma forma resumida.

A sumarização deve reduzir quer o tempo de leitura quer o tempo de avaliação dos aspetos relevantes dos documentos; em simultâneo, deve fornecer informação suficiente para que o leitor possa tomar as mesmas decisões que tomaria se tivesse lidos os documentos completos (Hobson (2007)).

Mas mesmo os sumários podem não ser suficientes, principalmente quando são relativos ao mesmo tópico, já que os diversos sumários, mesmo trazendo algumas novidades podem ser muito semelhantes, obrigando o leitor a reler a mesma informação vezes sem conta.

O *update summarization*[1] tenta solucionar este problema ao gerar sumários que resumem a informação relevante e que, em simultâneo acrescenta, traz alguma novidade em relação ao que o leitor já tinha lido anteriormente.

[1] Neste relatório optamos por utilizar a designação inglesa, já que não encontramos uma tradução que considerássemos adequada, sendo que a melhor tradução talvez fosse *Sumarização de Atualizações*.

A sumarização aplicada a notícias tem tido muita investigação nos últimos tempos (a generalidade dos trabalhos recentes usam os dados das conferências DUC e TAC). Contudo, os primeiros trabalhos sobre sumarização automática de textos incidiam sobre artigos científicos (Luhn (1958), Baxendale (1958) e Edmundson (1969)).

A importância do *update summarization* é evidente, bastando ter em consideração quer o número de artigos científicos publicados até agora, quer o número de publicações periódicas que continuam todos os dias a ser publicadas.

Jinha (2010) estimou que, desde 1665, altura em que surgiu a primeira publicação científica (*Philosophical Transactions of the Royal Society*), até 2009 foram publicados 50 milhões de artigos.

Larsen e Ins (2010) previam que em 2010 estivessem ativas cerca de 24.000 publicações científicas *sérias*, das áreas das Ciências Naturais e Sociais, Artes e Humanidades. Os autores entendem que uma publicação é considerada *séria* se assegura a revisão dos textos por pares.

Ainda para se ter uma ideia da quantidade de informação sobre ciência disponível na internet, o Scirus[2], um motor de pesquisa específico para a ciência da Elsevier, assegura que cobre mais de 545 milhões de páginas relacionadas com ciência, isto para além de milhões de artigos e documentos armazenados em bases de dados de referência (Scirus (2014)).

Por sua vez o Web of Knowledge (2014), que se afirma como o maior fornecedor de conteúdos científicos, garante ter cerca de 54 milhões de registos, de diversas publicações, em mais de 50 áreas diferentes.

Mas a dificuldade não surge apenas na pesquisa de nova informação. Os próprios autores/investigadores sentem dificuldade em acompanhar todas as citações que são feitas aos seus artigos, verificar em que contexto são citados e, acima de tudo, perceber o que é que os novos artigos trazem de novidade ao estado da arte. Consultando a lista dos oito autores mais citados no Microsoft Academics (dados recolhidos em Janeiro de 2014) verifica-se que o número médio de artigos publicados por estes autores é superior a 662 artigos, enquanto o número médio de citações é superior a 40.000. Um autor que tenha sido citado tantas vezes não

[2] http://www.scirus.com

conseguirá certamente acompanhar todos os artigos que o citam, perdendo rapidamente rasto de tudo o que de novo foi acrescentado aos seus artigos.

Todos estes números servem para sublinhar a necessidade de haver processos que facilitem o acesso às informações dos artigos científicos, por exemplo através da sumarização e do *update summarization*.

1.2. Objetivos

Com esta tese pretendemos estudar e explorar a sumarização automática de textos e o *update summarization*, através da implementação e avaliação de sistemas baseados em aprendizagem supervisionada e não supervisionada.

Pretendemos implementar diferentes sistemas que permitam criar sumários de um ou vários documentos. A ideia essencial é que os sistemas consigam atribuir uma ordenação às diversas frases do(s) documento(s), de acordo com a sua relevância e depois extraiam as mais bem pontuadas, garantindo ao mesmo tempo que não haja redundância ou que esta seja mínima.

Os sumários produzidos poderão ser genéricos ou baseados em tópicos ou questões. Desta forma poderão ser elaborados resumos diferentes conforme as necessidades/perguntas específicas do utilizador. Ou seja, o sistema terá de ser capaz, na ordenação das frases, de valorizar aquelas que estão mais relacionadas com o tópico em análise ou a questão do utilizador, caso exista.

De igual forma, será investigada e implementada a funcionalidade de *update summarization*. Essencialmente, perante dois conjuntos de documentos (já conhecidos e novos), pretendemos resumir as novidades relevantes trazidas pelos novos documentos.

Os algoritmos, baseados na literatura, serão implementados por nós no âmbito da tese. Pretendemos estudar a variação dos diversos parâmetros para verificar o impacto que essas alterações têm nos resultados finais.

Os sistemas serão testados com documentos oriundos da DUC 2007, da TAC 2008 e TAC 2009, para ser possível comparar os resultados com outras implementações.

Quanto à avaliação, será utilizada a avaliação automática, recorrendo à ferramenta ROUGE.

Ao longo desta tese esperamos conseguir responder a algumas questões relevantes:

- Que tipo de abordagem (supervisionada ou não supervisionada) consegue melhores resultados na sumarização? E no *update summarization*?
- É vantajosa a utilização de modelos múltiplos preditivos?
- O desempenho de modelos múltiplos homogéneos é diferente do desempenho de modelos múltiplos heterogéneos?
- Os sistemas mantêm um desempenho semelhante nos três conjuntos de documentos ou há diferenças significativas?
- Sistemas propostos para a sumarização podem ser usados em *update summarization* com alterações mínimas ou têm de ser completamente repensados?

1.3. Estrutura do relatório

Neste primeiro capítulo apresentamos a nossa motivação para o tema e definimos os objetivos do trabalho.

No capítulo 2 fazemos uma breve descrição da área de sumarização e de *update summarization* e procedemos a um levantamento de alguns trabalhos relacionados com a nossa tese.

No capítulo 3 apresentamos o nosso projeto de sumarização e *update summarization* e referimos as abordagens exploradas.

No capítulo 4 fazemos uma análise da configuração dos parâmetros, e apresentamos e discutimos os resultados obtidos com os sistemas criados.

Por fim, no capítulo 5 são tiradas algumas conclusões e são apresentados alguns tópicos interessantes para dar continuidade, no futuro, ao trabalho desenvolvido.

Em anexo disponibilizamos as tabelas com os resultados das experiências efetuadas, apresentamos alguns sumários extraídos com os sistemas implementados e disponibilizamos algum código implementado em R.

2. Contextualização e Trabalho Relacionado

2.1. Introdução às Áreas de Sumarização e *Update Summarization*

O objetivo da sumarização é criar uma descrição concisa de um ou vários documentos que tenha mais conteúdo que um simples título, mas que seja suficientemente pequena para poder ser assimilada muito rapidamente, preservando as informações mais importantes (Kupiec *et al.* (1995) e Das e Martins (2007)).

Segundo Li *et al.* (2013), um bom sumário dever ter o principal conteúdo dos textos sumarizados, deve ser relevante, contendo as informações que o leitor necessita, e não deve ser redundante; se houver conhecimento daquilo que o leitor já leu, o sumário deve conter apenas as informações que sejam novidade.

A *sumarização feita por humanos* é atualmente a que produz sumários de melhor qualidade e mais fiáveis. Em contrapartida, a sumarização feita por humanos pode ser muito demorada e custosa Hobson (2007). Tendo em consideração a enorme quantidade de informação produzida atualmente, torna-se impossível obter todos os sumários necessários recorrendo apenas pessoas.

Para colmatar este problema, a *sumarização automática*[1], gerada por máquinas (computadores), utiliza diversas técnicas e algoritmos para extrair, de forma condensada, as partes mais relevantes de um ou vários textos. A sumarização automática é mais rápida e mais barata, contudo, ainda não atingiu a qualidade dos sumários produzidos por humanos (Hobson (2007)).

A sumarização pode ser apenas de *um documento* (*single document summarization*) ou podem-se sumarizar diversos documentos em simultâneo, estando-se então perante *sumarização multidocumento*. Este tipo de sumarização é mais exigente, pois além de ser necessário descobrir as informações importantes de todos os documentos, é necessário garantir que o sumário reflete o que é expresso nos vários documentos, e não deve ser incoerente nem redundante (Das e Martins (2007)).

[1] Neste relatório, salvo indicação do contrário, sempre que se utilizar o termo *sumarização* está-se a referir *sumarização automática de textos*.

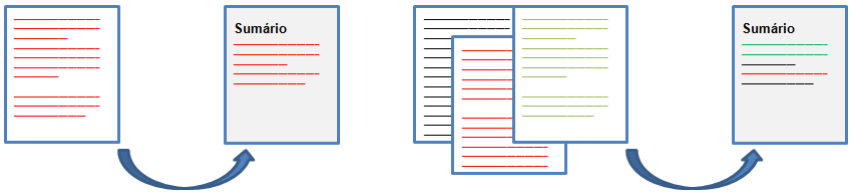


Figura 2.1 – Sumarização de um documento vs. Sumarização multidocumento

Uma distinção importante é entre *sumarização extrativa* e *sumarização abstrativa* (Hahn e Mani (2000)).

A questão fundamental a resolver na sumarização extrativa é determinar qual o conteúdo a extrair, isto é, escolher quais as frases dos documentos originais que deverão ser incluídas no sumário (Nenkova e Vanderwende (2005)). Basicamente a sumarização extrativa consiste em atribuir uma pontuação às frases (evidenciando a sua importância) e depois selecionar o conjunto das frases mais pontuadas (Kupiec *et al.* (1995)). O que distingue os vários trabalhos e abordagens apresentadas na literatura são as técnicas e algoritmos utilizados na atribuição da pontuação às frases (para realçar o que é importante) e a seleção do subconjunto que irá formar o sumário (garantindo a relevância ao mesmo tempo e evitando a redundância).

Na sumarização abstrativa há uma utilização mais profunda de técnicas de processamento de linguagem natural. Neste tipo de sumarização tenta-se compreender e interpretar os conceitos dos documentos e depois expressam-se esses conceitos de uma forma concisa e clara (Gupta e Lehal (2010)), podendo conter frases não presentes na origem. Este método aproxima-se mais àquele que é utilizado por humanos na sumarização de textos, onde a sumarização é mais do que a extração de frases dos textos originais (Hobson (2007)).

Os sumários podem ser *indicativos* ou *informativos* (Edmundson (1969), Hobson (2007)). Os indicativos têm a finalidade de dar a conhecer a estrutura e conteúdo do texto, para que o leitor possa decidir que textos pretende ler. São geralmente muito pequenos: algumas frases ou mesmo apenas algumas palavras-chave relacionadas com a área ou o tópico dos textos. Os sumários informativos pretendem ser *substitutos* dos documentos de origem, isto é, contêm a informação mais importante e podem ser lidos quando o leitor precisa de informação mas não tem tempo para ler os textos completos. Hahn e Mani (2000) referem ainda os sumários *críticos* (ou *reviews*) que, para além de serem informativos, transmitem uma opinião ou crítica sobre um determinado assunto ou objeto.

Os sumários podem ser *genéricos* ou *baseados em questões ou tópicos* (*query-based* ou *topic-based*). Nos sumários genéricos todos os temas abordados têm a importância que o autor dos textos lhe atribuiu, não se tendo em consideração os interesses específicos do leitor (Hovy e Lin (1998)).A ideia essencial dos sumários baseados em questões é que o sumário satisfaça um pedido de informação expresso numa *query* ou questão. Desta forma poderão ser produzidos inúmeros sumários diferentes, a partir dos mesmos documentos, dependendo dos interesses do utilizador/leitor e da questão colocada.

A sumarização automática pode ser dois tipos: *supervisionada* ou *não supervisionada*. Segundo Nomoto e Matsumoto (2001), as abordagens supervisionadas usam sumários elaborados por humanos para extrair atributos ou os parâmetros dos algoritmos, enquanto nas não supervisionadas toda a aprendizagem é feita sem se olhar a qualquer sumário manual.

Num processo *tradicional* de sumarização, assume-se que o documento (ou conjunto de documentos) a sumarizar é estático. Mas o que acontece na realidade é que os documentos sobre um determinado tópico estão em constante evolução ao longo do tempo, estando sempre a surgir documentos com novas informações relativas ao tópico em análise.

É desta necessidade de sumarizar um conjunto dinâmico ou evolutivo de documentos que surge o *update summarization*, cujo objetivo principal é informar o leitor das novidades que surgiram sobre um determinado assunto ou tópico, tendo em consideração o conhecimento que o leitor já tem de documentos lidos anteriormente. Assim, *update summarization* é uma melhoria da sumarização automática de textos, já que ao fazer o sumário tem-se em consideração a informação disponível sobre o conhecimento prévio dos leitores (Boudin e Torres-Moreno (2009)).

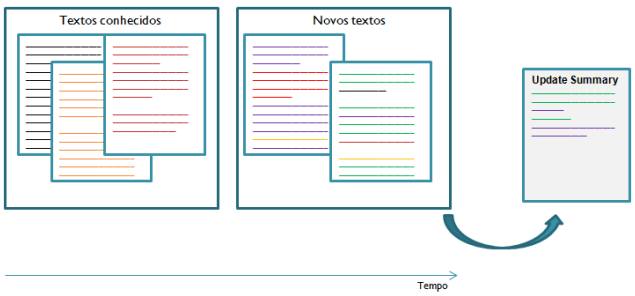


Figura 2.2 - *Update Summarization*

De uma forma muito simples, o *update summarization* consiste na sumarização de um conjunto de textos, tendo em consideração o conhecimento que já existe de textos anteriores. Está, portanto, relacionado com deteção de novidade. Enquanto o objetivo da deteção de novidade é determinar se a informação é nova, o objetivo do *update summarization* é extrair e sumarizar essa informação nova (Delort e Alfonseca (2012)).

Li *et al.* (2013) afirmam que um bom *update summary* deve estar de acordo com o conteúdo principal dos novos documentos, deve ser relevante para o tópico indicado, não deve repetir o conteúdo dos documentos já conhecidos e não deve ser redundante.

O *update summarization* emergiu da *Document Understanding Conference* (DUC) 2007, sendo então uma tarefa experimental. Na *Textual Analysis Conference* (TAC) 2008 passou a ser tarefa principal.

2.2. Trabalhos Relacionados na Área de Sumarização

Apresentamos de seguida vários trabalhos relacionados com a sumarização automática de textos e que são apresentados na literatura de referência, sendo evidenciadas várias abordagens e métodos.

2.2.1. Trabalhos Pioneiros

Apesar de a Sumarização Automática de Textos ter merecido muita atenção nos últimos anos, é um tema que já é estudado há muito tempo. A primeira proposta deve-se a Luhn (1958) com o trabalho pioneiro *The Automatic Creation of Literature Abstracts*, cujo objetivo é a criação de *abstracts* de artigos técnicos. A ideia base é que a frequência das palavras no artigo é um indicador da sua relevância: as palavras importantes repetem-se ao longo do texto. As palavras relevantes, aquelas cuja frequência está num intervalo pré-definido (em que são ignoradas as palavras muito frequentes – *stop words* – e as palavras que surgem poucas vezes), formam a lista dos termos importantes. É com base nesta lista que é calculado o fator de significância de cada frase, considerando que quanto mais palavras da lista existirem na frase, mais importante ela é. As frases com maior valor irão formar o *abstract* do artigo.

Baxendale (1958) faz um estudo com o intuito de descobrir onde estão localizadas as palavras mais importantes. Dos parágrafos que estudou, concluiu que em 85% dos casos, a primeira frase era a mais relevante, sendo que em 7% era a última frase. Apesar de importante, esta conclusão é demasiado generalista e não tem em consideração as especificidades inerentes ao género ou domínio dos documentos (Lin e Hovy (1997)).

Edmundson (1969) propõe quatro métodos que permitem atribuir pesos às frases:

- *Método das palavras indicativas* (*cue words*) – a presença de determinadas palavras podem indiciar a importância (ou não) da frase. Edmundson refere a existência de três dicionários de palavras: *Bonus words* (palavras positivamente relevantes, por exemplo, *significante* ou *importante*), *Stigma words* (palavras negativamente relevantes, como *impossível* ou *dificilmente*) e *Null words* (palavras não relevantes);
- *Método das palavras chave* – palavras chave são aquelas que, não sendo *cue words*, têm uma frequência superior a um dado limiar. O peso das palavras difere conforme a sua frequência no texto (método semelhante à lista de termos importantes de Luhn, apesar de utilizar um algoritmo diferente);
- *Método das palavras do título* – parte-se do pressuposto que os títulos são cautelosamente escolhidos pelos autores, o que indicia que frases que contêm palavras que fazem parte dos títulos são importantes;
- *Método da localização* – tal como no trabalho realizado por Baxendale (1958), considera-se que a posição das frases é importante. Edmundson dá especial importância à primeira frase do primeiro parágrafo e à última do último parágrafo.

O peso (ou importância) de uma frase é calculado através de uma combinação linear dos pesos obtidos através dos quatro métodos anteriores, sendo as frases mais importantes incluídas no sumário.

Estas indicações para calcular as frases mais relevantes, apesar de já terem muitos anos, continuam a ser importantes atualmente. Aliás, Lin e Hovy (1997) afirmou mesmo que, à data (1997), continuava a ser dos melhores métodos, conseguindo desempenhos superiores a algumas abordagens bastante mais recentes.

2.2.2. Abordagem Baseada em Atributos

Esta abordagem segue o mesmo método utilizado por Edmundson (1969), referido anteriormente.

O essencial desta abordagem é identificar quais os atributos que melhor identificam as frases consideradas relevantes. Depois uma combinação linear dos atributos (ponderados pelos respetivos pesos) dará a pontuação de cada frase. As frases mais bem classificadas serão selecionadas para o sumário.

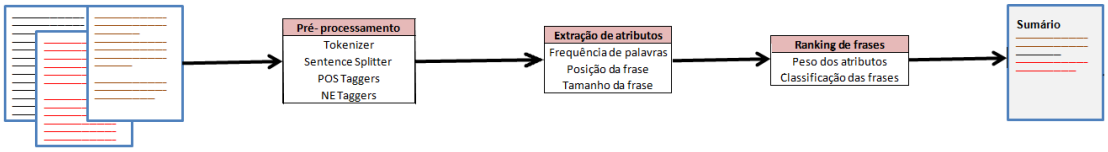


Figura 2.3 – Arquitetura de um Sistema Baseado em Atributos (Adaptado de adaptada de Kumar e Salim (2012))

Neste tipo de abordagem têm sido utilizados diversos atributos para a seleção das frases (Edmundson (1969), Kupiec *et al.* (1995) e Kumar e Salim (2012)): frequência das palavras, palavras do título ou cabeçalho, posição da frase no texto, posição da frase no parágrafo, tamanho da frase, palavras indicativas, nomes próprios, palavras com maiúscula, palavras temáticas, entre outros.

Há na literatura vários trabalhos que usam aprendizagem supervisionada, aprendendo os classificadores tendo em consideração os diversos atributos escolhidos. Foong *et al.* (2010) fazem um levantamento de várias técnicas que têm sido utilizadas em sumarização automática de textos ao longo dos anos, enumerando, por exemplo, naive Bayes, árvores de decisão, redes neuronais, ou máquinas de vetores de suporte.

Kupiec *et al.* (1995) utilizaram o naive Bayes para o seu sumarizador treinável. Utilizando como fonte artigos científicos, os autores treinaram o modelo com vários sumários feitos por humanos. A ideia era que o classificador aprendesse se, perante os atributos de uma frase, ela seria incluída ou não no sumário.

Lin (1999) utilizou uma árvore de decisão para aprender um modelo que combinava vários atributos e verificou que conseguia obter melhores resultados do que os obtidos com o naive Bayes.

Num trabalho essencial para perceber de que parte do texto são extraídas as frases incluídas nos sumários, Lin e Hovy (1997) introduzem a *Optimal Position Policy*, tendo por base a ideia de que as frase-chave tendem a ocorrer em certas zonas

específicas do texto (que dependem do género do texto em análise). Por exemplo, em notícias as frases mais importantes estão logo no início do texto, em outros textos podem aparecer nos *abstracts*, títulos ou nas conclusões.

Como na abordagem baseada em atributos a cada atributo é associado um peso, podem-se atribuir diferentes níveis de importância dos atributos. Isto levanta a questão de saber quais os pesos a atribuir a cada um dos diversos atributos. Kumar e Salim (2012) referem trabalhos que utilizam técnicas como o gradiente descendente, otimização através de enxames de partículas ou algoritmos genéticos, com a finalidade de aprenderem os melhores pesos.

2.2.3. Abordagem Não Supervisionada Baseada em Agrupamento (*Clustering*)

Segundo Kumar e Salim (2012), o *clustering* aplicado à sumarização agrupa frases de acordo com a sua semelhança. Depois de criados os *clusters*, são extraídas frases de cada um deles, selecionando-se as frases que estão mais próximas do centroide. Um centroide, segundo Radev *et al.* (2004), é um conjunto de palavras estatisticamente importantes.

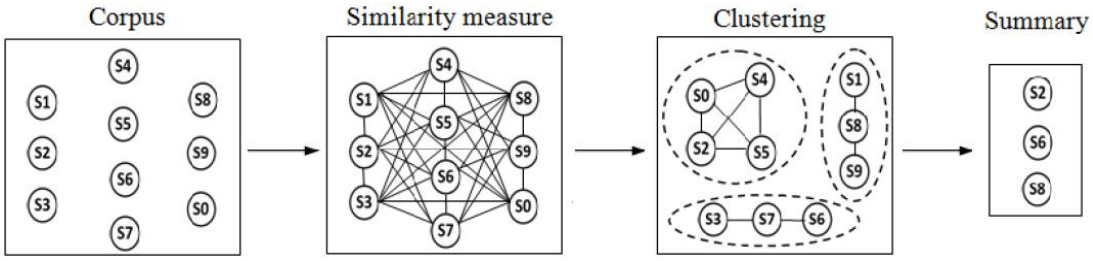


Figura 2.4 – Arquitetura de um Sistema Baseado em *Clustering* (Fonte: Kumar e Salim (2012))

Seki *et al.* (2005) afirma que os sistemas de sumarização baseados em *clustering* distinguem-se por quatro aspetos principais: algoritmos de *clustering*, unidade de *cluster* (por exemplo, frases, parágrafos, documentos), estratégia de extração de frases e tamanho dos *clusters*.

Nomoto e Matsumoto (2001) propõem uma abordagem centrada na informação, não estando focada em fazer sumários o mais parecidos com os produzidos por humanos, mas sim que representem o mais possível a informação contida nos textos sumarizados. Neste sentido implementam um sistema de sumarização de um único documento, baseado em *clustering* (usam o *k-means*), que primeiro encontra os

diversos tópicos do texto. Cada tópico é composto pelas frases que lhe estão relacionadas. Em seguida procedem à redução da redundância, escolhendo em cada tópico (*cluster*) a frase mais representativa.

Um trabalho importante nesta abordagem foi apresentado por Radev *et al.* (2004). Os autores incluíram uma abordagem baseada em centroides no seu sistema de sumarização multidocumento, MEAD[2]. Neste sistema os centroides são pseudo-documentos que têm palavras com um valor TF*IDF[3] acima de um *threshold* predefinido. Considera-se que as frases que contêm mais palavras do centroide são mais centrais, logo mais representativas da informação relacionada com o tópico principal do *cluster*.

2.2.4. Abordagem Não Supervisionada Baseada em Grafos

Neste tipo de abordagens utiliza-se a teoria dos grafos para representar as ligações entre os objetos. No caso de textos, as relações representam a similaridade entre as frases. Os documentos são representados por um grafo pesado não direcionado, em que as frases são os nós (ou vértices) e as ligações entre os nós (as arestas) são criadas para evidenciar as relações que existem entre as frases. Umas frases podem ser semelhantes a outras, enquanto algumas outras frases não partilham quase nenhuma informação entre si. Segundo Kumar e Salim (2012), a medida de similaridade mais utilizada é a medida do cosseno e uma ligação (aresta) apenas existirá entre duas frases se o peso da similaridade exceder um determinado *threshold* predefinido. Depois do grafo criado são identificadas as frases mais importantes.

Erkan e Radev (2004) desenvolveram o *LexRank* (baseado no *PageRank*, proposto por Brin e Page (1998)) com o intuito de medir a saliência das frases através de um grafo, tendo por base o conceito de centralidade do vetor próprio. Esta medida baseia-se na ideia de que a centralidade de um nó é definida pela centralidade dos nós com que está relacionado e assume que as ligações podem ter pesos diferentes,

[2] MEAD é uma plataforma de sumarização multilíngue que implementa diversos algoritmos de sumarização. Está disponível em http://www.summarization.com/mead/.
[3] TF (*Term Frequency*) * IDF (*Inverse Document Frequency*)

valorizando, assim, não só a sua quantidade, mas sobretudo a sua qualidade (Gama *et al.* (2012)).

O *LexRank* considera que frases que são semelhantes a muitas outras são mais salientes (ou centrais) para o tópico em análise (Erkan e Radev (2004)). Mas tem também em consideração o *prestígio*: não basta ser semelhante a muitas outras frases, é também relevante a qualidade dessas frases.

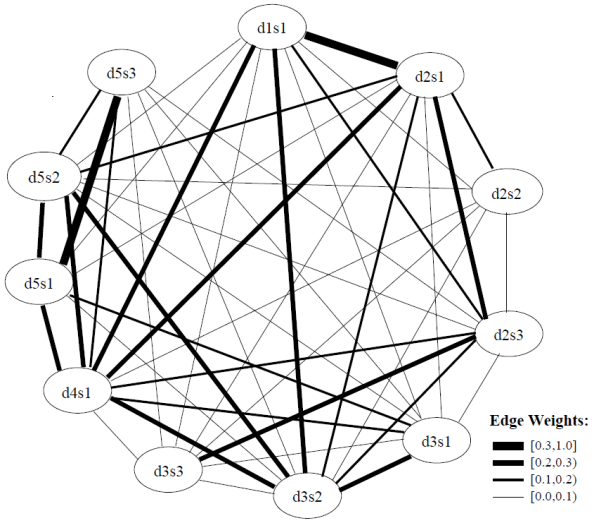


Figura 2.5 – Grafo de frases pesado (Fonte: Erkan e Radev (2004))

O score LR de uma frase *u* é apresentado da seguinte forma em Otterbacher *et al.* (2009)

$$R_{LR}(u) = \frac{d}{N} + (1-d) \sum_{v \in adj[u]} \frac{w(v,u)}{\sum_{z \in adj[v]} w(v,z)} R_{LR}(v) \tag{2.1}$$

em que *N* é o número de frases (vértices) do grafo; *d* é um parâmetro a ser determinado empiricamente e que indica a probabilidade de saltar do nó atual para um outro, aleatoriamente; *adj[u]* são as frases vizinhas e *u* no grafo e *w*(*v*,*u*) corresponde ao peso do vértice entre as frases *v* e *u*. O valor LR de uma frase é, portanto, a soma de um valor constante e a média pesada dos valores LR das suas frases vizinhas.

Na mesma altura em que o *LexRank* foi desenvolvido, num projeto paralelo surge o *TextRank* (Mihalcea e Tarau (2004) e Mihalcea (2004)), com uma ideia bastante semelhante, também baseada na utilização de grafos para obter o *ranking* das frases. O *TextRank* é baseado no *PageRank* e no HITS e foi aplicado à sumarização de textos (apenas de um documento) e à extração de *keywords*.

Otterbacher *et al.* (2005) apresentam o *Topic-sensitive LexRank* (*T-LexRank*) que adequa o *LexRank* à sumarização baseada em tópicos ou questões (*queries*). O *LexRank* foi desenvolvido para a sumarização genérica, enquanto o *Topic-sensitive LexRank* introduz um fator de enviesamento, modificando os pesos do grafo para valorizar as frases mais relacionadas com o tópico.

Valizadeh e Brazdil (2013) propõem uma modificação do *LexRank* para passar a incorporar o conceito de *densidade* dos documentos (um documento será mais denso se as suas frases forem mais similares), de forma atribuir um maior valor às frases provenientes de documentos mais densos.

2.3. Trabalhos Relacionados na Área de Update Summarization

O *update summarization* foi introduzido na DUC 2007, sendo apenas a partir daí que começam a surgir trabalhos nesta área. Contudo a tarefa de deteção de novidades em documentos (muito relacionada com *update summarization*) é já estudada há mais tempo. Nas conferências TREC de 2002 a 2004, havia uma tarefa em que era fornecido um conjunto de documentos ordenados temporalmente e em que se solicitava que fossem indicadas as frases simultaneamente relevantes e que representassem uma certa novidade em relação ao que já tinha sido visto nos documentos anteriores (Soboroff e Harman (2005)).

Um trabalho pioneiro para a sumarização de documentos evolutivos, foi o algoritmo *TimedTextRank* (Wan (2007)). Este algoritmo, adaptado do *TextRank* (Mihalcea e Tarau (2004)), introduz o conceito temporal na ordenação dos documentos, destacando a importância de valorizar o contributo dos documentos mais recentes, atribuindo-lhes mais relevância do que aos documentos antigos, o que permite que maior novidade seja incluída nos sumários.

Delort e Alfonseca (2012) afirmam que maioria das soluções apresentadas para resolver o problema de *update summarization* aplicam abordagens próprias para a sumarização multidocumento, adicionando-lhe alguma funcionalidade para remover as frases que contêm informação redundante com os documentos já conhecidos.

Um exemplo disto é o sistema proposto por Boudin e Torres-Moreno (2009), onde é implementada uma abordagem baseada em maximização-minimização. A ideia central é maximizar a relevância das frases escolhidas, minimizando, ao mesmo

tempo, a redundância com o histórico dos documentos já conhecidos. Para isto utilizam um método a que chamaram *Novelty Boosting* que é responsável por detetar termos importantes, e que ainda não foram mencionados anteriormente, sendo considerados *novidade*. Estes termos são depois utilizados na seleção das frases que irão formar o sumário.

Segundo os autores, este método é bastante simples, rápido e eficiente, contudo apresenta alguns aspetos que têm de ser melhorados, nomeadamente o facto de o *Novelty Boosting* detetar como novidade vários termos que não estão relacionados com o tópico em análise. Por outro lado, dada a simplicidade do método e o facto de não haver qualquer tipo de tratamento linguístico leva a que o sumário gerado possa não ser muito fluente.

Aggarwal *et al.* (2009) propõem uma implementação com dois componentes principais: sumarização e deteção de novidade. Para a sumarização dos documentos usam *clustering* (usando o algoritmo *k-means*), selecionando para o sumário a frase, de cada *cluster*, que tem mais conteúdo informativo. O processo inicia-se com a sumarização dos documentos já conhecidos (A). Posteriormente, cada frase do(s) novo(s) documento(s) (B) é comparada com o sumário de A, selecionando-se as frases que têm informação nova. As frases de B com maior novidade são incluídas no sumário, obtendo-se assim o *update summarization*. Neste sistema, os autores recorrem à WordNet[4] para calcularem a semelhança entre frases usando a similaridade semântica[5].

Bossard e Rodrigues (2011) também utilizam uma abordagem baseada em *clustering*, num sistema a que chamaram CBSEAS (*Clustering Based SentenceExtractor for Automatic Summarization*). A inovação deste trabalho foi o uso de um algoritmo genético para otimizar a combinação dos diversos parâmetros de entrada do sistema. O CBSEAS foi testado com os parâmetros antes e depois da otimização, tendo-se verificado melhorias importantes.

Li *et al.* (2008) apresentaram o PNR^2, baseado num grafo de frases. Geralmente num grafo de frases, uma frase é considerada importante se se correlaciona com

[4] A WordNet é uma grande base de dados lexical da língua inglesa. Pode ser consulta em http://wordnet.princeton.edu/.

[5] Na similaridade semântica duas palavras são consideradas semelhantes não apenas quando são exatamente iguais, mas também quando são diferentes mas têm o mesmo significado.

outras frases importantes, sendo isto considerado um reforço positivo. A grande inovação do PNR^2, e que permite a sua utilização no *update summarization*, é a atribuição de reforços quer positivos quer negativos.

Considerando que há um conjunto de documentos já conhecidos (A) e um conjunto de novos documentos (B), a ideia é que uma frase receba uma influência positiva das frases correlacionadas que são do mesmo conjunto de documentos, e que receba uma influência negativa de frases com que está correlacionada, mas que sejam do outro conjunto. Isto pode ser observado na seguinte figura.

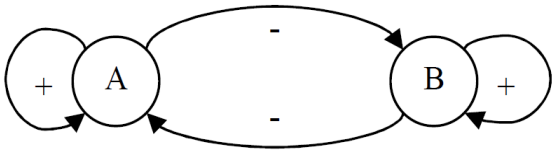


Figura 2.6 – Positive and Negative Reinforcement (Fonte: Li *et al.* (2008))

O reforço positivo evidencia que uma frase é mais importante se se associa a frases importantes de mesma coleção. Ao mesmo tempo, o reforço negativo indica que a frase será menos importante se se relacionar com frases importantes da outra coleção, uma vez que essas frases estarão a repetir a informação que é suposto já ser conhecida.

A implementação do PNR^2 pode gerar sumários genéricos ou pode ser facilmente adaptada para gerar sumários baseados em *queries* ou tópicos.

O PNR^2 calcula a semelhança entre frases usando a medida do cosseno entre os dois vetores de palavras, considerando ainda que duas frases iguais têm uma semelhança igual a 0.

A utilização do PNR^2 irá classificar (ou ordenar) as frases, sendo que as que tiverem melhor classificação serão as candidatas a serem incluídas no sumário. Há contudo necessidade de verificar a redundância, para garantir que a semelhança entre quaisquer duas frases incluídas no sumário não ultrapassa um determinado *threshold* definido.

O PNR^2 também poderá apresentar alguns aspetos que poderão ser menos positivos. Li *et al.* (2013) afirmam que a integração dos documentos de A e de B podem inviabilizar as seleção das frases mais salientes. Os próprios autores do PNR^2 (Li *et al.* (2008)) apontam para o facto de as frases antigas influenciarem negativamente as novas, salientando que isso pode ser incorreto (caso essas frases

antigas abordem novos tópicos emergente). Da mesma forma, as frases novas, que presumivelmente expressam novas ideias ou tópicos, podem não ser mais que continuações dos tópicos antigos. Para estes problemas, os autores propõem a classificação prévia de todas as frases em duas categorias: frases orientadas a tópicos antigos e frases orientadas a novos tópicos (é importante referir que esta ideia é sugerida apenas como trabalho futuro).

A proposta de Wang e Li (2010) foge um pouco ao conceito de *update summarization* como foi apresentado nesta tese. O objetivo deles é terem um sumário sempre atualizado com todas as novidades que forem surgindo. Ao contrário de todas as implementações analisadas até aqui, que tratam todos os documentos em lote, o sistema proposto sumariza os documentos logo que eles fiquem disponíveis, em tempo real. Para isto utilizam *clustering* hierárquico incremental, sendo utilizado o algoritmo COBWEB.

2.4. Avaliação

Um aspeto muito importante é a avaliação dos sumários produzidos. Esta é uma tarefa crítica, porque permite aferir a qualidade do sumário e verificar, por exemplo, se ele contém a informação relevante dos documentos. Dificilmente se poderá afirmar que um texto que contenha algumas frases extraídas aleatoriamente de um documento seja um sumário desse documento. É pois essencial ter metodologias e métricas que permitam avaliação e comparação da qualidade de sumários.

Carenini *et al.* (2011) distinguem técnicas de avaliação *intrínsecas* de *extrínsecas*: as intrínsecas, que são as mais utilizadas, avaliam o conteúdo do sumário (comparando com sumários criados por humanos ou com o documento completo); as extrínsecas (que não vão ser abordadas neste documento) avaliam a utilidade do sumário para a realização de determinadas tarefas (um sumário é feito para facilitar uma dada tarefa, as métricas extrínsecas avaliam quão bem isso é conseguido).

Podemos considerar que o objetivo da sumarização automática de textos é conseguir fazer sumários com a mesma qualidade daqueles que são produzidos por humanos, que são geralmente utilizados como sumários modelo ou de referência (sendo muitas vezes referidos como *golden standard*). O problema é que normalmente não existe um sumário ideal (um que possa ser considerado o melhor

de todos). Diferentes pessoas escreverão sumários diferentes, sobre o mesmo documento (ou conjunto de documentos). Mais, a mesma pessoa, a fazer várias vezes a sumarização do mesmo documento, irá produzir sumários diferentes. No estudo feito por Rath *et al.* (1961) concluiu-se que a seleção de frases a incluir nos sumários é muito variável: a sobreposição dos sumários feitos pela mesma pessoa, do mesmo documento, com um intervalo de 8 semanas, foi apenas de 55%; por outro lado, quatro pessoas diferentes a sumarizar o mesmo texto apenas tiveram 25% das frases iguais.

As diversas edições de competições como o TREC[6], DUC[7] ou TAC[8] têm disponibilizado vários recursos que vão desde textos para treino e testes até resumos produzidos por humanos, cuja finalidade é servirem de base de comparação com os sumários produzidos automaticamente pelos diversos sistemas submetidos às competições (Das e Martins (2007)). Muitos dos trabalhos recentes consultados neste relatório (por exemplo, Boudin e Torres-Moreno (2009), Bossard e Rodrigues (2011) e Valizadeh e Brazdil (2013)) utilizam o material disponibilizado por estas conferências quer para avaliar o desempenho das suas propostas quer para proceder à comparação com os resultados obtidos por outras abordagens ou metodologias.

2.4.1. Medidas ROUGE

ROUGE, acrónimo de *Recall-Oriented Understudy for Gisting Evaluation*, apresentado por Lin e Hovy (2003) e Lin (2004a, b). Consiste num conjunto de métricas e num software[9] que permite fazer a avaliação automática de sumários, tendo por base um conjunto de sumários de referência (criados manualmente).

São disponibilizadas as seguintes métricas, descritas por Lin (2004b):

- ROUGE-N – coocorrência de n-gramas. Mede a sensibilidade, em termos de n-gramas, entre o sumário a ser avaliado e os sumários de referência;
- ROUGE-L – utiliza uma combinação da sensibilidade, precisão e da maior subsequência comum (em inglês, *Longest Common Sequence – LCS*) entre

[6] http://trec.nist.gov/
[7] http://duc.nist.gov/
[8] http://www.nist.gov/tac
[9] O software encontra-se disponível em http://www.berouge.com/Pages/default.aspx.

o sumário a avaliar e os sumários de referência, para calcular a medida-F Hobson (2007);

- ROUGE-W – *Weighted Longest Common Sequence*, maior subsequência comum pesada (para valorizar subsequências consecutivas);
- ROUGE-S – coocorrência de pares de palavras, pela ordem em que surgem na frase, podendo estar separadas por outras palavras (*Skip-Bigram*). A partir desta métrica pode-se utilizar também a ROUGE-SN, que permite indicar qual é a distância máxima (N) entre as duas palavras;
- ROUGE-SU – acrescenta ao ROUGE-S as coocorrências de unigramas.

Hobson (2007) afirma que o ROUGE-1 é o preferido dos investigadores para avaliação de sumários de um único documento, enquanto o ROUGE-2 é o preferido quando se trata da avaliação de sumários multidocumento.

2.4.2. Método da Pirâmide

Este método foi apresentado por Nenkova e Passoneau (2004). Ao contrário do ROUGE que é automático, este é semiautomático. Foi desenvolvido para avaliar de forma fiável a qualidade do conteúdo dos sumários.

Tal como no ROUGE são usados sumários de referência, elaborados por humanos. Mas neste método há ainda a anotação dos sumários para a identificação de SCUs (*Summary Content Units*) que são unidades semânticas não maiores que uma frase e que basicamente representam um conceito. Estas SCUs são identificadas e depois verifica-se em quantos sumários surgem. Cria-se assim uma pirâmide de camadas, sendo cada camada constituída por SCUs que surgem no mesmo número de sumários. Assim, as SCUs que apenas aparecem num sumário estão na base da pirâmide; as que aparecem em todos os sumários estão no topo da pirâmide.

A ideia essencial é que um sumário será considerado mais informativo se contiver as informações ou conceitos que surgem com mais frequência nos sumários de referência.

Hobson (2007) refere que a grande vantagem deste método sobre o ROUGE é não se limitar a comparar palavras, comparando antes unidades semânticas, garantindo assim a comparação de conceitos/ideias, mesmo que não sejam expressas pelas mesmas palavras. Por outro lado, o facto de não ser completamente automático,

exigindo grande esforço humano (anotadores para identificarem as SCUs nos sumários de referência e depois verificarem a sua ocorrência nos sumários a avaliar), faz com que possa ser um método muito demorado e dispendioso.

2.4.3. Qualidade Linguística

Carenini *et al.* (2011) referem que muito importante na avaliação de um sumário é a análise da qualidade linguística e a facilidade de leitura, avaliando a gramática, a clareza, estrutura, coerência e a não redundância. O método da pirâmide ou o ROUGE incide mais sobre o conteúdo do sumário, enquanto a qualidade linguística se centra mais na forma como esse conteúdo se encontra escrito. A grande dificuldade é que este tipo de avaliação exige que sejam pessoas a fazer a avaliação.

A qualidade linguística e a facilidade de leitura fazem parte do modelo de avaliação da TAC, sendo avaliadas as categorias: gramática, não redundância, clareza, estrutura e coerência Carenini *et al.* (2011). Na TAC, à qualidade linguística juntam-se ainda mais três métodos para se chegar à avaliação final: pirâmide, ROUGE e avaliação global (Lapalme *et al.* (2008)).

2.4.4. Avaliação Sem Resumos de Referência

Louis e Nenkova (2013) apresentam várias métricas para serem utilizadas em situações em que não estejam disponíveis resumos de referência (criados por humanos). A ideia é conseguir-se fazer a avaliação sem nenhuma (ou pouca) intervenção humana. Foi disponibilizada a ferramenta SIMetrix[10] (*Summary Input similarity Metrics*) que permite calcular todas as métricas propostas.

Para a avaliação dos sumários tendo apenas por base os textos de origem, Louis e Nenkova (2013) sugerem métricas de três classes diferentes: similaridade da distribuição, semelhança do sumário e utilização de palavras representativas do tópico.

As autoras afirmam que as métricas conseguem bons desempenhos, havendo na maioria dos casos uma grande correlação com as métricas que utilizam resumos escritos por humanos (por exemplo, o ROUGE).

[10] Está disponível para download em http://www.seas.upenn.edu/~lannie/IEval2.html.

3. Projeto de Sumarização e *Update Summarization*

Neste capítulo são apresentadas as etapas realizadas no desenvolvimento deste trabalho.

3.1. Processo de Sumarização Extrativa

O processo que implementamos para criar os sumários segue um conjunto de passos bem definidos e que podem ser observados na figura seguinte.

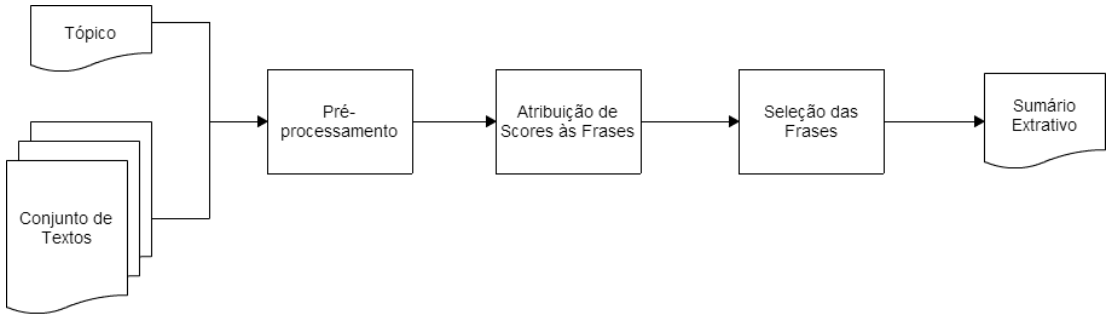


Figura 3.1 – Processo de Sumarização Extrativa

O input é o conjunto de textos que pretendemos sumarizar, acompanhados por um tópico (no caso de estarmos perante um processo de sumarização baseado em tópicos).

As três etapas principais, que analisaremos em detalhe nas secções seguintes, são:

- Pré-processamento – onde é feita a limpeza dos textos e são realizadas várias tarefas que transformam os textos em dados estruturados, passíveis de serem manipulados pelos diversos algoritmos responsáveis pela sumarização;
- Atribuição de scores (ou pontuação) às frases – aqui está o cerne dos sistemas de sumarização extrativa. É nesta tarefa que os diversos modelos se distinguem. A diferença prende-se essencialmente na forma como cada sistema avalia e classifica cada uma das frases. No âmbito desta tese iremos explorar diversos sistemas de dois tipos de abordagens distintas: modelos de aprendizagem não supervisionada, baseados em grafos, e modelos de aprendizagem supervisionada;

- Seleção das frases – tendo por base a ordenação dos scores da tarefa anterior, as frases são selecionadas dos textos originais, criando-se assim o sumário (que é o output). Sublinhamos que nesta tese abordamos apenas a sumarização extrativa.

O processo de *update summarization* é muito semelhante, com a diferença a residir essencialmente em dois aspetos: no input, que passa a incluir o conjunto de textos já conhecidos (histórico); e nos algoritmos de atribuição de pontuação de frases, que agora têm de ter em consideração que apenas se pretende sumarizar o que é novo em relação ao histórico.

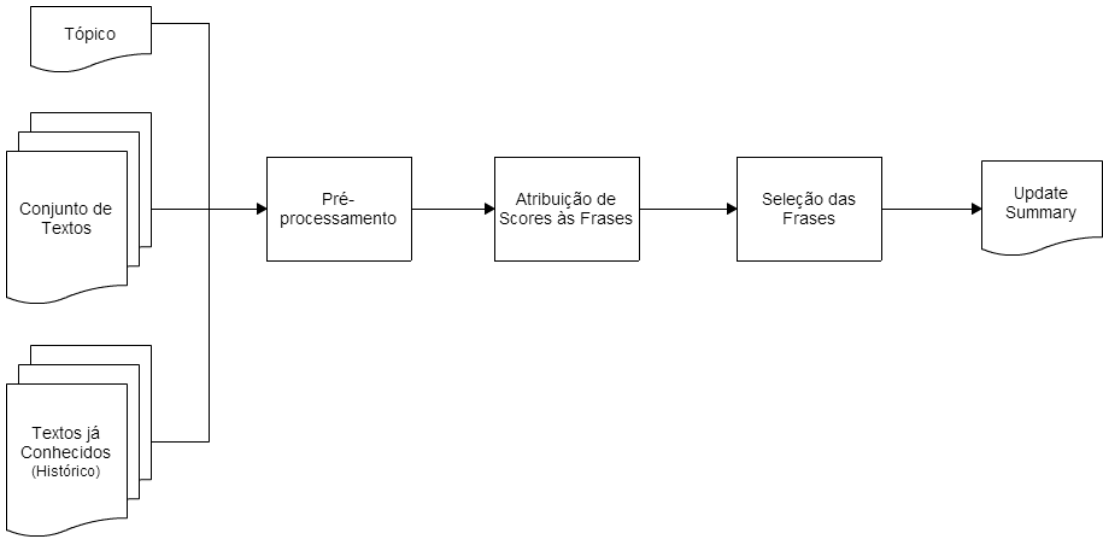


Figura 3.2 – Processo de *Update Summarization*

De seguida iremos analisar cada uma destas fases.

3.2. Pré-processamento

Os textos a serem sumarizados podem ser provenientes de várias fontes e terem diferentes formatos. Geralmente quem produz os textos não os estrutura a pensar que eles poderão, porventura, no futuro, ser alvo de um processo automático de sumarização. Desta forma, é necessário sujeitar os textos a diversas alterações antes de iniciar o processo de sumarização propriamente dito. Depois é preciso *transformar* os textos em dados estruturados, com os quais se irá lidar nos algoritmos desenvolvidos, para proceder à sumarização automática de textos.

Passamos a apresentar as tarefas envolvidas.

3.2.1. *Limpeza* dos Textos

Dos conjuntos de textos utilizados, no caso textos fornecidos por agências noticiosas, alguns são fornecidos em texto *limpo*, sem qualquer tipo de informação para além do texto propriamente dito; contudo, a maioria tem informações que nada têm que ver com o essencial.

Por exemplo, a maioria dos textos utilizados são fornecidos em formato *xml*, sendo necessário remover as respetivas *tags* e extrair o texto que se pretende sumarizar. Mas mesmo o texto propriamente dito necessita de ser alvo de várias tarefas de limpeza.

Inúmeros textos têm meta-informações para os leitores, como por exemplo "(*story can end here. Optional material follows)"*, "(*begin optional trim)*", "*end optional trim*", "*attention - adds quotes details*".

Outros textos incluem dados sobre os autores, como o nome, o correio eletrónico ou a entidade para quem trabalham. Esta informação também não tem interesse para a sumarização.

Implementar um programa que seja capaz de detetar e corrigir, de forma automática, todos estes detalhes foi uma tarefa que exigiu muito tempo e sabemos que há detalhes que escaparam ao sistema implementado. Estamos convencidos que o desempenho de todo o processo de sumarização depende muito desta fase. Quanto menos *ruído* existir nos textos, melhor decorrerão todos os passos seguintes, e, consequentemente, maior será a qualidade dos sumários produzidos.

3.2.2. Extensão do Tópico

Uma vez que o tópico fornecido é um texto relativamente curto, geralmente composto por um título (que pode existir ou não) e por uma frase descritiva, decidimos que seria importante ampliar o vocabulário do tópico, usando para isso sinónimos das suas palavras.

Recorremos à WordNet (Princeton University (2010)) para procurar sinónimos dos substantivos e adjetivos que fazem parte do tópico.

As figuras seguintes mostram um tópico antes e depois da extensão. As cores indicam as palavras do tópico original e os respetivos sinónimos que foram

acrescentados. Podemos observar, por exemplo, que para o termo "*struggle*" foram acrescentados os sinónimos "*battle*" e "*conflict*".

Tamil Tigers in Sri Lanka. Describe developments in the struggle between Tamil rebels and the government of Sri Lanka.

Figura 3.3 – Tópico D0910B (TAC 2009)

Tamil Tigers in Sri Lanka. Describe developments in the struggle between Tamil rebels and the government of Sri Lanka. Tamil Liberation Tigers of Tamil Eelam LTTE Tamil Tigers Tigers World Tamil Association World Tamil Movement battle conflict authorities governance governing government activity political science politics regime

Figura 3.4 – Tópico D0910B (TAC 2009) depois da extensão

3.2.3. Divisão dos Textos em Frases

Na sumarização extrativa, o essencial é determinar quais as frases que devem ser selecionadas para o sumário. Isto faz com que seja necessário identificar as frases de cada um dos textos.

Esta tarefa, que a princípio aparenta ser trivial, não é fácil de implementar, já que não é suficiente dividir as frases pelos caracteres de pontuação (.,!?, entre outros). Há inúmeras exceções em que esta regra não se aplica, por exemplo, abreviaturas, endereços eletrónicos (quer sejam de email ou endereços web), ou simplesmente horas (a.m. ou p.m.).

Fazendo uma análise às frases obtidas com a divisão simples, rapidamente percebemos a sua ineficácia.

Com a utilização do *package*[1] NLP (Hornik (2014)) na divisão de frases, houve uma melhoria nas frases obtidas. Mas mesmo assim, foi preciso proceder a diversas correções, algumas básicas. Por exemplo, o *package* NLP não consegue dividir corretamente frases que contenham endereços eletrónicos.

[1] Neste trabalho, *package* refere-se a um pacote de software externo que irá ser usado na plataforma de programação R. Os *packages* geralmente disponibilizam aos utilizadores um conjunto de funcionalidades não implementadas de raiz no R. Por exemplo, o *package* NLP (Natural Language Processing), tem um vasto conjunto de funções para o processamento de linguagem.

Um problema que não ficou resolvido foi a separação de frases de discurso direto. Neste caso pretendíamos que as frases não fossem divididas para aumentar a coesão. Este é um problema que temos de tentar resolver no futuro.

3.2.4. Divisão das Frases em Palavras

A divisão das frases em palavras (*tokenização*) consiste na identificação de cada palavra que compõe essa frase. Este processo é geralmente básico, uma vez que os delimitadores das palavras são bem conhecidos (por exemplo, espaços em branco, tabulações, sinais de pontuação, quebras de linha).

3.2.5. Remoção de *Stop Words*

As *stop words* são palavras muito comuns e que não são representativas para os textos, ou seja, palavras que podem ser removidas sem que haja perda para o processo de *Text Mining*[2] (Manning *et al.* (2008)). Como por exemplo os termos *a*, *de*, *pelo*, *quem*, entre muitos outros.

Utilizamos a lista de *stop words* para a língua inglesa, disponível no *package tm* (Feinerer e Hornik (2014)).

3.2.6. *Stemming*

O processo de *stemming* consiste em reduzir as palavras à sua base comum, de forma a facilitar a comparação de vocabulário (Manning *et al.* (2008)). Com o *stemming* conseguimos representar, num único termo, diversas palavras com o mesmo significado mas escritas de forma diferente (por exemplo, por umas estarem no feminino outras no masculino, umas estarem no singular outras no plural, ou por terem sufixos diferentes, entre outros casos).

Utilizamos o algoritmo de Porter (van Rijsbergen *et al.* (1980)), que é o mais comum (Manning *et al.* (2008)), implementado no *package* SnowballC (Bouchet-Vallat (2014)).

[2] Em português, Extração de Conhecimento de Texto. Algumas áreas do *Text Mining* são classificação de documentos, extração de informação, análise de sentimentos, sumarização automática de textos, entre outras.

3.2.7. Outras Operações

Os textos são ainda sujeitos a outras operações de limpeza, nomeadamente a conversão do texto para minúsculas e a remoção de espaços em branco e sinais de pontuação.

Ponderamos também a remoção de números, mas decidimos mantê-los porque consideramos que os números podem ser importantes para a valorização das frases (por exemplo, as datas).

Convém salientar que estas modificações (assim como a remoção de *stop words* e o *stemming*) não serão observadas no sumário criado, o qual será composto pelas frases como surgem nos textos originais.

3.2.8. *Vector Space Model*

Para tornar possível (ou pelo menos para tornar mais fácil) o tratamento computacional dos textos, é habitual proceder à conversão dos documentos num vetor numérico em que cada posição desse vetor indique o peso do termo no documento. Assim, um documento será composto por esse vetor de números, sendo possível fazer comparações e cálculos entre documentos (comparando ou efetuando operações com os respetivos vetores).

Este modelo vetorial, conhecido por *Vector Space Model*, foi proposto por Salton *et al.* (1975) e é o modelo padrão para a representação de documentos e frases na generalidade de processos de *Text Mining* (quer estejam relacionados com classificação de documentos, extração de informação ou sumarização, por exemplo).

À matriz composta por todos os documentos, chamamos *Document Term Matrix*, sendo que cada linha representa um documento e cada coluna (ou atributo) representa um termo.

Desta forma, uma célula tem o peso atribuído ao termo indicado pela coluna, no documento indicado pela linha.

O problema deste tipo de representação é o grande número de atributos (colunas) com que se tem de trabalhar, sendo que o número de células com o valor zero poderá ser enorme. Matrizes que têm uma grande quantidade de elementos com o valor zero são designadas de *matrizes esparsas*.

Importa salientar que nesta representação, os documentos são representados como um *bag-of-words* (literalmente, saco de palavras), em que a ordem deixa de ser tida em consideração. Por exemplo, consideremos dois documentos, um com a frase *"O cão mordeu o homem."* e outro com a frase *"O homem mordeu o cão"*. Neste *bag-of-words*, ambos os documentos são iguais. Isto é evidentemente uma desvantagem. No entanto, esta representação tem a vantagem de ser simples e proporcionar soluções também simples a algumas tarefas, como por exemplo a classificação de textos.

3.2.9. Peso dos Termos

Conforme referido, às palavras ou termos dos documentos são associados valores numéricos. Estes poderão ser 0 ou 1 (num modelo booleano, indicando se o termo está ou não presente nesse documento) ou um qualquer valor que indique o peso (ou importância) desse termo no documento.

O peso poderá ser calculado de diversas formas, por exemplo, tendo em consideração o número de vezes que o termo surge no documento, a frequência do termo ponderada pelo número de vezes que o termo surge num conjunto alargado de documentos, ou pelo relacionamento do termo com o tópico ou questão em análise, entre diversas outras alternativas.

A seguir apresentamos algumas abordagens comuns, incluindo aquela que foi adotada neste estudo.

TF – Frequência do Termo (*Term Frequency*)

Uma das medidas mais simples para indicar o peso de um termo num documento é a simples contagem do número de vezes que esse termo surge no documento.

$$tf(t, d) = n \tag{3.1}$$

Aqui n representa o número de vezes que o termo t ocorre no documento d.

Há implementações mais elaboradas do *term frequency* que têm em consideração não apenas a contagem de termos mas a sua normalização, para evitar, por exemplo, o enviesamento quando se estiverem a comparar documentos de diferentes tamanhos. Pode-se por exemplo dividir o número de ocorrência pelo número total de termos do documento ou pela frequência do termo que surge mais vezes no documento.

IDF – Frequência de Documento Inversa (*Inverse Document Frequency*)

Um termo que apareça muitas vezes e em muitos documentos é um termo pouco relevante ou com fraca capacidade de discriminação de documentos. É por esse motivo que são removidas as palavras comuns (*stop words*). Neste sentido um termo relevante para a discriminação de documentos é aquele que surge muitas vezes num documento, mas que aparece poucas vezes nos outros. Ou seja, não é suficiente analisar as frequências apenas num documento, é necessário analisar a importância do termo tendo em consideração um vasto conjunto de documentos.

É neste contexto que surge o IDF (Jones (1972)), cuja ideia básica é que termos que surgem em muitos documentos deverão ter um peso menor do que os que surgem em menos documentos, já que estes têm um maior potencial discriminativo.

$$idf(t) = \log\left(\frac{|D|}{\{|d|: t \in d\}}\right) \tag{3.2}$$

Em que $|D|$ é o número de documentos e $\{|d|: t \in d\}$ indica o número de documentos em que o termo ocorre.

Há também algumas variantes desta expressão. Uma alteração comum é modificar o denominador para $1 + \{|d|: t \in d\}$ para evitar divisões por zero (no caso do termo não ocorrer no conjunto de textos).

Para um estudo aprofundado sobre o IDF, remetemos para Robertson (2004).

TF*IDF

Um termo será tão mais importante (terá maior peso) se ocorre muitas vezes num documento (é relevante), mas aparece poucas vezes nos outros (tem capacidade de discriminação).

É com esta ideia que se faz a conjugação da frequência dos termos TF com o seu IDF.

$$TF * IDF(t, d) = tf(t, d) * idf(t) \tag{3.3}$$

O TF*IDF é usado há muitos anos no âmbito do *Text Mining* sendo provavelmente a medida mais utilizada para quantificar o peso ou importância dos termos num documento.

TF*ISF

O TF*IDF considera que cada linha do espaço vetorial representa um documento. E isto deve-se a ter sido originalmente pensado para trabalhar ao nível do documento (por exemplo, para extrair os documentos mais relevantes para responder a uma questão).

Contudo, quando se trabalha ao nível da frase, em que não se pretende extrair os documentos, mas sim as frases relevantes, faz sentido considerar o vetor como representação de uma frase. Surge assim o TF*ISF, em que ISF significa Frequência de Frase Inversa (*Inverse Sentence Frequency*).

Há muitos trabalhos que usam esta medida em vez do TF*IDF para determinar as frases relevantes (ver por exemplo, Otterbacher *et al.* (2005), Kogilavani e Balasubramanie (2012) ou Li *et al.* (2008)). Blake (2006) apresenta um estudo comparativo da utilização destas duas medidas de pesagem dos termos.

No âmbito desta dissertação, e depois de se efetuar uma comparação dos resultados obtidos com TF*IDF e TF*ISF (ver resultados), decidimos adotar o TF*ISF, calculando o ISF conforme indicado em Otterbacher *et al.* (2005)

$$isf(t) = \log\left(\frac{N+1}{0.5 + \{|f|: t\ \epsilon\ f\}}\right) \tag{3.4}$$

em que N indica o número total de frases e $\{|f|: t\ \epsilon\ f\}$ representa o número de frases em que o termo t ocorre.

3.2.10. Similaridade do Cosseno

Uma vez que se pretende trabalhar com as frases ou documentos representados por vetores, é importante definir uma medida que permita comparar esses dois vetores.

No âmbito do *Text Mining*, costuma-se utilizar a *similaridade do cosseno*, onde se analisa o ângulo formado pelos dois vetores. Esta medida irá indicar o relacionamento ou ligação que existe entre dois documentos ou duas frases.

Dados dois vetores *A* e *B*, a similaridade do cosseno é calculada da seguinte forma:

$$sim(A,B) = \cos(\theta) = \frac{\sum_{i=1}^{n} A_i * B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} * \sqrt{\sum_{i=1}^{n}(B_i)^2}} \tag{3.5}$$

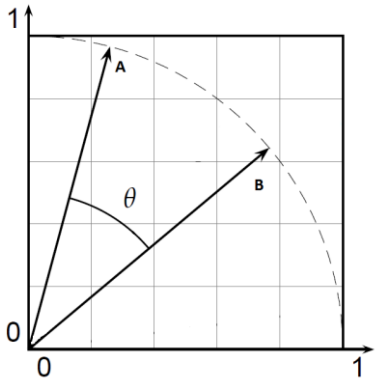


Figura 3.5 – Similaridade do cosseno (Adaptado de Manning et al. (2008))

Por definição a similaridade do cosseno pode assumir valores no intervalo [-1; 1]. O valor -1 indica que os dois vetores são opostos e 1 indica que são iguais.

Contudo, no contexto de *Text Mining*, em que o valor dos vetores é geralmente calculado pelo TF*IDF ou TF*ISF, que nunca podem assumir valores negativos, a similaridade do cosseno dá sempre valores entre [0; 1]. Quando não há nenhum relacionamento entre os dois vetores, toma o valor 0. Se os dois vetores forem coincidentes, o resultado é 1.

Nas secções seguintes iremos analisar os métodos explorados para sumarização, primeiro os não supervisionados, baseados em grafos, e depois os supervisionados. Posteriormente fazemos a mesma análise para os métodos usados em *update summarization*.

3.3. Métodos Não Supervisionados Baseados em Grafos para Sumarização

Neste trabalho foram considerados três sistemas, todos baseados em grafos. Na construção do grafo considera-se que as frases são os nós e as ligações (ou vértices) indicam a relação entre as frases, sendo o valor indicado pela similaridade do cosseno entre as duas frases ligadas.

A ideia subjacente a todos os algoritmos é a criação do grafo e depois proceder a um processo iterativo de atribuição dos scores às frases. Neste processo, em cada iteração os scores são atualizados. O processo termina quando o algoritmo converge para a solução, terminando quando entre duas iterações consecutivas não ocorrer

uma variação dos scores superior a um *threshold* (valor muito reduzido). No final, a cada frase é atribuído o score dado pelo algoritmo.

3.3.1. *Topic-sensitive LexRank* (*T-LexRank*)

Este algoritmo já foi introduzido no capítulo 2. Em termos de algoritmos de sumarização baseados em grafos, o *T-LexRank* é a referência, sendo usado na generalidade dos trabalhos como padrão de comparação.

O *T-LexRank* foi proposto por Otterbacher *et al.* (2005) para adequar o *LexRank* (Erkan e Radev (2004)) à sumarização baseada em tópicos. Tem como ideia essencial que a importância de uma frase não é dada apenas pelo número de frases com que se relaciona, mas também pelo *prestígio* dessas frases, uma ideia semelhante à implementada no *PageRank*, que está na origem do motor de pesquisa *Google*[3] (Brin e Page (1998)).

O valor (score) do *T-LexRank* da frase u, tendo em consideração o tópico T, é calculado tendo em conta os scores de frases vizinhas no grafo:

$$\mathrm{R_{LR}}(u,T) = (1-d)\cdot sim(u,T) + d\cdot \sum_{v\in adj[u]} \left(\mathrm{R_{LR}}(v,T)\cdot \frac{sim(v,u)}{\sum_{z\in adj[v]} sim(v,z)} \right) \quad (3.6)$$

em que *sim*(*v*,*u*) corresponde ao peso da ligação entre as frases *v* e *u*, e *d* (*damping factor*) é um parâmetro a ser determinado empiricamente e que indica a probabilidade de saltar do nó atual para um outro, aleatoriamente.

Na nossa implementação do *T-LexRank* introduzimos duas alterações ao algoritmo original, depois de termos realizado diversas experiências e verificado um melhor desempenho. Estas modificações são baseadas em implementações usuais do *PageRank*. Um dos pressupostos do *LexRank* é que a diagonal da matriz de similaridade é sempre 1. Ou seja, na pior das hipóteses, a frase relaciona-se sempre pelo menos consigo mesma. Na nossa implementação consideramos que diagonal é sempre 0 e se uma frase não tiver nenhum relacionamento com qualquer uma das outras, então consideramos que está relacionada com todas com um peso de $\frac{1}{N}$, sendo *N* o número de frases do grafo.

[3] www.google.pt

Para a implementação do algoritmo seguimos as indicações de Erkan e Radev (2004) e de Otterbacher *et al.* (2005).

3.3.2. Sumarização Baseada na Densidade (*DensityBased*)

Uma alteração ao *T-LexRank* foi proposta por Valizadeh e Brazdil (2013), incorporando o conceito de densidade dos documentos. Este conceito indica quão próximas estão as frases de um documento. Os autores consideram que as frases de um documento com maior densidade deverão ser mais valorizadas, pois contribuirão para um melhor sumário.

A proposta para calcular a densidade de um documento é através da inversa do seu raio.

Alterando a expressão do *LexRank* (no caso de sumarização) genérica ou do *T-LexRank* (para sumarização baseada em tópicos ou questões) de forma a incluir o conceito de densidade, obtém-se o *Density & Graph-Based Ranking Algorithm* (*DensityBased*).

A expressão seguinte mostra como se calcula o score do DensityBased, para sumarização baseada em tópicos, conforme indicado por Valizadeh e Brazdil (2013).

$$\mathrm{R_{DB}}(u,T) = (1-d)\cdot sim(u,T) + d\cdot \sum_{v\,\in adj[u]} \left(\mathrm{R_{DB}}(v,T)\cdot \frac{sim(v,u)}{\sum_{z\in adj[v]} sim(v,z)} \cdot \frac{1}{1+r_u} \right) \tag{3.7}$$

A expressão é semelhante à que foi apresentada para o *T-LexRank*, tendo apenas sido acrescentada a densidade, através da expressão $\frac{1}{1+r_u}$ em que r_u indica o raio da frase *u*, calculado como a distância euclidiana entre essa frase e o centroide do documento

$$r_u = \sqrt{\left(\overrightarrow{X_u} - \overrightarrow{X_0}\right)^2} \tag{3.8}$$

sendo $\overrightarrow{X_0}$ o centroide do documento

$$\overrightarrow{X_0} = \frac{\sum_{i=1}^{N} \overrightarrow{X_i}}{N} \tag{3.9}$$

3.4. Métodos Supervisionados para Sumarização

Nesta secção descrevemos a metodologia que utiliza aprendizagem supervisionada no processo de sumarização.

Foram aprendidos três modelos para sumarização, utilizando *Redes Neuronais* (que designamos de *Sup-RN*[4]), *Máquinas de Vetores de Suporte* (*Sup-SVM*) e *Random Forests* (*Sup-RF*).

Além desses modelos, exploramos também a utilização de dois modelos múltiplos, um heterogéneo (designado *Sup-EnsHetero*) e um homogéneo (*Sup-EnsHomo*).

A aprendizagem supervisionada implica a realização de tarefas que não são necessárias com uma abordagem não supervisionada, nomeadamente a criação do conjunto de dados e o treino de cada um dos modelos.

Cada exemplo do conjunto de treino corresponde a uma frase, a que foi associado uma pontuação (score). A ideia essencial é treinar os modelos para que consigam atribuir um score a novas frases.

3.4.1. Criação do Conjunto de Dados

Para ser possível proceder a um processo de aprendizagem automática temos de criar um conjunto de dados baseados em atributos e valores, a partir dos diversos textos. A ideia essencial é que a cada frase seja associado o seu valor de importância (score).

Isto pode ser analisado como um problema de classificação, sendo que a classe é binária, indicando apenas se a frase deve ou não pertencer ao sumário.

Outra abordagem atribui a cada frase um valor numérico indicativo da sua importância (Wong *et al.* (2008), Bysani *et al.* (2009), Kogilavani e Balasubramanie (2012), Valizadeh e Brazdil (2014b)). Tendo o score de cada frase, é relativamente simples chegar ao sumário. Este é um problema de regressão.

No âmbito deste trabalho, optamos por esta última abordagem.

[4] Para permitir uma melhor fluidez da leitura, neste documento podemos referirmo-nos aos sistemas quer pela sua designação quer pela indicação do algoritmo subjacente. Por exemplo, podemos utilizar *Sup-RN* ou modelo *Redes Neuronais* para indicar o mesmo sistema. Isto aplica-se a todos os sistemas.

Surgem então duas questões relevantes: quais os atributos que devem ser tidos em consideração; como calcular o score de uma frase (ou seja, como calcular o atributo de saída).

O conjunto de dados deve ser criado de forma automática a partir dos textos fornecidos, incluindo o atributo de saída no caso do conjunto de treino, em que há sumários de referência.

De seguida apresentamos os atributos selecionados para as nossas tabelas. Na maioria usamos atributos mencionados noutros artigos (Bysani *et al.* (2009), Valizadeh e Brazdil (2014b) e Kogilavani e Balasubramanie (2012)).

Atributos de Entrada

Posição da frase (*pos*, *pos1* e *pos2*)

A posição da frase no texto é uma característica muito importante e que já vem sendo estudada no âmbito da sumarização há muitos anos (ver, por exemplo, Edmundson (1969)).

Geralmente as frases mais importantes estão no início dos textos. Isto é particularmente evidente no tipo de textos noticiosos. Haverá outros tipos de textos, como por exemplo artigos científicos, em que as frases importantes poderão ocorrer noutro tipo de secções.

No presente trabalho, utilizamos três atributos relativos à posição da frase no texto:

- Posição da frase no documento – índice da frase no documento. A ideia é treinar o modelo de forma a ser capaz de identificar a melhor posição da frase no documento. Isto é importante principalmente no caso de se terem documentos de géneros diferentes.

  $$pos(f_{id}) = i \tag{3.10}$$

  Neste caso, f_{id} é a frase na posição i do documento d.

- Posição relativa da frase no documento – a importância da frase é proporcional à sua posição no texto, ou seja, quanto mais no início, mais importante é.

  $$pos1(f_{id}) = 1 - \frac{i-1}{n} \tag{3.11}$$

Sendo f_{id} a frase na posição i do documento d e n o número de frases desse documento.

- Posição da frase – analisando os textos da TAC, Bysani *et al.* (2009) afirmam que as primeiras três frases do documento contêm o seu conteúdo mais importante. Daí a necessidade de atribuir um valor bastante maior a estas frases.

$$pos2(f_{id}) = 1 - \frac{i}{1000}, \qquad se\ i \leq 3 \tag{3.12}$$

$$pos2(f_{id}) = \frac{i}{1000}, \qquad caso\ contrário$$

Assumindo que o documento tem menos de 1000 frases.

Tamanho da frase (*sent.length*)

O tamanho da frase pode também ser um bom indicador sobre se a frase deve ou não entrar no sumário. Frases muito pequenas não são informativas e frases muito grandes ocupam muito espaço no sumário

$$sent.length(f) = |palavras(f)| \tag{3.13}$$

sendo $|palavras(f)|$ o número de palavras da frase *f*.

Tamanho da frase sem *stop words* (*sent.length.wsw*)

Consideramos que uma frase será mais importante quanto mais palavras informativas tiver. Uma frase com muitas *stop words* será normalmente menos informativa

$$sent.length.wsw(f) = sent.length(f) - |stopwords(f)| \tag{3.14}$$

em que $|stopwords(f)|$ indica o número de *stop words* da frase *f*.

Semelhança e relevância com o tópico (*topic.sim*, *topic.rel*)

Na sumarização baseada em questões ou em tópicos é importante perceber quão semelhante e relevante é uma frase em relação à dada questão ou tópico. Para aferir isso são usados dois atributos:

- Semelhança com o tópico – semelhança do cosseno entre a frase e o tópico.

$$topic.sim(f, T) = \text{sim}\ (f, T) \tag{3.15}$$

- Relevância para o tópico – quanto mais relevante para o tópico for a frase, maior será o valor deste atributo. Em Otterbacher *et al.* (2005) define-se a relevância pela expressão seguinte.

$$topic.rel(f,T) = \sum_{p \in T} log(tf(p,f)+1) \cdot log(tf(p,T)+1) \cdot idf(p) \tag{3.16}$$

Soma da semelhança com as outras frases (*sim.to.others*)

Este atributo indica quão semelhante a frase é de todas as outras. São consideradas apenas as semelhanças superiores a um *threshold* de 0.05 (Valizadeh e Brazdil (2014b)).

$$sim.to.others(f) = \sum_{i=1}^{N} sim\,(f, f_i) \tag{3.17}$$

Aqui *N* representa o número total de frases de todos os documentos.

Soma da semelhança com as frases top 5 (*sim.to.top5*)

Este atributo calcula a soma das semelhanças entre a frase e as cinco que lhe são mais semelhantes.

$$sim.to.top5(f) = \sum_{f_i \in Top5} sim\,(f, f_i) \tag{3.18}$$

Frequência de frases (*sent.freq*)

A pontuação *sent.freq* de uma palavra é definida como a razão entre o número de frases em que essa palavra ocorre e o número total de frases.

$$sfw(p) = \frac{\{|f|: p \in f\}}{N} \tag{3.19}$$

A frequência de frases é dada pela média da frequência de frases de cada uma das suas palavras.

$$sent.freq(f) = \frac{\sum_{i=1}^{n} sfw\,(p_i)}{n} \tag{3.20}$$

Aqui *n* indica o número de palavras da frase *f*, e p_i representa cada uma das suas palavras.

Frequência de documentos (*doc.freq*)

De uma forma semelhante ao atributo anterior, mas agora calculado ao nível do documento, surge a frequência de documentos.

O valor *doc.freq* de uma palavra é definido da seguinte forma

$$dfw(p) = \frac{\{|d|: p \in d\}}{|D|} \tag{3.21}$$

sendo que $|D|$ indica o número de documentos.

A frequência de documentos é dada pela média da frequência de documentos de cada uma das suas palavras.

$$doc.freq(f) = \frac{\sum_{i=1}^{n} dfw\,(p_i)}{n} \tag{3.22}$$

Aqui *n* indica o número de palavras da frase *f*, e p_i representa cada uma das suas palavras.

TF*IDF (*sent.tf.idf*)

Considera-se que o TF*IDF de uma frase é a média dos TF*IDF das suas palavras

$$sent.tf.idf(f) = \frac{\sum_{i=1}^{n} TF * IDF(p_i, f)}{n} \tag{3.23}$$

sendo *n* o número de palavras da frase *f*.

Raio da frase (*sent.radius*)

Raio da frase calculado conforme explicado anteriormente (ver secção *Sumarização Baseada na Densidade*).

Score *T-LexRank* da frase (lex.rank)

A ideia é utilizar o valor do *T-LexRank* (Otterbacher *et al.* (2005)) de cada uma das frases como um dos atributos para a aprendizagem supervisionada, como proposto por Valizadeh e Brazdil (2014b).

Atribuição de Pontuação às Frases dos Dados de Treino

Considerando que os sumários de referência (modelos) disponibilizados são sumários de qualidade, a forma que tem sido escolhida para pontuar cada uma das frases baseia-se na relação que cada uma das frases tem com os sumários de referência (Bysani *et al.* (2009), Valizadeh e Brazdil (2014b)). Quanto mais relacionada estiver com os sumários de referência, maior deve ser a sua pontuação (score).

Bysani *et al.* (2009) optaram pela métrica ROUGE-2 para obter este score, enquanto que Valizadeh e Brazdil (2014b) utilizaram o ROUGE-1.

No âmbito desta tese, analisamos as duas alternativas. Tendo observado melhores resultados com a utilização de *bigramas*, adotamos a seguinte expressão para obter o valor (score) de cada frase

$$score(f) = \frac{\sum_{m \in modelos} \left| Bigramas_m \cap Bigramas_f \right|}{n} \qquad (3.24)$$

em que n é o número de palavras da frase f e $\left| Bigramas_m \cap Bigramas_f \right|$ representa o número de *bigramas* partilhados pelo modelo *m* e pela frase *f*.

Exemplo do Conjunto de Dados

A imagem seguinte mostra alguns registos do conjunto de dados criado e que esteve na base dos modelos desenvolvidos com aprendizagem supervisionada.

topico	frases	pos	pos1	pos2	sent.length	sent.length.wsw	topic.rel	sim.to.others	sim.to.top5	topic.sim	sent.freq	doc.freq	sent.tf.idf	sent.radius	lex.rank	score
D0801A-A	The Airbus A380: fron	1	1	0.999	11	6	0.117853569	0.541589261	0.605308102	0.044039563	0.136871508	0.466666667	1.387951553	0.286086474	0.007089877	0.2
D0801A-A	The A380, the new Ai	2	0.952380952	0.998	41	23	0.117853569	1.481551857	0.496705535	0.020698548	0.063881467	0.352173913	1.594631684	0.598200694	0.009108327	0.125
D0801A-A	Here are some key da	3	0.904761905	0.997	35	21	0.057589935	1.02292821	0.621170883	0.009650274	0.034583666	0.252380952	1.763531305	0.615549882	0.005588083	0.029411765
D0801A-A	June 1994: Airbus beg	4	0.857142857	0.004	14	9	0.057589935	0.553396354	0.553396354	0.015448391	0.066418374	0.277777778	1.634710554	0.386724106	0.005858618	0.230769231
D0801A-A	July 2000: Emirates Ai	5	0.80952381	0.005	20	15	0	1.797328334	0.747050871	0	0.031284916	0.3	1.652433387	0.484845628	0.004335885	0.052631579
D0801A-A	December 19, 2000: A	6	0.761904762	0.006	12	8	0.117853569	0.645539465	0.590700713	0.0389155	0.120810056	0.4125	1.375456409	0.320895969	0.006944971	0.363636364
D0801A-A	January 2001: The US	7	0.714285714	0.007	21	13	0.060263634	1.422924057	0.890169566	0.014367214	0.049419854	0.323076923	1.632258155	0.452319357	0.006947452	0.368421053
D0801A-A	February 20, 2001: Air	8	0.666666667	0.008	19	12	0.117853569	0.917513189	0.504321274	0.027980506	0.074487896	0.291666667	1.633679866	0.446209603	0.009213862	0.055555556
D0801A-A	Assembly of the plan	9	0.619047619	0.009	12	5	0	1.832587431	0.78802462	0	0.052513966	0.36	1.428954823	0.248611929	0.005464146	0.181818182
D0801A-A	January 23, 2002: Proc	10	0.571428571	0.01	9	7	0.25513327	0.813999588	0.650285501	0.166808127	0.120510774	0.385714286	1.427760469	0.312683506	0.017419683	0.125
D0801A-A	July 16, 2002: French	11	0.523809524	0.011	21	15	0.060263634	1.153820672	0.724123998	0.013141476	0.046927374	0.313333333	1.65889434	0.494387045	0.008356469	0.2
D0801A-A	June 15, 2003: Emirat	12	0.476190476	0.012	10	6	0.060263634	1.068754384	0.705284098	0.022289542	0.080074488	0.333333333	1.51069928	0.294770344	0.006421794	0
D0801A-A	July 4, 2003: Inaugura	13	0.428571429	0.013	20	10	0.117853569	0.720506919	0.503668575	0.030942163	0.086592179	0.3	1.599382572	0.404275565	0.009335055	0.263157895
D0801A-A	August 19, 2003: The	14	0.380952381	0.014	26	14	0	1.081703738	0.510952021	0	0.027932961	0.264285714	1.723082583	0.490187368	0.00184264	0.12
D0801A-A	March 25, 2004: The fi	15	0.333333333	0.015	25	12	0	1.073095812	0.562488286	0	0.025139665	0.241666667	1.724749624	0.451925907	0.00268422	0.083333333

Figura 3.6 – Conjunto de Dados – Sumarização TAC 2008

3.4.2. Algoritmos de Aprendizagem Supervisionada

Nesta tese exploramos a utilização de três algoritmos de aprendizagem supervisionada, aplicados a um problema de regressão. A opção por três modelos resulta do interesse em querer comparar o desempenho de vários modelos diferentes, para a averiguar qual é o que tem melhor desempenho nas tarefas de sumarização e de *update summarization*.

Escolhemos as *Redes Neuronais* e as *Random Forests*[5] motivados pelos bons resultados destes algoritmos alcançados por Valizadeh e Brazdil (2013) na tarefa de sumarização. O bom desempenho das *Máquinas de Vectores de Suporte* para Regressão[6] (*SVM*) relatado por Bysani *et al.* (2009), na tarefa de *update summarization* levou à decisão de também explorarmos este algoritmo.

[5] Em português, *Florestas Aleatórias*. No âmbito deste documento, optamos pela designação *Random Forests*.

[6] No âmbito deste documento sempre que nos referirmos a máquinas de vetores de suporte (SVM) estamos a considerar a sua aplicação a problemas de regressão.

Os modelos foram treinados usando apenas o conjunto de textos da TAC 2008 para treino e avaliação.

Na avaliação dos modelos foi usada validação cruzada com os textos divididos em 12 conjuntos mutuamente exclusivos, sendo que cada um corresponde a quatro dos 48 tópicos da TAC 2008. Desta forma, cada quatro tópicos são avaliados por um modelo treinado com os textos dos outros 44 tópicos. A métrica que usamos para avaliar os modelos foi a métrica ROUGE-2 aplicada aos sumários criados.

3.4.3. Modelos Múltiplos

Na base da utilização de modelos múltiplos está a ideia de tirar proveito das diferenças subjacentes aos diversos modelos. Ou seja, pretende-se construir um modelo formado por vários modelos básicos que, trabalhando em conjunto, consigam um melhor desempenho do que cada um deles individualmente (Gama *et al.* (2012)).

Investigamos a vantagem de utilizar modelos múltiplos em duas situações distintas:

- combinação de diferentes algoritmos, num Ensemble Heterogéneo;
- combinação de diferentes variantes de um tipo de algoritmo, criando um Ensemble Homogéneo.

Ensemble de Sumarizadores Heterogéneos (*S-EnsHetero*)

Com este modelo pretendemos combinar os algoritmos indicados anteriormente (*Redes Neuronais, Random Forests* e *SVM*), tentando tirar proveito da sua diversidade.

O score final do Ensemble será obtido pela média dos scores dos três algoritmos.

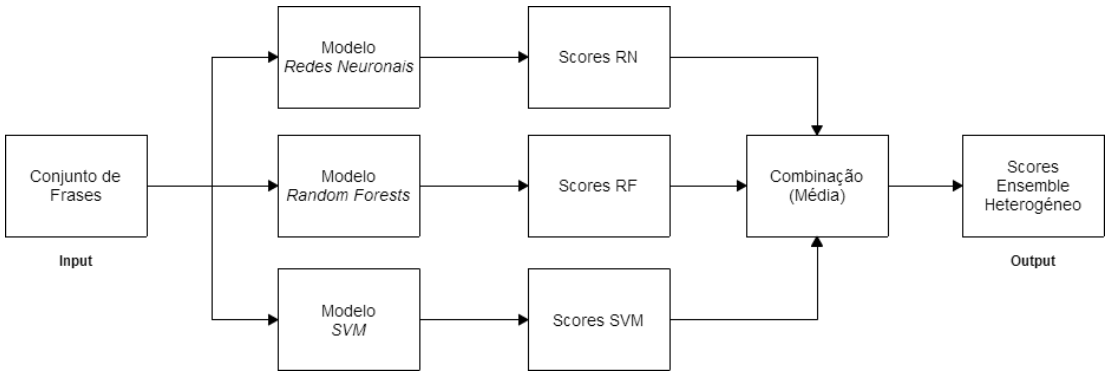


Figura 3.7 – Esquema do Ensemble de Sumarizadores Heterogéneos

Ensemble de Sumarizadores Homogéneos (*S-EnsHomo*)

Perante um mesmo conjunto de textos, diferentes pessoas irão produzir diferentes sumários. Com o Ensemble de Sumarizadores Homogéneos pretendemos criar um modelo que expresse as diferenças de cada um dos sumarizadores humanos que criaram os sumários modelo.

Utilizamos o algoritmo das redes neuronais em todos os modelos.

Perante uma nova frase que pretendemos avaliar, solicitamos a cada um dos modelos que lhe atribua um score. O score final é obtido pela média dos scores individuais. É quase como se pedíssemos a cada um dos sumarizadores humanos que emitisse uma *opinião* (no caso um score) sobre a frase.

Esta ideia baseia-se nos trabalhos Valizadeh e Brazdil (Valizadeh e Brazdil (2014b) e Valizadeh e Brazdil (2014a)) que apresentam resultados bastante bons na tarefa de sumarização.

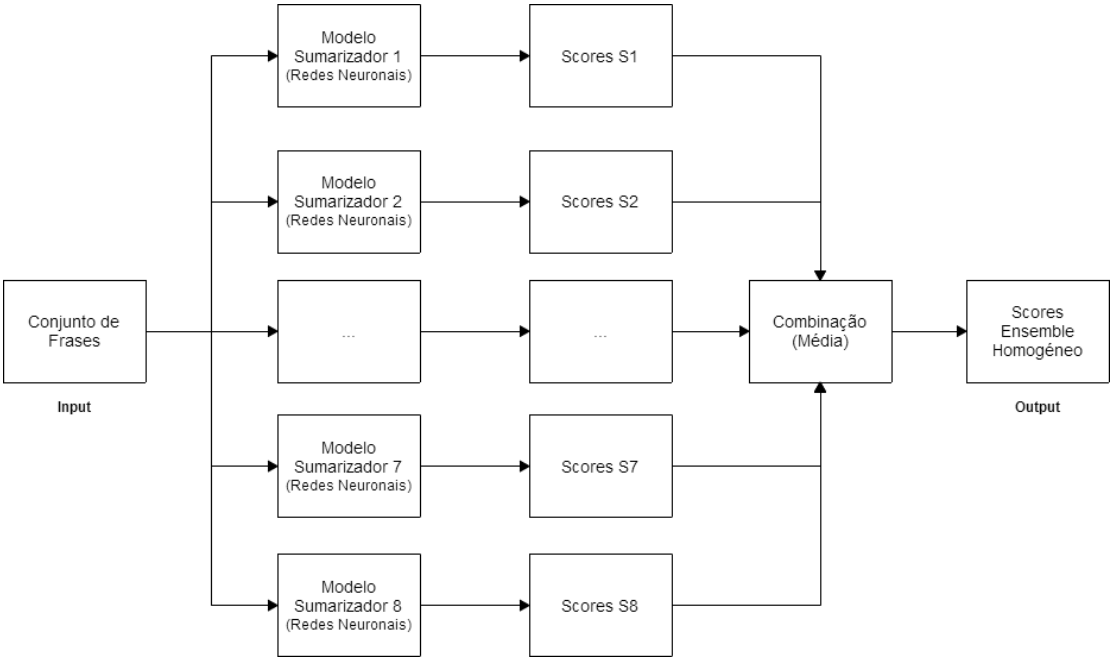


Figura 3.8 – Esquema do Ensemble de Sumarizadores Homogéneos

Cada um dos vários modelos utiliza um conjunto de treino diferente, sendo que o score atribuído a cada uma das frases é calculado apenas com base no sumário desse sumarizador humano. O conjunto de treino é constituído apenas por metade dos textos, uma vez que cada humano apenas avalia metade dos tópicos.

Os dados usados no treino dos modelos foram obtidos a partir dos textos da TAC 2008 e na fase de avaliação apenas foram considerados as frases que não fizeram

parte do seu treino, evitando-se assim que os modelos sejam treinados e avaliados com os mesmos exemplos. Assim, na TAC 2008 o score de cada frase é dado pela média de quatro modelos, já que os outros quatro foram treinados com essa frase.

No caso da DUC 2007 e TAC 2009, como não tiveram qualquer intervenção na fase de treino, o score é a média dos resultados dos oito modelos, como se pode observar na figura anterior.

3.5. *Update Summarization* com Aprendizagem Não Supervisionada

Na tarefa de *update summarization* exploramos, em primeiro lugar, a abordagem não supervisionada baseada em grafos. Foram utilizados os algoritmos considerados e implementados na tarefa de sumarização e outros desenvolvidos a pensar no *update summarization*.

3.5.1. Utilização de um *threshold* para evitar redundância com histórico

A grande diferença da tarefa de sumarização para a tarefa de *update summarization* reside essencialmente na deteção de alguma novidade que está inerente a esta tarefa.

Nota-se que os algoritmos baseados em aprendizagem não supervisionada explorados na tarefa de sumarização (*T-LexRank* e *DensityBased*) não têm isto em consideração.

Decidimos experimentar uma abordagem simples para lidar com este problema, estabelecendo um *threshold* máximo que poderá ocorrer entre as frases dos textos atuais, que pretendemos sumarizar, e as frases do histórico. A ideia é que se uma frase ultrapassa esse *threshold*, então é muito semelhante ao que já é conhecido do leitor, logo não deve fazer parte do *update summary*.

É importante salientar que isto não exige a alteração do processo iterativo de atribuição de scores às frases. Sendo um processo que se realiza posteriormente.

A dificuldade passa apenas pela determinação do valor do *threshold*. O processo que seguimos foi o da experimentação do algoritmo com diversos valores e selecionando o que permitiu alcançar melhores resultados com a métrica ROUGE-2.

3.5.2. *Update Summarization* com Reforço Positivo e Negativo (*PNR²*)

O PNR2, já referido no capítulo 2, foi proposto por Li *et al.* (2008).

A motivação para investigar o *PNR²* deve-se a ser um algoritmo pensado de raiz para ser utilizado na tarefa de *update summarization*, já que os scores atribuídos às frases já têm em consideração o relacionamento destas com o histórico. Contudo, o *PNR²* pode perfeitamente ser usado em sumarização.

Os algoritmos analisados anteriormente (*T-LexRank* e *DensityBased*), apenas usavam as ligações entre as frases como reforço positivo. A grande inovação introduzida pelo *PNR²* é considerar dois tipos de reforço, no grafo: o reforço positivo é atribuído a duas frases relacionadas pertencentes à mesma coleção de textos (textos atuais ou textos de histórico); o reforço negativo permite penalizar as ligações entre frases de coleções diferentes (sendo uma do conjunto de textos atual e outra do histórico).

Apresentamos agora o método iterativo que é usado no *PNR²* para atribuição de scores às frases, conforme apresentado por Li *et al.* (2008).

Consideremos que R_A e R_B representam o score da frase no conjunto de textos atual e de textos do histórico, respetivamente. Consequentemente, estes scores são calculados pela expressão

$$\begin{cases} R_A^{(k+1)} = \alpha_1 \cdot M_{AA} \cdot R_A^{(k)} + \beta_1 \cdot M_{AB} \cdot R_B^{(k)} + \gamma_1 \cdot \vec{p}_A \\ R_B^{(k+1)} = \beta_2 \cdot M_{BA} \cdot R_A^{(k)} + \alpha_2 \cdot M_{BB} \cdot R_B^{(k)} + \gamma_2 \cdot \vec{p}_B \end{cases} \quad (3.25)$$

em que M_{AA}, M_{BB}, M_{AB} e M_{BA} são as matrizes de similaridades das frases de A, das frases de B, das frases de A com B e das frases de B com A, respetivamente. Os pesos são atribuídos pela matriz

$$W = \begin{bmatrix} \alpha_1 & \beta_1 \\ \beta_2 & \alpha_2 \end{bmatrix}$$

Considerando que α_1e α_2 representam o reforço positivo, respetivamente de M_{AA} e de M_{BB}.

β_1 e β_2 deverão ser inferiores a 0, de forma a expressarem o reforço negativo, a atribuir a M_{AB} e a M_{BA}.

γ_1 e γ_2 são dois *damping factors* (similares ao parâmetro *d* do *T-LexRank*).

Os vetores $\vec{p}_A$ e $\vec{p}_B$ representam, no caso da sumarização baseada em tópicos, o relacionamento das frases com os tópicos, ou seja,

$$\vec{p}_i = sim(f_{i,}, T)$$

A implementação deste método seguiu o algoritmo disponibilizado por Li *et al.* (2008).

3.5.3. *Update Summarization* Baseado em Grafo e Reordenação (*T-LexReRank*)

O *T-LexRank* mostra um bom desempenho na tarefa de sumarização, mas não incorpora nenhuma característica específica para *update summarization*. Conforme vimos atrás, a solução encontrada para adequar o *T-LexRank* ao *update summarization* foi recorrer a um *threshold* limite na semelhança entre as frases dos textos atuais e as dos textos de histórico. Quisemos então averiguar se não seria possível abordar o problema de uma forma alternativa.

Inspirados no MMR (Carbonell e Goldstein (1998)), adotamos uma solução simples que reordena os rankings de forma a penalizar proporcionalmente as frases, de acordo com a sua semelhança máxima em relação às frases de histórico.

Chamamos a esta solução *ReRanker for Update Summarization* (*T-LexReRank*), sendo o score atribuído pela expressão

$$\mathrm{R}_{\mathrm{RR}}(f) = \mathrm{R}_{\mathrm{LR}}(f) \cdot (1 - \max(sim(f, f_h))) \tag{3.26}$$

Em que $\mathrm{R}_{\mathrm{LR}}(f)$ indica o score do *T-LexRank* da frase *f*, f_h representa cada uma das frases do histórico e *sim* indica a medida de semelhança entre as frases (conforme referido anteriormente, a similaridade do cosseno).

É usado um *threshold* de 0.05, abaixo do qual a semelhança entre as frases é considerada inexistente.

O *T-LexReRank* será zero quando a frase é igual a uma frase existente no histórico e será igual ao valor do *T-LexRank* (portanto, sem penalização) no caso de não haver semelhança com qualquer frase anteriormente conhecida).

Quando não há textos de histórico, como acontece na tarefa de sumarização, o *T-LexReRank* é coincidente com o *T-LexRank*.

Importa reforçar que com esta implementação deixa de existir o *threshold* anteriormente considerado. O ranking obtido com o *T-LexReRank* será usado para a elaboração do sumário, sem haver nenhuma verificação adicional com os textos de histórico.

3.6. Métodos Supervisionados para *Update Summarization*

Nesta tarefa foram explorados e implementados os mesmos algoritmos baseados em aprendizagem supervisionada, analisados no âmbito da sumarização.

Uma vez que a tarefa de *update summarization* exige mais informação que a necessária para a sumarização, decidimos criar um novo conjunto de dados.

Os modelos foram treinados com o novo conjunto de dados e todo o processo decorreu conforme explicado anteriormente para a tarefa de sumarização com aprendizagem supervisionada.

3.6.1. Criação do Conjunto de Dados

Para além dos atributos criados anteriormente para a tarefa de sumarização, é necessário considerar outros que permitam aferir a novidade em relação ao que já é conhecido, para ser possível proceder ao *update summarization*. Este conjunto de dados terá os atributos anteriores e os que são agora apresentados.

Atributos de Entrada para *Update Summarization*

Fator de novidade (*novelty.factor*)

Este atributo foi proposto por Bysani *et al.* (2009). A ideia é medir a relevância e novidade de uma frase. O fator de novidade de uma palavra é calculado através da expressão

$$fn(p) = \frac{\{|d|: p \in D_A\}}{\{|d|: p \in D_H\} + |D|} \tag{3.27}$$

em que D_A indica o conjunto de documentos atuais, enquanto D_H representa o conjunto de documentos de histórico. Basicamente no numerador temos o número de textos atuais que contêm a palavra p, enquanto o denominador é a soma do número de textos de histórico que contém a palavra p com o número total de textos.

O fator de novidade de uma frase é dado pela média do fator de novidade de cada uma das suas palavras

$$novelty.factor(f) = \frac{\sum_{i=1}^{n} fn\,(p_i)}{n} \tag{3.28}$$

sendo que *n* indica o número de palavras da frase *f*, e p_i representa cada uma das suas palavras.

Medida de novidade e relevância para o tópico (*novel.relevant*)

Aproveitando a medida anterior, Kogilavani e Balasubramanie (2012) associam ao fator de novidade, uma componente de relevância com o tópico

$$novel.relevant\ (f,T) = \frac{novel.factor(f) + topic.sim(f,T)}{n} \tag{3.29}$$

em que n é o número de palavras da frase f e T identifica o tópico em análise.

Semelhança com o histórico (*sum.sim.history, max.sim.history, perc.sim.history*)

Para aferir a novidade de uma frase, em relação às frases que já são conhecidas do histórico, tendo em consideração a semelhança do cosseno, são introduzidos três novos atributos. Tanto quanto sabemos, estes três atributos não foram anteriormente explorados no âmbito da aprendizagem automática de um modelo para *update summarization*.

- Soma das semelhanças com o histórico

$$sum.sim.history\ (f) = \sum_{h=1}^{N_H} sim\ (f, f_h) \tag{3.30}$$

  em que N_H é o número de frases do histórico, f é uma frase do conjunto de textos atual e f_h representa uma frase do histórico.

- Maior semelhança com as frases do histórico

$$max.sim.history\ (f) = \max(sim(f, f_h)) \tag{3.31}$$

  em que f é uma frase do conjunto de textos atual e f_h representa cada uma das frases do histórico.

- Percentagem das semelhanças com o histórico – tendo em consideração a soma de todas as semelhanças da frase, indica qual a porção que se refere à semelhança com o histórico.

$$perc.sim.historical\ (f) = \frac{sum.sim.history\ (f)}{sim.to.others(f)} \tag{3.32}$$

Nos três atributos é usado um *threshold* mínimo de 0.05, para evitar as ligações devidas ao acaso.

Exemplo do Conjunto de Dados

A imagem seguinte mostra alguns registos do conjunto de dados criado.

topico	frases	pos	pos1	pos2	sent.len	sent.length.ws	topic.re	sim.to.other	sim.to.top5	topic.sim	sent.freq	doc.freq	sent.tf.idf	sent.radius	lex.rank	novelty.fa	novel.relev	sum.sim.hist	max.sim.his	perc.sim.hi	score
D0901A-B	Kashmir bus rol	1	1	0.999	12	9	0.1315	0.57822008	0.5190767	0.036355995	0.0670194	0.388889	1.766206	0.4151697	0.007095	0.2559319	0.29228786	0.386111624	0.13028128	0.400393	0
D0901A-B	The first trans-I	2	0.96	0.998	39	23	0.0973	1.17220138	0.5415977	0.011749306	0.0510697	0.343478	1.749133	0.6387638	0.006072	0.2392076	0.25095694	1.690584957	0.20762545	0.5905383	0.2632
D0901A-B	"We basically s	3	0.93	0.997	21	13	0.2053	0.63877144	0.4844691	0.100888799	0.044363	0.284615	1.878098	0.5114772	0.012018	0.2074541	0.30834288	0.858782922	0.12457252	0.5734569	0
D0901A-B	"We must unde	4	0.89	0.004	19	10	0.053	0.57817335	0.4117957	0.009159476	0.0312169	0.24	2.044407	0.4876114	0.00494	0.1770779	0.1862374	0.465535732	0.13309294	0.4460397	0
D0901A-B	The first bus se	5	0.86	0.005	29	18	0.126	2.04211809	0.8066777	0.020496589	0.0749559	0.511111	1.660034	0.5380164	0.00779	0.3257866	0.34628317	3.248152454	0.32996391	0.6139861	0.037
D0901A-B	Passengers cro	6	0.82	0.006	42	23	0.1001	0.62215917	0.4609429	0.011632024	0.0331263	0.26087	1.895792	0.6896803	0.006349	0.1868754	0.19850747	2.21929681	0.41587754	0.7810421	0.0732
D0901A-B	At the end of th	7	0.79	0.007	29	12	0	0.45745233	0.3525512	0	0.0145503	0.183333	1.996305	0.5128968	0.000126	0.138961	0.13896104	1.069248842	0.13325038	0.7003655	0.0769
D0901A-B	They were gree	8	0.75	0.008	32	14	0	0.45174346	0.3998206	0	0.0151172	0.185714	1.970829	0.5467305	0.001227	0.1389547	0.13895474	1.289438606	0.12759674	0.7405536	0.0968
D0901A-B	"All these mea	9	0.71	0.009	17	10	0.053	0.75475432	0.6253267	0.009956341	0.0645503	0.3	1.829488	0.4471868	0.007321	0.1917532	0.20170959	0.533897858	0.10563338	0.4143072	0
D0901A-B	"And President	10	0.68	0.01	22	12	0.1998	1.91174714	0.6901231	0.087915505	0.053351	0.4	1.70271	0.4469776	0.009749	0.2715465	0.35946201	1.734403165	0.31274181	0.4756807	0
D0901A-B	Indian and Paki	11	0.64	0.011	20	13	0.3054	0.63309891	0.5300913	0.114679101	0.046805	0.315385	1.886274	0.521312	0.012996	0.2203578	0.33503695	0.664760371	0.08633427	0.5121976	0.1579
D0901A-B	Nearly 1,000 co	12	0.61	0.012	8	5	0	0.68912763	0.4785202	0	0.0169312	0.22	2.101065	0.3495296	0.001632	0.1969231	0.19692308	0.291742405	0.10117934	0.2974323	0
D0901A-B	I.A. Rehman, di	13	0.57	0.013	16	11	0.053	0.43646853	0.4364685	0.008472227	0.027417	0.190909	2.122215	0.5270131	0.003615	0.1461433	0.15461548	0.246104234	0.0940794	0.3605538	0
D0901A-B	"It was good th	14	0.54	0.014	28	14	0	0.91641264	0.542897	0	0.0226757	0.257143	1.950113	0.5409379	0.002788	0.1944327	0.19443265	0.974349666	0.17893203	0.5153211	0.0385

Figura 3.9 – Conjunto de Dados – *Update Summarization* TAC 2009

3.7. Seleção de Atributos na Aprendizagem Supervisionada

Tendo por base os conjuntos de dados (ou tabelas) criados para a sumarização e *update summarization*, passamos à verificação se seria ou não necessário reduzir a dimensionalidade, isto é, analisar se selecionando apenas um subconjunto dos atributos disponíveis obtínhamos um melhor desempenho que o conseguido com a utilização de todos.

Para isto implementámos um processo iterativo de eliminação para trás (*backward elimination*), começando com todos os atributos e, em cada iteração eliminando o atributo que, sendo removido, melhor resultado permitia alcançar em termos de métricas ROUGE. Neste processo seguimos a estratégia de ir eliminando atributos até numa dada iteração a remoção de atributos não trazer qualquer ganho. Como não foi implementado um método de procura completa, não garantimos que tenhamos selecionado o melhor subconjunto de atributos, mas consideramos que a melhoria (entre a utilização de todos os atributos e apenas os selecionados) é relevante.

Na seleção de atributos tivemos em conta o resultado com a utilização dos três algoritmos diferentes: *Redes Neuronais*, *SVM* e *Random Forests*. A ideia foi avaliar o desempenho nos três preditores para não selecionar um subconjunto de atributos muito bom para um deles, mas que tenha péssimos resultados com os outros dois. Foi necessário haver uma *negociação* entre os resultados dos vários modelos, de forma a fazer melhor escolha, já que a seleção de um atributo nem sempre originou em melhor resultado para os três regressores.

A tabela seguinte mostra os atributos selecionados, assim como a variação na métrica ROUGE-2, nos dois conjuntos de dados (sumarização e *update summarization*).

		ROUGE-2			ROUGE-4		
	Atributos Retirados	**RNA**	**RF**	**SVM**	**RNA**	**RF**	**SVM**
Sumarização	Nenhum	0.0943	0.0898	0.0935	0.1279	0.1219	0.1257
	Pos1; sim.to.top5	0.0962	0.0964	0.0932	0.1291	0.1297	0.1254
Δ %		1.96%	7.38%	-0.37%	0.93%	6.35%	-0.25%
Update Summarization	Nenhum	0.0936	0.0955	0.0893	0.1281	0.1309	0.1273
	Pos1; sim.to.top5; novelty.factor; novel.relevant; perc.sim.history	0.0936	0.0946	0.0906	0.1291	0.1304	0.1275
Δ %		0.04%	-0.88%	1.51%	0.77%	-0.33%	0.17%

Tabela 3.1 – Variação das métricas ROUGE com a seleção de atributos

A decisão de optar apenas por apenas dois atributos na tarefa de *update summarization*, não foi fácil. Mas a utilização de todos os atributos penalizava o desempenho dos modelos e isto ocorreu com os três, indiciando que realmente os atributos não deviam ser incluídos na aprendizagem automática. Consideramos que os atributos eliminados provavelmente seriam redundantes com os outros já existentes no conjunto de dados. Esta hipótese requer uma investigação mais aprofundada a realizar no futuro.

3.8. Seleção das Frases e Criação do Sumário

Num processo de sumarização como que o foi implementado, a tarefa central é a atribuição de um score (ou pontuação) a cada frase, sendo com base na ordenação desses scores que o sumário será criado.

Contudo, a criação do sumário não é imediata, isto é, não é suficiente selecionar as frases melhor classificadas e inseri-las no sumário até se atingir o limite de palavras do sumário. Por outro lado é também necessário decidir como serão colocadas as frases selecionadas no sumário.

Estes aspetos serão abordados a seguir.

3.8.1. Seleção das Frases

Geralmente o sumário terá de obedecer a um tamanho máximo, quer seja em bytes quer seja no número de palavras. No âmbito das conferências DUC 2007, TAC 2008 e TAC 2009, os sumários foram limitados a um máximo de 100 palavras.

Outro aspeto muito importante a ter em consideração é que num sumário é essencial evitar a redundância. Se uma frase tiver um bom score, mas se for muito semelhante a uma que já tenha sido selecionada, em princípio, será melhor não selecionar essa frase, escolhendo outra que, apesar de ter um score mais baixo, traga algo de novo ao sumário.

Para evitar a redundância, seguimos uma estratégia inspirada no MMR (Carbonell e Goldstein (1998)). Assim, não serão selecionadas frases que sejam muito semelhantes (de acordo com um *threshold*) às que já fazem parte do sumário.

Por fim, tivemos também em consideração o número de palavras da frase, evitando extrair frases muito curtas, tidas como não muito informativas (Erkan e Radev (2004), Patil e Brazdil (2007)).

3.8.2. Criação do Sumário

Tendo por base a lista de frases que irão ser incluídas no sumário, tem de se estabelecer a ordem pela qual irão aparecer. Sabe-se que para *n* frases, é possível organizá-las de *n!* maneiras possíveis. Em termos de avaliação ROUGE o resultado será o mesmo. Contudo, do ponto de vista linguístico e de coerência textual esses sumários serão bastante distintos.

Desde o início da sumarização automática de um único documento que vem sendo aplicada uma heurística mais ou menos consensual que consiste em colocar as frases pela ordem em que elas surgem no texto. Esta regra passou depois para a sumarização multidocumento. Sendo que neste caso já é discutível se é ou não uma boa prática.

O algoritmo que implementamos tem em consideração a data do texto, sendo que os textos são tratados cronologicamente. Assim, as frases são colocadas pela ordem cronológica dos textos de que foram extraídas. Havendo mais do que uma frase do mesmo texto, serão colocadas pela ordem em que surgem no texto.

A ideia para esta decisão resulta do interesse em ter uma coerência temporal no sumário. Também estudamos a hipótese de colocar as frases pela ordem cronológica inversa: isto é, pondo no início as frases dos textos mais recentes. Esta alternativa também é lógica, até porque estando a trabalhar com *update summarization*, os textos mais recentes serão, em princípio, os que estão mais atualizados. Contudo, optamos por manter a primeira ideia, porque pensamos que assim o sumário poderá permitir observar a evolução de como as novidades foram surgindo.

Temos de salientar que não foi feito um estudo comparativo entre as diversas formas de arranjar as frases, a fim de perceber quais as que dão melhor resultado em termos de leitura. Isto terá de ser efetuado no futuro.

Neste ponto, sabendo a lista de frases selecionadas e a ordem pela qual elas deverão aparecer, a criação do sumário é uma tarefa trivial.

4. Análise dos Resultados

Neste capítulo apresentamos os conjuntos de textos utilizados e os sistemas de comparação. Fazemos também referência à parametrização do ambiente de implementação e avaliação, indicando os valores dos parâmetros dos diversos algoritmos. Posteriormente são apresentados e analisados os resultados obtidos.

4.1. Textos Utilizados

Foram utilizados os textos disponibilizados nas conferências DUC 2007[1], TAC 2008[2] e TAC 2009[3] para as tarefas de *update summarization*.

O conjunto de textos da DUC 2007 está dividido em dez tópicos, cada um com cerca de 25 documentos. Cada tópico está dividido em três grupos, A, B e C, sendo que, cronologicamente, A é anterior a B, que por sua vez antecede C.

Pretende-se a criação de três sumários bem organizados, fluentes e pequenos (com limite de 100 palavras). Os sumários são do tipo multidocumento e orientados para um tópico ou questão (*query-based*). As três tarefas consistem na criação de: sumário dos textos contidoS em A; sumário dos textos contidos em B, assumindo que o leitor já conhece os textos de A; sumário dos textos de C, assumindo que já são conhecidos os textos de A e B.

A TAC 2008 e a TAC 2009 são semelhantes à DUC 2007, sendo que a tarefa consiste em produzir um sumário semelhante ao da DUC 2007, a partir de um conjunto de notícias, assumindo que o utilizador já leu um conjunto de notícias anteriores.

O sumário pretendido também é orientado para um dado tópico, pelo que juntamente com os textos das notícias é também fornecida informação sobre o que o leitor procura. Ou seja, são disponibilizados dois conjuntos de notícias (documentos antigos e novos) e o *tópico* em análise.

O conjunto de textos da TAC 2008 é constituído por 48 tópicos. Cada tópico tem um *título*, uma *descrição* e um conjunto de 20 documentos, divididos em dois

[1] http://duc.nist.gov/duc2007/tasks.html#pilot
[2] http://www.nist.gov/tac/2008/summarization/update.summ.08.guidelines.html
[3] http://www.nist.gov/tac/2009/Summarization/update.summ.09.guidelines.html

conjuntos: conjunto A e conjunto B (sendo que o conjunto A precede cronologicamente o conjunto B).

A figura seguinte mostra um texto do conjunto da TAC 2008.

```
<DOC>
<DOCNO> XIE19981113.0066 </DOCNO>
<DATE_TIME> 1998-11-13 </DATE_TIME>
<BODY>
<HEADLINE> Euro May Pose Strategical and Tactical Problems </HEADLINE>
<TEXT>
<P>
BUCHAREST, November 12 (Xinhua) -- The introduction of the single European currency, the Euro, would pose strategical as well as tactical problems to the economy of Romania, an associate EU country seeking admission, the Rompres news agency Thursday quoted Romania's Central Bank Governor Mugur Isarescu as saying.
</P>
<P>
Romanian bankers and economists could not be indifferent to the future of Euro in the international payment system and as a reserve currency, the evolution of the Euro-Dollar exchange rate, the relations which will be established between the national currency Leu and Euro, or Euro's effect on Romanian commercial and payments balance, Isarescu said at a Wednesday colloquium on Euro
by the Romanian Society of Economics.
</P>
<P>
The answers to these matters represent the precondition for identifying the monetary and currency policies which might ensure, in the shortest delay, the criteria for a convergence between the Romanian economic and monetary system, on the one hand, and the Economic and monetary Union, on the other hand, the Governor added.
</P>
</TEXT>
</BODY>
</DOC>
```

Figura 4.1 – Exemplo de um documento da TAC 2008

Para a avaliação, cada tópico tem 4 sumários modelo ou de referência (produzidos por pessoas selecionadas pelos organizadores do evento). Estes sumários também são designados de *modelos* ou *golden standard*.

O conjunto de textos da TAC 2009 é em tudo idêntico ao da TAC 2008, com a diferença de que apenas tem 44 tópicos.

4.2. Sistemas de Comparação

Para ser possível uma mais correta interpretação dos resultados obtidos com os diversos sistemas, é necessário ter alguns pontos de referência, quer sejam referência base que determinam um limite inferior (*baselines*) ou um teto máximo ou limite superior (*upper bound*).

A TAC 2008 disponibiliza o *Lead* (Sistema 0) que é um sistema *baseline* que consiste na seleção das primeiras frases do texto mais recente no conjunto de documentos em análise, até ser atingido o limite de 100 palavras (Dang e Owczarzak (2008)).

Na TAC 2009 são disponibilizados três sistemas de comparação (Dang e Owczarzak (2009)):

- *Lead* (Sistema 1) – idêntico ao *Lead* da TAC 2008, ou seja, as frases do início do texto mais recente, até um limite de 100 palavras;
- *Model* (Sistema 2) – um dos sumários de modelo (criado por um dos sumarizadores humanos), mas com as frases ordenadas aleatoriamente. Importa referir que este sistema não deve ser utilizado para comparação dos valores ROUGE, já que o sumário irá ser comparado com ele próprio, o que vai inflacionar e desvirtuar os valores. Devido a isto, como se poderá confirmar nas tabelas em anexo, este sistema atinge valores significativamente superiores a qualquer um dos outros.
- *Manual (Sistema 3)* – sumário extrativo, composto por frases selecionadas manualmente por elementos da Universidade de Montereal.

Com estes três sistemas temos uma *baseline* (*Lead*), um *upper bound* (*Model*) e um sistema extrativo manual (onde um humano executa um processo semelhante à generalidade dos sistemas de sumarização automática, em que a tarefa principal é a identificação das frases mais relevantes a incluir no sumário).

4.2.1. Sistemas de Comparação Criados

Para uma melhor avaliação dos três conjuntos de documentos (DUC 2007, TAC 2008 e TAC 2009), e uma vez que os sistemas de comparação disponibilizados não são iguais em todos, decidimos incluir mais alguns sistemas de comparação que apresentamos de seguida.

Aleatório

A seleção das frases é feita de forma aleatória. Os valores das métricas ROUGE são calculados com base na média dos resultados de 10 sumários aleatórios gerados para cada um dos tópicos.

Topo

São selecionadas as frases do início do texto, até um limite de 100 palavras. Ao contrário do *Lead* da TAC 2008 e da TAC 2009, aqui é considerada a média dos

resultados de cinco sumários. A ideia é perceber se a data do texto é ou não importante para este tipo de sumário.

Oráculo 1

Sumário gerado selecionando as frases que têm maior número de termos coincidentes com os sumários modelo (elaborados por humanos).

Oráculo 2

Semelhante ao anterior, mas em vez de palavras analisa-se a sobreposição de bigramas, sendo selecionadas as frases que partilham mais bigramas comuns com os sumários de referência.

Humano

Com este sistema analisamos os resultados dos sumários modelo, elaborados por humanos, quando avaliados em comparação com os modelos dos outros humanos.

Ao contrário do sistema *Model* da TAC 2009, o sistema *Humano* é avaliado de forma a que nunca seja comparado consigo próprio, o que permite a sua comparação com os outros sistemas.

4.2.2. Resultados e Comparação

Analisando os resultados de todos os sistemas de comparação (as tabelas ROUGE estão disponíveis em anexo, onde os sistemas de comparação surgem em itálico) verificamos que o patamar superior da sumarização extrativa é dado pelos sistemas *Oráculo 1* e *Oráculo 2*, sendo que este último é o que obtém melhores resultados nas métricas ROUGE.

O sistema *Humano* obtém melhores resultados que os sistemas de sumarização automática, salvo na tarefa de sumarização da DUC 2007, em que há um sistema que obtém resultados melhores. Contudo, é interessante verificar que o sistema *Humanos* fica bastante aquém dos resultados dos sistemas *Oráculo 1* e *Oráculo 2*, o que vem confirmar que pessoas diferentes fazem sumários substancialmente diversos.

Quanto a *baselines* verifica-se que o sistema *Aleatório* fica sempre nos últimos lugares. É contudo de estranhar que haja vários sistemas que não consigam sequer obter os resultados desta *baseline*.

Se considerarmos o sistema de comparação *Topo* vemos que ainda fica à frente de mais sistemas.

Analisando estes sistemas ficamos cientes de até onde é possível ir com um sistema de sumarização meramente extrativo, e sabemos qual é o valor mínimo que o sistema tem de alcançar para lhe ser reconhecido algum mérito.

4.3. Parametrizações e Escolha de Opções nas Experiências

O trabalho foi desenvolvido na plataforma R[4] (versão 3.02) (R Core Team (2014)).

Para a avaliação automática dos sumários utilizamos as métricas ROUGE (Lin (2012)), já referidas no capítulo 2. A versão utilizada foi o ROUGE-1.5.5[5].

Os parâmetros ROUGE utilizados coincidem com os parâmetros utilizados na TAC 2008: -e data -c 95 -2 -1 -U -r 1000 -x -m -2 4 -u -n 2 -f A -p 0.5 -t 0 -d -a.

Nos resultados analisamos as métricas ROUGE-2 e ROUGE-SU4 e, salvo indicação do contrário, os valores dizem respeito ao *recall*[6] médio.

De seguida apresentamos os parâmetros finais que implementamos nos diversos sistemas. Para chegarmos a estes valores foi necessário efetuar um processo de experimentação de diferentes combinações, no sentido de maximizar o resultado obtido com as métricas ROUGE.

Sempre que a alteração dos parâmetros não trazia melhoria evidente ou quando surgiram dúvidas sobre as melhores alternativas, optamos por manter os valores sugeridos pelos autores dos algoritmos.

T-LexRank e *DensityBased*

Para determinar qual o melhor valor para parâmetro *damping factor (d)* foram realizadas diversas experiências usando os textos da TAC 2008. O gráfico seguinte mostra os valores da medida ROUGE-2.

[4] Disponível para download em http://www.r-project.org/.
[5] Disponível em http://www.berouge.com/Pages/default.aspx.
[6] O *recall* indica a taxa de *n-gramas* dos sumários de referência que são incluídos no sumário que se está a avaliar.

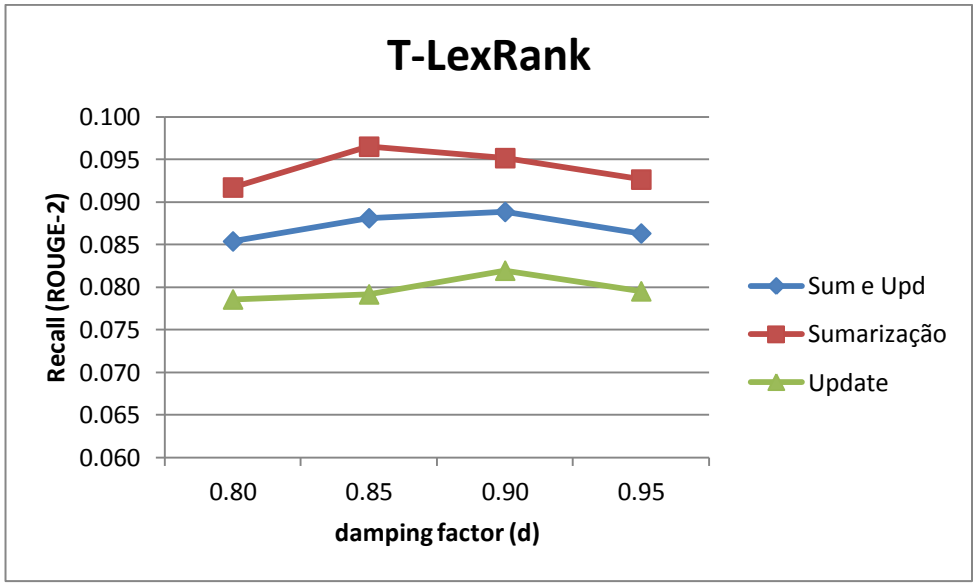


Gráfico 4.1 – Variação do ROUGE-2 usando diferentes valores do *damping factor* para o algoritmo *T-LexRank*

Foi escolhido para *damping factor* o valor de 0.90, porque é o que permite obter um valor mais elevado na tarefa de *update summarization* e consequentemente no desempenho global.

A tabela seguinte apresenta os parâmetros utilizados nos algoritmos *T-LexRank* e *DensityBased*.

Parâmetro	Valor
damping factor (d)	0.90
ε	0.0001
Nº máximo de iterações	1000

Tabela 4.1 – Parâmetros do *T-LexRank* e *DensityBased*

Redes Neuronais (*Sup-RN*)

Utilizamos uma *Multilayer Feed Forward Neural Network*, implementada no *package* AMORE (Limas *et al.* (2014)), com os seguintes parâmetros:

Parâmetro	Valor
Hidden Layers	2
Neurons per Hidden Layer	10
Learning Rate	0.01
Momentum	0.5
Error Criterium	LMS - Least Mean Squares
Hidden Layer Activation Function	Sigmoid
Output Layer Activation Function	Purelin
Prefered Training Method	Adaptative gradient descend with momentum

Tabela 4.2 – Parâmetros das *Redes Neuronais*

Random Forests (*Sup-RF*)

Foi utilizado o package *randomForest* (Liaw e Wiener (2014)), com os seguintes parâmetros:

Parâmetro	Valor
Type	regression
Number of trees	250
Number of variables tried at each split	3

Tabela 4.3 – Parâmetros das *Random Forests*

SVM (*Sup-SVM*)

Foi utilizada a função *svm* do package e1071 (Meyer *et al.* (2014)), com os seguintes parâmetros:

Parâmetro	Valor
Type	nu-regression
Kernel	radial basis
Cost	1
Gamma	0.833
nu	0.2

Tabela 4.4 – Parâmetros do *SVM*

$\mathbf{PNR^2}$

A implementação seguiu o algoritmo dos autores (Li *et al.* (2008)), utilizando os parâmetros:

Parâmetro	Valor
W	$\begin{bmatrix} 1 & -0.5 \\ -0.5 & 1 \end{bmatrix}$
γ_1 e γ_2	0.5
θ	0.5

Tabela 4.5 – Parâmetros do PNR^2

Criação do Sumário

Para evitar a redundância, uma frase apenas será selecionada, se não existir nenhuma no sumário que seja semelhante em mais de 0.7, na medida do cosseno.

Depois de analisarmos diversos valores, estabelecemos que uma frase apenas seria selecionada se tiver pelo menos 6 palavras.

No *update summarization* o *threshold* de similaridade com o histórico foi fixado nos 0.4.

Apresentamos agora os resultados que levaram à decisão de optar pelo TF*ISF e à utilização de todas as frases nos grafos da tarefa de *update summarization.*

Comparação dos resultados usando TF*IDF vs. TF*ISF

A escolha da forma como se quantifica o peso de cada uma das palavras no modelo de espaço vetorial tem uma influência significativa no desempenho do *T-LexRank*, assim como dos outros algoritmos.

Isto pode ser observado na tabela seguinte, onde são comparados os resultados do *T-LexRank* obtidos nas métricas ROUGE-2 e ROUGE-SU4, com o TF*IDF e o TF*ISF, usando os textos da TAC 2008.

	Global		Sumarização		*Update Summarization*	
	ROUGE-2	ROUGE-SU4	ROUGE-2	ROUGE-SU4	ROUGE-2	ROUGE-SU4
TF*IDF	0.05109	0.09230	0.04995	0.09080	0.05229	0.09377
TF*ISF	0.08589	0.12231	0.09177	0.12560	0.07952	0.11855
Δ %	68%	33%	84%	38%	52%	26%

Tabela 4.6 – Comparação do recall obtido usando TF*IDF vs. TF*ISF

Os resultados das métricas ROUGE-2 e ROUGE-SU4 são superiores quando se utiliza o TF*ISF. Isto levou a que tivéssemos adotado o TF*ISF.

Frases a considerar na construção do grafo

É importante decidir quais as frases que devem ser inseridas na construção do grafo. Num processo de simples sumarização esta questão é irrelevante, já que são consideradas todas as frases. Contudo, quando se trata de *update summarization* importa ponderar se devem, ou não, ser incluídas também as frases do histórico ou apenas as frases dos textos novos.

Para analisar esta questão, com os textos da TAC 2008 e utilizando o *T-LexRank*, geramos os sumários com as frases do conjunto A (textos do histórico) e B (textos novos) e apenas com as dos textos novos (B) e fizemos a avaliação usando as métricas ROUGE-2 e ROUGE-SU4. A tabela seguinte apresenta os resultados.

	Update Summarization	
	ROUGE-2	ROUGE-SU4
Frases de B	0.07973	0.12037
Frases de A e B	0.08196	0.12215
Δ %	3%	1%

Tabela 4.7 - Comparação do recall obtido usando no grafo apenas as frases dos textos novos vs. todas as frases

Podemos verificar que as frases do histórico, apesar de não serem selecionadas para a criação do sumário, podem influenciar positivamente o ranking das frases dos textos novos. Isto vem confirmar o que Li *et al.* (2008) já haviam afirmado.

Apesar dos custos computacionais, uma vez que o grafo fica mais complexo, decidimos que as implementações baseadas em grafos terão em consideração todas as frases.

4.4. Resultados com Textos da TAC 2008

De seguida apresentamos o resultado das métricas ROUGE-2 e ROUGE-SU4 dos diversos sistemas desenvolvidos, tendo por base os textos da TAC 2008. As tabelas com todos os resultados são disponibilizadas em anexo.

4.4.1. Comparação dos Modelos

O gráfico seguinte mostra os resultados obtidos nas tarefas de sumarização, de *update summarization* e as duas em conjunto.

Para ser mais fácil a comparação entre os modelos dos dois tipos de aprendizagem, os sistemas da aprendizagem supervisionada têm a forma de triângulo.

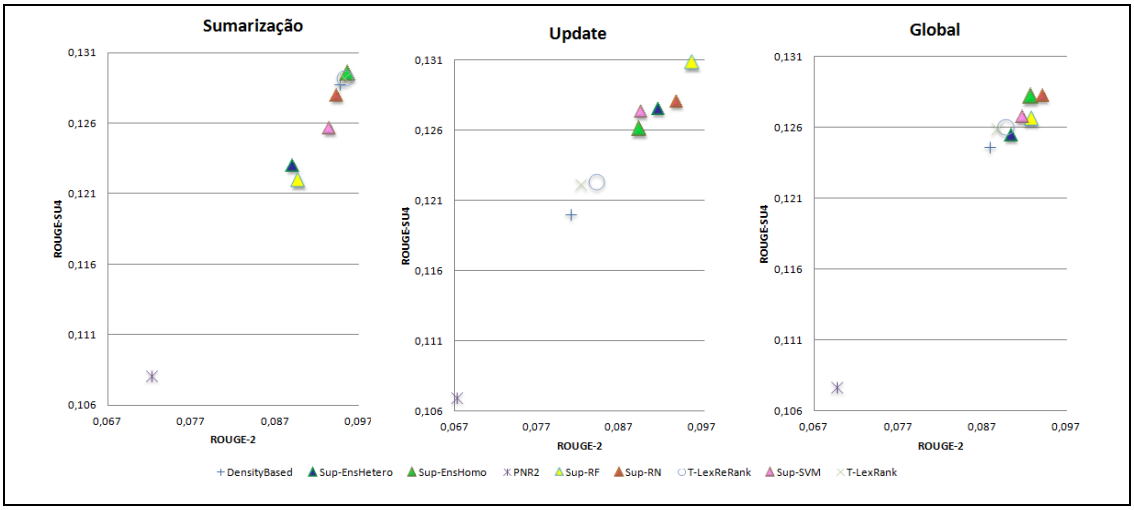


Gráfico 4.2 – Comparação dos Modelos (TAC 2008)

Na sumarização, o Ensemble de Sumarizadores Homogéneos *(Sup-EnsHomo)* conseguiu o melhor desempenho. Importa também destacar os bons resultados do *T-LexRank* (o *T-LexReRank* na sumarização tem sempre os mesmos valores do *T-LexRank*) e do *DensityBased*, confirmando que os modelos baseados em grafos

permitem obter bons resultados na sumarização automática de textos. Pelo contrário, o *PNR*^{2} é nitidamente o que apresenta os valores mais baixos.

Comparando os resultados da sumarização com os de *update summarization*, verificamos que o modelo que obteve melhor resultado na tarefa de sumarização, *Sup-EnsHomo*, passou para o pior dos modelos de aprendizagem supervisionada na tarefa de *update summarization*.

Nesta tarefa, todos os sistemas baseados em aprendizagem não supervisionada apresentam piores resultados que os de aprendizagem supervisionada, quer se analise o ROUGE-2 ou o ROUGE-SU4. Aliás, o desempenho do modelo com pior desempenho na aprendizagem supervisionada, *Sup-EnsHomo*, é superior em cerca de 6.21%, em relação ao *T-LexReRank*, que é o modelo da aprendizagem não supervisionada com o melhor resultado na tarefa de *update summarization*.

Quanto aos resultados globais, muito por influência dos valores mais baixos conseguidos no *update summarization*, todos os modelos baseados em grafos têm piores resultados que os da abordagem supervisionada (tendo em consideração o ROUGE-2). Analisando o ROUGE-SU4 isto já não se verifica, havendo dois sistemas baseados em grafos (*T-LexReRank* e *T-LexRank*) a ficar à frente do Ensemble de Sumarizadores Heterogéneo (*Sup-EnsHetero*).

Pela análise dos resultados globais obtidos com o conjunto de textos da TAC 2008, não se vê ganho na utilização do Ensemble de Sumarizadores Heterogéneos, já que, por exemplo, as *Redes Neuronais* (*Sup-RN*) ou as *Random Forests* (*Sup-RF*) obtêm melhores resultados em qualquer das tarefas.

De forma ainda mais evidente, o *PNR*^{2} mostra os valores mais baixos, havendo um decréscimo de mais de 21% em relação ao sistema que vem imediatamente acima (*DensityBased*).

4.4.2. Comparação com os Outros Sistemas Participantes na TAC2008

Para ser possível averiguar o desempenho dos sistemas reutilizados e/ou desenvolvidos no âmbito desta tese, é relevante compará-los com os sistemas que participaram na TAC 2008 e também com os sistemas de comparação (*upper* e *lower bounds*).

As tabelas seguintes mostram o resultado, tendo em consideração o ROUGE-2 médio dos diversos sistemas. Como há inúmeros sistemas participantes, decidimos apresentar aqui apenas as tabelas com alguns dos sistemas: o sistema de referência *Humanos*, os dois melhor classificados, os nossos sistemas e o sistema de comparação *Topo*.

Os nossos sistemas são indicados a negrito e os sistemas de comparação (LB – *lower bound* e UB – *upper bound*) são mostrados em itálico. É também indicado o *rank* (posição) dos sistemas, obtido a partir dos valores do ROUGE-2.

Pos	Sistema	ROUGE-2
UB	*Humanos*	*0.11606*
1	S43	0.11089
2	S13	0.10978
8	**Sup-EnsHomo**	**0.09562**
9	**T-LexRank**	**0.09518**
10	**T-LexReRank**	**0.09518**
11	**DensityBased**	**0.09465**
13	**Sup-RN**	**0.09432**
15	**Sup-SVM**	**0.09351**
23	**Sup-RF**	**0.08978**
25	**Sup-EnsHetero**	**0.08909**
57	**PNR^2**	**0.07214**
LB	*Topo*	*0.06346*

Tabela 4.8 – Comparação Sistemas Sumarização TAC 2008

Pos	Sistema	ROUGE-2
UB	*Humanos*	*0.11482*
1	S14	0.10015
2	S65	0.09579
4	**Sup-RF**	**0.09548**
5	**Sup-RN**	**0.09357**
7	**Sup-EnsHetero**	**0.09141**
9	**Sup-SVM**	**0.08928**
12	**Sup-EnsHomo**	**0.08904**
20	**T-LexReRank**	**0.08383**
21	**T-LexRank**	**0.08196**
26	**DensityBased**	**0.08070**
52	**PNR^2**	**0.06692**
LB	*Topo*	*0.05354*

Tabela 4.9 – Comparação Sistemas *Update* TAC 2008

Pos	Sistema	ROUGE-2
UB	*Humanos*	*0.11635*
1	S43	0.10361
2	S13	0.09862
6	**Sup-RN**	**0.09422**
9	**Sup-RF**	**0.09287**
10	**Sup-EnsHomo**	**0.09280**
13	**Sup-SVM**	**0.09173**
14	**Sup-EnsHetero**	**0.09048**
16	**T-LexReRank**	**0.08983**
19	**T-LexRank**	**0.08886**
21	**DensityBased**	**0.08794**
55	**PNR^2**	**0.06972**
LB	*Topo*	*0.05871*

Tabela 4.10 – Comparação Sistemas Global TAC 2008

A análise comparativa com os outros sistemas evidencia os bons resultados, nas métricas ROUGE, dos sistemas desenvolvidos.

Na tarefa de sumarização há três sistemas que ficam no top dez (*Sup-EnsHomo*, *T-LexRank* e *T-LexReRank*), sendo que os outros, não contando com o *PNR²*, ficam nos primeiros 25 (entre 80 sistemas em análise).

No *update summarization* os resultados comparativos são ainda melhores, com o modelo *Sup-RF* a ficar em quarto lugar, logo seguido pelo *Sup-RN*, em quinto. Os cinco modelos da aprendizagem supervisionada ficam nos 12 primeiros lugares.

Na comparação global (que contempla sumarização e *update summarization*) todos os sistemas (com exceção do *PNR²*) ficam nos primeiros 21 lugares, havendo três que ficam nos dez primeiros (*Sup-RN*, *Sup-RF* e *Sup-EnsHomo*). Isto é significativo tendo em consideração que os sumários criados são meramente extrativos, sem terem em consideração qualquer pós-processamento ou diminuição do tamanho das frases.

Quanto ao *PNR²* verificamos resultados bastante abaixo dos outros sistemas, o que resulta em classificações consideravelmente más, se bem que acima dos *lower bounds* e à frente de muitos participantes na TAC 2008 (por exemplo, na comparação global, o *PNR²* ficou à frente de 25 sistemas participantes).

4.5. Resultados com Textos da TAC 2009

4.5.1. Comparação dos Modelos

O gráfico seguinte mostra o desempenho dos diversos modelos desenvolvidos com os textos da TAC 2009, tendo em consideração as métricas ROUGE-2 e ROUGE-SU4. São apresentados os resultados para as tarefas de sumarização, de *update summarization* e para as duas em conjunto.

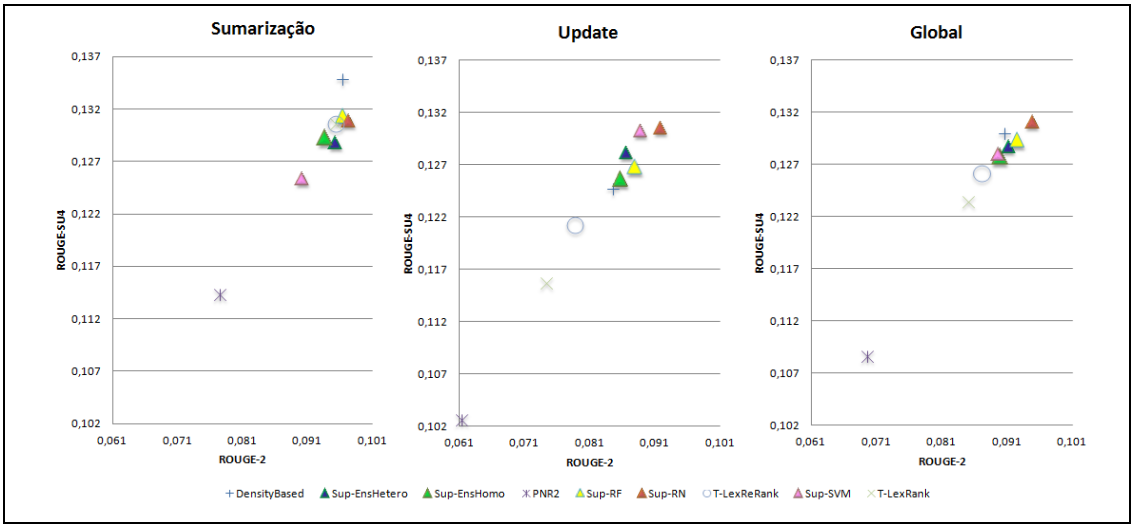


Gráfico 4.3 - Comparação dos Modelos (TAC 2009)

Verificamos que os modelos das *Redes Neuronais* (*Sup-RN*) são os que garantem resultados mais elevados na medida ROUGE-2, sendo que com a ROUGE-SU4, é ultrapassado na tarefa de sumarização pelos sistemas *DensityBased* e *Sup-RF*.

O sistema *Sup-SVM* tem um resultado comparativamente modesto na tarefa de sumarização, ficando apenas à frente do PNR^2, contudo na tarefa de *update summarization* ficam logo atrás do *Sup-RN*.

Os modelos baseados em grafos, tirando o sistema PNR^2, têm um bom desempenho na tarefa de sumarização. Contudo, na tarefa de *update summarization* todos os modelos baseados em aprendizagem supervisionada obtêm melhores resultados, sendo que o melhor destes modelos (*Sup-RN*) é cerca de 8.55% superior ao melhor dos modelos de aprendizagem não supervisionada (*DensityBased*).

Comparando os modelos múltiplos, o *Sup-EnsHetero* apresenta melhores resultados. Contudo importa referir que nunca obtém melhores resultados que os diferentes modelos que o compõem. Isto também acontece com o *Sup-EnsHomo*, já que os modelos formados pelas redes neuronais conseguem melhores resultados que os obtidos pela conjugação dos oito modelos que compõem este Ensemble.

4.5.2. Comparação com os Outros Sistemas Participantes na TAC 2009

Tal como analisado para a TAC 2008, de seguida fazemos uma comparação entre os sistemas desenvolvidos no âmbito desta tese e os sistemas participantes na TAC 2009. Em anexo é possível consultar as tabelas completas.

Pos	Sistema	ROUGE-2
UB	*Humanos*	*0.12553*
1	S34	0.12102
2	S40	0.12049
14	**Sup-RN**	**0.09735**
15	**Sup-RF**	**0.09649**
16	**DensityBased**	**0.09634**
18	**Sup-EnsHetero**	**0.09526**
19	**T-LexRank**	**0.09523**
20	**T-LexReRank**	**0.09523**
26	**Sup-EnsHomo**	**0.09361**
31	**Sup-SVM**	**0.09014**
49	**PNR^2**	**0.07745**
LB	*Topo*	*0.06822*

Tabela 4.11 – Comparação Sistemas Sumarização TAC 2009

Pos	Sistema	ROUGE-2
UB	*Humanos*	*0.10643*
1	S34	0.10433
2	S40	0.10403
6	**Sup-RN**	**0.09180**
9	**Sup-SVM**	**0.08876**
10	**Sup-RF**	**0.08787**
12	**Sup-EnsHetero**	**0.08664**
14	**Sup-EnsHomo**	**0.08570**
16	**DensityBased**	**0.08457**
25	**T-LexReRank**	**0.07871**
34	**T-LexRank**	**0.07438**
49	**PNR^2**	**0.06127**
LB	*Topo*	*0.06048*

Tabela 4.12 – Comparação Sistemas *Update* TAC 2009

Pos	Sistema	ROUGE-2
UB	*Humanos*	*0.11636*
1	S34	0.11289
2	S40	0.11244
9	**Sup-RN**	**0.09496**
10	**Sup-RF**	**0.09260**
12	**Sup-EnsHetero**	**0.09125**
13	**DensityBased**	**0.09073**
17	**Sup-EnsHomo**	**0.08993**
18	**Sup-SVM**	**0.08966**
27	**T-LexReRank**	**0.08723**
30	**T-LexRank**	**0.08512**
49	**PNR^2**	**0.06962**
LB	*Topo*	*0.06457*

Tabela 4.13 – Comparação Sistemas Global TAC 2009

No top 20 da tarefa de sumarização, entre 61 sistemas em comparação, há seis modelos implementados nesta tese, sendo que o modelo *Sup-RN* fica na posição 14, imediatamente seguido pelos modelos *Sup-RF* e *DensityBased*. É relevante salientar que, não contando com o caso excecional do *PNR*^{2}, os modelos baseados em grafos estão entre os 20 sistemas que apresentam melhores resultados.

A classificação é ainda mais interessante no caso da tarefa *de update summarization* em que três dos nossos sistemas (*Sup-RN*, *Sup-SVM* e *Sup-RF*) se encontram entre os dez primeiros.

Analisando o desempenho global (sumarização e *update summarization*) verificamos o bom resultado do *Sup-RN* (nono classificado) e do *Sup-RF* (décimo classificado). Nos dez lugares seguintes há a presença de mais quatro dos sistemas propostos, sendo de realçar a posição do *DensityBased*.

Como já havia acontecido com os textos da TAC 2008, o PNR^2 apresenta o desempenho mais baixo de todos os sistemas propostos.

Importa concluir que os sistemas desenvolvidos (excluindo o PNR^2) ficaram todos na primeira metade da tabela.

Comparando com as classificações da TAC 2008, verificamos que há na TAC 2009 mais sistemas com melhores resultados do que os nossos sistemas. Pensamos que isto pode dever-se ao facto de os nossos modelos terem sido treinados e os parâmetros ajustados tendo em consideração apenas os textos da TAC 2008, ou pelo facto de os sistemas participantes na TAC 2009 serem mais sofisticados, uma vez que a tarefa de *update summarization* já estava mais amadurecida em 2009.

4.6. Resultados com Textos da DUC 2007

Como já foi salientado anteriormente, a tarefa de *update summarization* surgiu pela primeira vez nesta conferência, sendo então uma tarefa piloto. Isto pode ter influência nos resultados obtidos.

4.6.1. Comparação dos Modelos

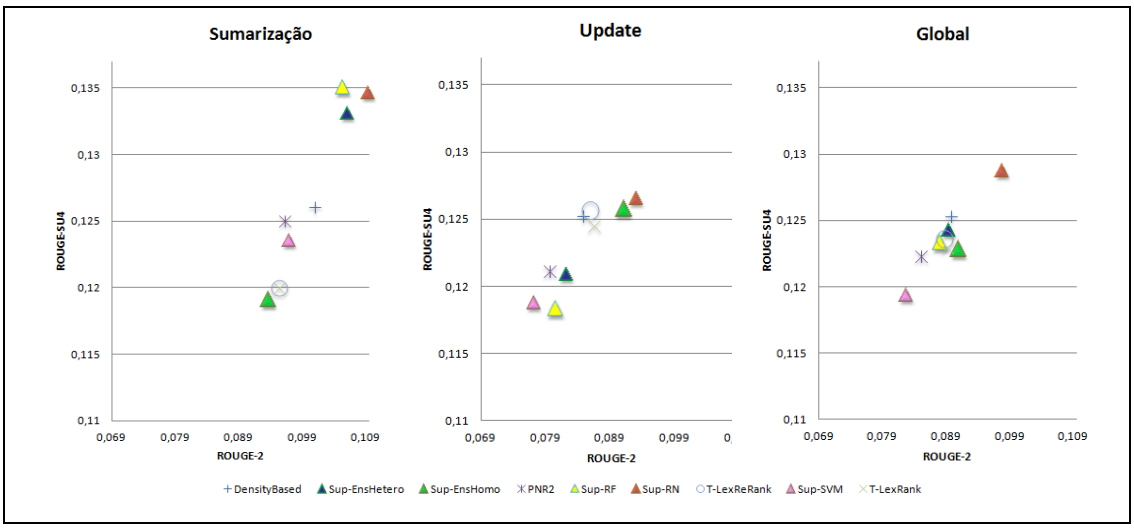


Gráfico 4.4 – Comparação dos Modelos (DUC 2007)

Um aspeto que se evidencia logo à partida é o facto de o PNR^2 não ser o sistema com pior desempenho, o que aconteceu com os textos da TAC 2008 e da TAC 2009.

Convém salientar que os resultados continuam a ser inferiores aos do artigo que deu a conhecer o PNR^2 (Li *et al.* (2008)) e que também utilizavam os textos da DUC 2007. Isto provavelmente ocorre pelo facto dos autores apenas utilizarem a primeira das duas tarefas de *update summarization*: sumarizar B, tendo como histórico os textos de A, ao contrário do que nós fizemos, que tem em consideração também a sumarização dos textos de C, dado que são conhecidos A e B.

Quanto ao modelo que apresenta melhores resultados, o *Sup-RN* aparece no topo das três tabelas. Os lugares seguintes variam conforme a tarefa, sendo de salientar a boa classificação de três sistemas baseados em grafos na tarefa de *update summarization*.

Se com os textos da TAC 2008 e TAC 2009 era evidente verificar que os modelos que utilizavam aprendizagem supervisionada tinham um resultado superior aos modelos da aprendizagem não supervisionada, isso já não é evidente com os textos da DUC 2007, onde os modelos *Sup-RF* e *Sup-SVM* tiveram um baixo desempenho na tarefa de *update summarization*.

No que diz respeito aos modelos múltiplos, verifica-se que o *Sup-EnsHetero* consegue uma boa posição na sumarização, mas desce no *update summarization*. Maior diferença ocorre com o *Sup-EnsHomo* que apresenta o pior resultado na sumarização, mas que sobe para segundo no *update summarization*.

4.6.2. Comparação com os Outros Sistemas Participantes na DUC 2007

Na sumarização o *Sup-RN* fica em quarto lugar, entre 31 sistemas em comparação, descendo para quinto na tarefa de *update summarization*.

Na tarefa de sumarização há quatro dos nossos modelos no top 10, e dois no *update summarization*.

Na comparação global o *Sup-RN* alcança o terceiro lugar, havendo três outros modelos (os dois *Ensembles* e o *DensityBased*) que ficam nos dez primeiros.

Pos	Sistema	ROUGE-2
1	S40	0.12724
UB	*Humanos*	*0.11896*
2	S55	0.11519
4	**Sup-RN**	**0.10936**
5	**Sup-EnsHetero**	**0.10602**
6	**Sup-RF**	**0.10531**
8	**DensityBased**	**0.10094**
13	**Sup-SVM**	**0.09690**
14	**PNR^2**	**0.09616**
16	**T-LexRank**	**0.09526**
17	**T-LexReRank**	**0.09526**
19	**Sup-EnsHomo**	**0.09370**
LB	*Topo*	*0.06199*

Tabela 4.14 – Comparação Sistemas Sumarização DUC 2007

Pos	Sistema	ROUGE-2
UB	*Humanos*	*0.13498*
1	S40	0.10550
2	S45	0.09765
5	**Sup-RN**	**0.09299**
7	**Sup-EnsHomo**	**0.09104**
11	**T-LexRank**	**0.08656**
12	**T-LexReRank**	**0.08595**
13	**DensityBased**	**0.08482**
14	**Sup-EnsHetero**	**0.08231**
16	**Sup-RF**	**0.08064**
17	**PNR^2**	**0.07968**
19	**Sup-SVM**	**0.07717**
LB	*Topo*	*0.05916*

Tabela 4.15 – Comparação Sistemas *Update* DUC 2007

Pos	Sistema	ROUGE-2
UB	*Humanos*	*0.12757*
1	S40	0.11207
2	S55	0.09931
3	**Sup-RN**	**0.09785**
8	**Sup-EnsHomo**	**0.09097**
9	**DensityBased**	**0.08964**
10	**Sup-EnsHetero**	**0.08942**
11	**T-LexRank**	**0.08906**
13	**T-LexReRank**	**0.08863**
16	**Sup-RF**	**0.08808**
17	**PNR^2**	**0.08497**
18	**Sup-SVM**	**0.08266**
LB	*Topo*	*0.06200*

Tabela 4.16 – Comparação Sistemas Global TAC 2007

4.7. Comparação das Classificações nos Três Conjuntos de Textos

Os sistemas desenvolvidos obtiveram diferentes resultados a que correspondem diferentes posições nas tabelas dos resultados. Ao estudarmos as posições conseguidas nos diferentes conjuntos de textos, pretendemos analisar como é que o resultado dos sistemas fica posicionado quando comparado com os outros que participaram nas três competições.

Dado que o número de sistemas em *competição* é diferente em cada um dos conjuntos de textos, optamos por analisar a posição relativa em vez da posição

absoluta, para que o sistema que fique em primeiro lugar tenha o valor 1 e o que fique em último tenha o valor 0, sendo o valor das restantes posições distribuído proporcionalmente.

O gráfico seguinte mostra este estudo para a medida ROUGE-2, obtida com os diversos sistemas a efetuarem globalmente as tarefas de sumarização e *update summarization*.

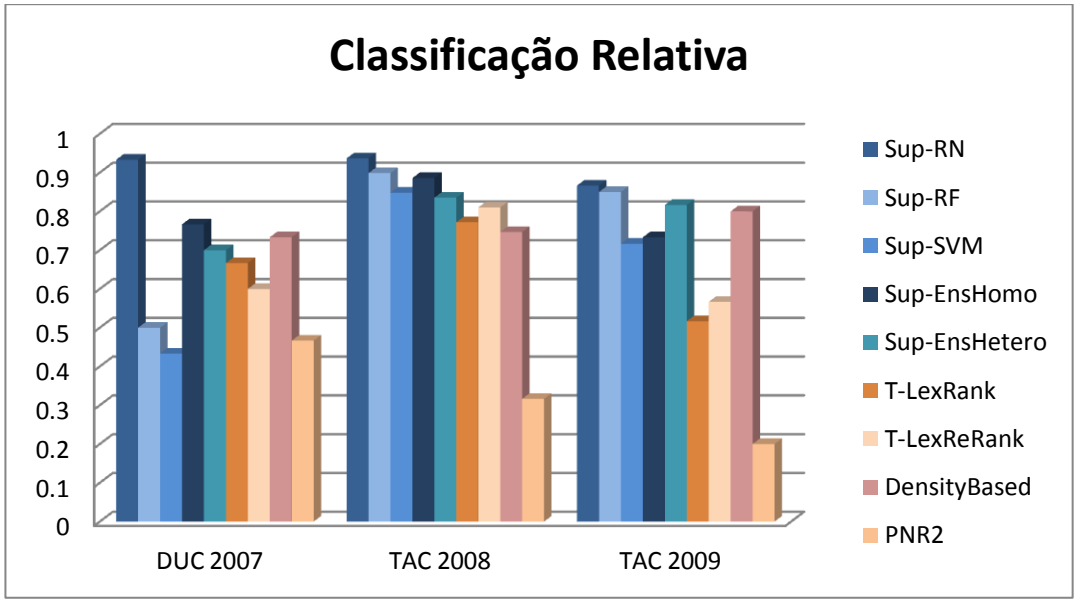


Gráfico 4.5 – Comparação da Classificação Relativa nos Três Conjuntos de Textos

Há duas conclusões que é possível verificar rapidamente:

- o modelo *Redes Neuronais* (*Sup-RN*) consegue sempre a melhor classificação;
- a classificação dos modelos baseados em aprendizagem supervisionada (sem considerar os modelos múltiplos) surge sempre pela mesma ordem: *Sup-RN*, *Sup-RF* e *Sup-SVM*.

Para além destas, não são evidentes mais conclusões que se apliquem aos três conjuntos de textos. Vejamos alguns exemplos:

- comparando os modelos múltiplos, o Ensemble de Sumarizadores Homogéneos consegue uma melhor classificação, na DUC 2007 e TAC 2008, mas isto já não se confirma na TAC 2009;
- o PNR^2 apresenta a pior classificação na TAC 2008 e TAC 2009, sendo que a diferença é significativa; contudo, isto já não se verifica na DUC 2007 onde fica à frente do modelo *Sup-SVM*;

- comparando os modelos baseados em aprendizagem não supervisionada, na DUC 2007 e na TAC 2009 a melhor classificação é obtida pelo *DensityBased*, mas no TAC 2008 o melhor classificado é o *T-LexReRank*.

4.8. Resultados nos Três Conjuntos de Textos

Com o intuito de analisarmos o desempenho dos nove sistemas propostos, tendo em consideração todos os textos (englobando os três conjuntos de textos), procedemos à avaliação das medidas ROUGE criando um cenário em que existe uma competição composta por todos os textos das diferentes conferências. Neste cenário, estão envolvidos textos de 102 tópicos diferentes, sendo avaliados 1926 sumários, tendo por base a comparação com 856 sumários modelo (divididos pelos diferentes tópicos).

4.8.1. Comparação dos Modelos

	R-2	R-SU4
Sup-RN	0.09806	0.13043
DensityBased	0.09659	0.12965
T-LexRank	0.09595	0.12956
T-LexReRank	0.09595	0.12956
Sup-EnsHomo	0.09499	0.12856
Sup-RF	0.09471	0.12769
Sup-EnsHetero	0.09378	0.12670
Sup-SVM	0.09250	0.12499
PNR^2	0.07712	0.11271

Tabela 4.17 – ROUGE Sumarização Textos Todos

	R-2	R-SU4
Sup-RN	0.09251	0.12854
Sup-RF	0.09025	0.12714
Sup-EnsHetero	0.08757	0.12606
Sup-EnsHomo	0.08725	0.12489
Sup-SVM	0.08689	0.12698
DensityBased	0.08247	0.12298
T-LexReRank	0.08175	0.12196
T-LexRank	0.07980	0.11985
PNR^2	0.06648	0.10697

Tabela 4.18 – ROUGE *Update Summarization* Textos Todos

	R-2	R-SU4
Sup-RN	0.09538	0.12980
Sup-RF	0.09220	0.12741
Sup-EnsHomo	0.09162	0.12747
Sup-EnsHetero	0.09080	0.12677
Sup-SVM	0.08967	0.12630
DensityBased	0.08941	0.12649
T-LexReRank	0.08883	0.12598
T-LexRank	0.08763	0.12476
PNR^2	0.07212	0.11037

Tabela 4.19 – ROUGE Global Textos Todos

Dos resultados apresentados verificamos que o modelo *Sup-RN* consegue o melhor desempenho em qualquer das tarefas e o PNR^2 mantém os piores resultados.

Analisando os modelos supervisionados, não incluindo os múltiplos, constatamos que os resultados do modelo *Sup-RF* é melhor que o *Sup-SVM*, ficando contudo atrás do *Sup-RN*.

Verificamos ainda que na sumarização, os sistemas baseados em aprendizagem não supervisionada conseguem uma boa classificação, surgindo imediatamente a seguir às *Redes Neuronais*.

Contudo isto não se mantém na tarefa de *update summarization*, onde os modelos baseados em aprendizagem supervisionada obtêm todos melhores resultados que os sistemas baseados em aprendizagem não supervisionada. Observamos um decréscimo acentuado nestes últimos sistemas. Por exemplo, o resultado do *T-LexRank* diminui em 16.83% da tarefa de sumarização para a de *update summarization*.

Verificamos que em todas as tarefas, o *DensityBased* supera os outros sistemas baseados em grafos. Por sua vez, *T-LexReRank* obtém resultados superiores ao *T-LexRank* quer na tarefa de *update summarization* quer globalmente. Conforme, já verificamos anteriormente, estes dois sistemas são iguais na tarefa de sumarização.

Quanto aos modelos múltiplos, o *Sup-EnsHetero* é melhor na tarefa de *update summarization*, mas é ultrapassado pelo *Sup-EnsHomo* quer na tarefa de sumarização quer globalmente (ao serem consideradas as duas tarefas).

4.8.2. Aprendizagem Supervisionada vs. Aprendizagem não Supervisionada

Na sumarização os sistemas baseados em aprendizagem não supervisionada conseguem bons resultados, tanto no ROUGE-2 como no ROUGE-SU4.

Por outro lado, tanto no *update summarization* como na avaliação global, os sistemas baseados em aprendizagem supervisionada conseguem superar, nas duas medidas, os sistemas baseados em aprendizagem não supervisionada.

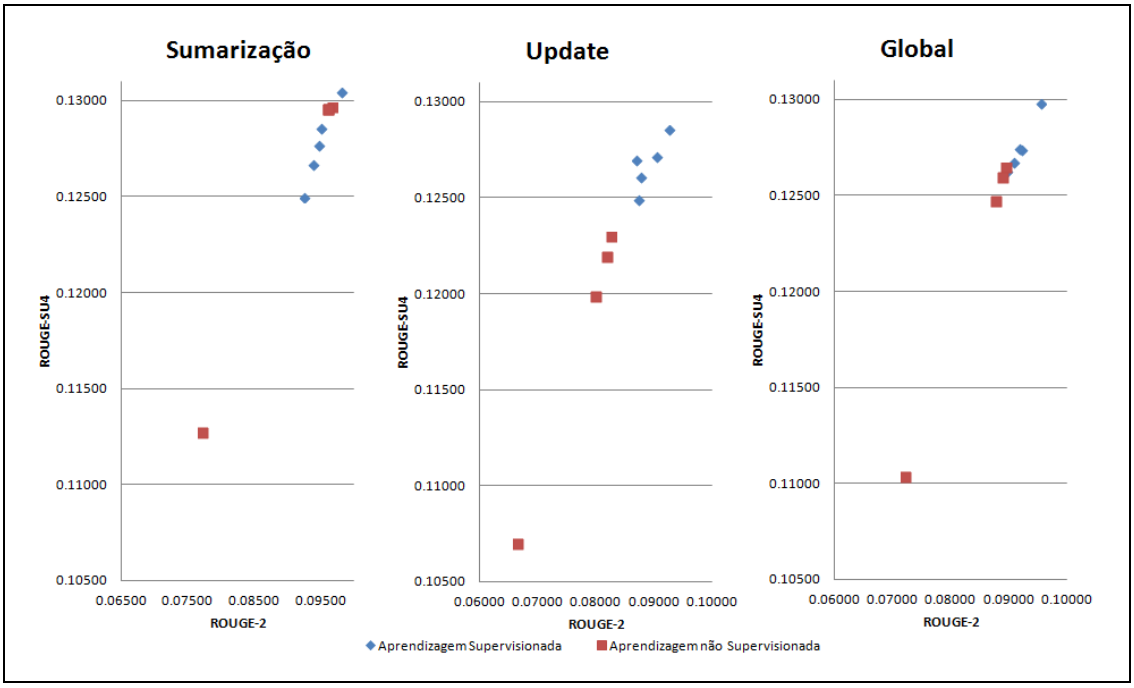


Gráfico 4.6 – Comparação Aprendizagem Supervisionada vs. Aprendizagem Não Supervisionada

4.8.3. Teste Estatístico

Para verificar se se pode concluir que as médias do ROUGE-2 são iguais (analisando os resultados globais das duas tarefas), para um nível de significância de 5%, realizamos o teste de Krsukal-Wallis e verificamos que devemos rejeitar essa hipótese (p-value aproximadamente 0). Fazendo uma análise par-a-par entre os sistemas em análise, verificamos que todos os sistemas apresentam uma média significativamente diferente da do *PNR*^{2}.

Fizemos também um teste *Wilcoxon signed-ranks* para comparar o desempenho dos modelos *Redes Neuronais* e *DensityBased*, com a hipótese nula a indicar que os dois modelos têm um desempenho equivalente. Com um p-value=0.0859, não podemos rejeitar a hipótese de os dois modelos serem equivalentes, para um nível de significância de 5%.

4.8.4. Comparação Par-a-Par

Uma vez que com os testes estatísticos não se rejeita a hipótese dos sistemas terem um desempenho semelhante (não considerando o sistema *PNR*^{2}), decidimos comparar alguns modelos par-a-par, analisando usando sistema *Ganha-Empata-*

Perde. Consideramos que um sistema ganha se o ROUGE-2 que obtém for superior ao do outro sistema. Em cada *confronto*, são analisados os resultados de 214 comparações, considerando o ROUGE-2 da combinação das tarefas de sumarização e de *update summarization*.

Redes Neuronais vs. Outros Modelos

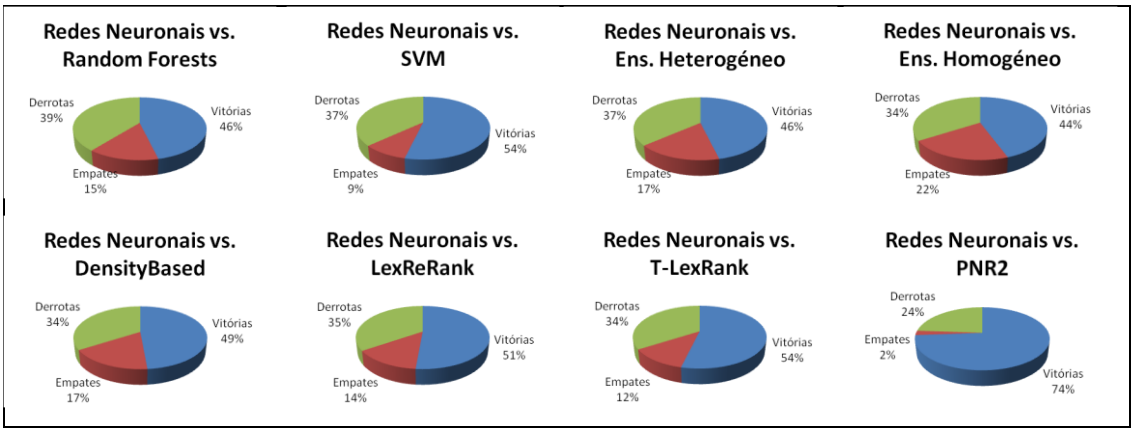


Gráfico 4.7 – Redes Neuronais vs. Outros Modelos (*Ganha-Empata-Perde*)

Verificamos que o modelo das *Redes Neuronais* vence sempre mais vezes do que as que perde. Por exemplo, contra o *DensityBased*, o modelo baseado em aprendizagem não supervisionada com melhor desempenho, o modelo *Redes Neuronais* consegue vencer 49% das vezes, perdendo apenas em 34% das comparações.

É interessante verificar que, apesar de ser o que apresenta melhores resultados, o modelo *Redes Neuronais* contra o *PNR*^{2}, sistema que se destaca pelo oposto, é derrotado quase um quarto das vezes, isto mostrando que é realmente difícil afirmar que um modelo é incontestavelmente superior a todos os outros.

Ensembles

Fazendo o mesmo teste *Ganha-Empata-Perde* entre os dois modelos múltiplos, verificamos que o *Ensemble de Sumarizadores Homogéneos* ganha metade das vezes, perdendo apenas em 41% dos casos, o que confirma a melhor posição na classificação.

Sistemas Baseados em Aprendizagem Não Supervisionada

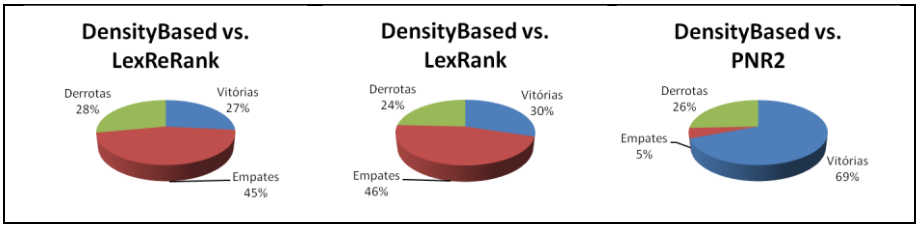


Gráfico 4.8 – *DensityBased* vs. Outros Sistemas Baseados em Grafos (*Ganha-Empata-Perde*)

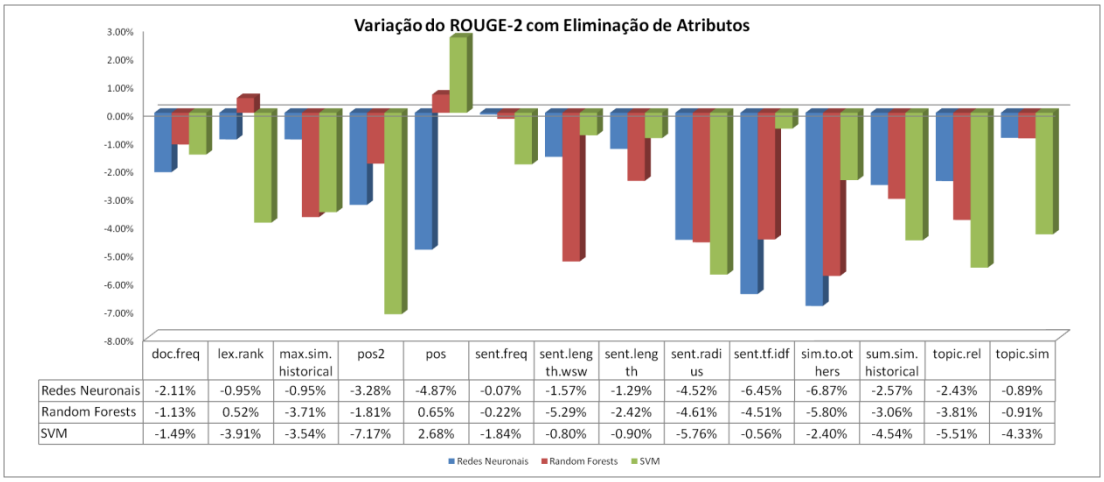
O ROUGE-2 médio do sistema *DensityBased* é superior ao dos outros três sistemas baseados em grafos. Contudo, fazendo a comparação *Ganha-Empata-Perde*, verificamos que o *T-LexReRank* consegue ganhar mais vezes ao *DensityBased* do que o que ocorre ao contrário, apesar de estarmos perante uma diferença mínima de 1%. Neste *confronto* prevalecem os empates, ou seja, os sumários gerados por ambos os sistemas é coincidente em cerca de 45% das situações.

Quanto às outras duas comparações, o *DensityBased* ganha sempre mais do que as vezes que perde, com maior amplitude na comparação com o PNR^2.

4.9. Importância dos Atributos

Com o intuito de analisar a importância dos atributos nos resultados de cada um dos modelos de aprendizagem supervisionada, analisamos a medida ROUGE-2 quando se elimina um atributo de cada vez. Desta forma percebemos qual é a variação e o impacto que a retirada desse atributo causa no desempenho dos sistemas. Uma variação negativa indica que os resultados do sistema baixaram, podendo-se concluir que o atributo é importante para o seu desempenho. Quanto mais negativa for essa variação mais relevante é o atributo. Um processo semelhante a este foi implementado por Valizadeh e Brazdil (2014b). Bysani *et al.* (2009) adotaram um método diferente, começando sem nenhum atributo e acrescentando um de cada vez, avaliaram o ganho obtido com esse atributo.

Utilizamos os textos da TAC 2008 para treinar os modelos e os da TAC 2009 para a avaliação. A base de comparação é o ROUGE-2 dos modelos quando treinados com todos os atributos.



	doc.freq	lex.rank	max.sim. historical	pos2	pos	sent.freq	sent.length.wsw	sent.length	sent.radius	sent.tf.idf	sim.to.others	sum.sim. historical	topic.rel	topic.sim
Redes Neuronais	-2.11%	-0.95%	-0.95%	-3.28%	-4.87%	-0.07%	-1.57%	-1.29%	-4.52%	-6.45%	-6.87%	-2.57%	-2.43%	-0.89%
Random Forests	-1.13%	0.52%	-3.71%	-1.81%	0.65%	-0.22%	-5.29%	-2.42%	-4.61%	-4.51%	-5.80%	-3.06%	-3.81%	-0.91%
SVM	-1.49%	-3.91%	-3.54%	-7.17%	2.68%	-1.84%	-0.80%	-0.90%	-5.76%	-0.56%	-2.40%	-4.54%	-5.51%	-4.33%

Gráfico 4.9 – Variação do ROUGE-2 com a Eliminação de Atributos (*Update Summarization* – TAC 2009)

Podemos verificar que o atributo que traria maior quebra no desempenho dos três modelos, caso fosse removido do conjunto de dados, é o *sim.to.others* que seria responsável pela diminuição dos resultados em cerca de 6.87% nas *Redes Neuronais*, 5.80% nas *Random Forests* e 2.40% nas *SVM*.

Segue-se o *sent.radius* e o *pos2* com um decréscimo médio de 4.96% e 4.08% respetivamente. Quanto a este atributo (*pos2*) verificamos que a sua eliminação do conjunto de dados provocava a maior variação num único modelo: diminuição do resultado do modelo *SVM* em mais de 7%.

Os atributos cuja eliminação provocaria menos impacto tendo em consideração a globalidade dos três modelos, seriam *pos* e *sent.freq*, com variações médias negativas de 0.51% e 0.71%, respetivamente.

Da análise do gráfico é possível verificar ainda que existem dois atributos que não têm variação negativa: o *lex.rank* (no modelo *Random Forests*) e o *pos* (nos modelos *Random Forests* e *SVM*). Isto indicia que se não entrassem no treino desses modelos seria possível obter melhores resultados. Não foram excluídos na seleção de atributos, porque são importantes para os outros modelos. A eliminação do *lex.rank* melhoraria o resultado do modelo das *Random Forests* em 0.52%, mas iria reduzir o desempenho das *Redes Neuronais* em cerca de 0.95% e das *SVM* em 3.91%. De forma semelhante, apesar de ao eliminar o atributo *pos* haver um ganho de 0.65% (*Random Forests*) e de 2.68% (*SVM*), isso iria provocar um decréscimo de 4.87% no modelo *Redes Neuronais*.

5. Conclusões

No início desta tese colocamos algumas questões que queríamos responder neste trabalho. Com as experiências realizadas pensamos ter sido capazes de responder a todas essas questões. Contudo, como não poderia deixar, outras foram surgindo e, não houve tempo para tentar esclarecer todas as alternativas e cenários possíveis. Neste capítulo, tecemos algumas considerações finais relativas aos resultados e trabalho realizado e apresentamos diversas possibilidades que ficam em aberto e que poderão ser exploradas no futuro.

5.1. Sobre os sistemas e resultados

No início tínhamos algumas expectativas quanto aos resultados dos diversos modelos implementados. Estávamos confiantes num bom desempenho dos resultados nos modelos baseados em grafos na sumarização, já que há muita literatura que refere isso. Mas também estávamos expectantes de obter resultados significativos com a implementação do sistema *PNR*^{2} por parecer uma ideia muito promissora e pelos bons resultados apresentados pelos autores. No entanto, tínhamos algumas dúvidas se, simplesmente com a introdução de técnicas para detetar a redundância com os textos do histórico, seríamos capazes de adequar os algoritmos específicos para a sumarização à tarefa de *update summarization*.

Nos modelos baseados em aprendizagem supervisionada tínhamos a convicção de que, conseguindo fazer um levantamento apurado dos atributos a incluir na fase de treino, seria possível obter um bom desempenho tanto na sumarização como no *update summarization*. E pareceu-nos muito promissora a ideia de utilizar modelos múltiplos, de forma a aproveitar quer as diferenças dos vários algoritmos quer as diferentes avaliações das frases pelos humanos, de forma a criar modelos que em conjunto superassem qualquer um dos modelos individuais que o compunham.

Fazendo uma análise final e global aos resultados, verificamos que os modelos da abordagem não supervisionada, todos baseados em grafos, conseguem um bom desempenho na tarefa de sumarização. Contudo, há um decréscimo evidente quando se aplicam os algoritmos ao *update summarization*, mesmo tendo sido implementadas

alterações para tentar evitar a ocorrência de redundância com o histórico. Desta forma, na tarefa de *update* as abordagens supervisionadas conseguem melhores resultados.

Importa realçar que não é notório um comportamento constante dos sistemas baseados em abordagens não supervisionadas, nos diferentes conjuntos de textos. Com os dados da TAC 2008, o *T-LexReRank* é o que tem melhor desempenho, seguindo-se o *T-LexRank*, contudo, com textos da DUC 2007 e TAC 2009, o *DensityBased* passa a destacar-se, seguido pelo *T-LexRank*, com textos da DUC 2007, ou pelo *T-LexReRank*, se forem considerados os textos da TAC 2009. A única constante é o facto de o PNR^2 ser sempre o que tem um desempenho inferior.

Pelo contrário, nos modelos da abordagem supervisionada, em todos, nos três conjuntos de textos, o modelo *Redes Neuronais* supera o modelo das *Random Forests* que por sua vez é superior ao modelo *SVM*.

O modelo *Redes Neuronais* é o que permite alcançar melhores resultados, tendo em consideração os três cenários: sumarização, *update summarization* e as duas tarefas em conjunto.

Pelo contrário, o PNR^2 evidencia-se claramente com os piores resultados, não se verificando as expectativas que tínhamos inicialmente. Importa referir que apenas com os textos da DUC 2007, este sistema consegue resultados melhores que alguns dos outros modelos. Apesar de não termos conseguido os bons resultados indiciados por Li *et al.* (2008), confirmamos os resultados não tão animadores das experiências realizadas por outros autores, com outros conjuntos de textos (por exemplo, Li *et al.* (2013)).

O modelo que utiliza *Random Forests* tem bons resultados, principalmente na tarefa de *update summarization*, surgindo em termos globais logo a seguir às *Redes Neuronais*. Mas estávamos à espera de um desempenho mais semelhante ao destes dois modelos por parte do modelo *SVM*. Tendo em consideração os bons resultados relatados por Bysani *et al.* (2009), talvez seja possível ajustar os diversos parâmetros deste algoritmo de forma a alcançar melhores resultados, principalmente na tarefa de sumarização.

Os modelos múltiplos não conseguiram alcançar os resultados que inicialmente tínhamos pensado. Na realidade, e limitando estas conclusões meramente à analise dos resultados, verificamos que não há justificação para utilizar estes modelos na medida em que globalmente são ultrapassados pelos modelos de um único preditor. O *Ensemble*

de Sumarizadores Heterogéneos fica sempre atrás do modelo baseado em redes neuronais. O *Ensemble de Sumarizadores Homogéneos* apenas consegue um resultado interessante na tarefa de sumarização com o conjunto da TAC 2008, ficando à frente de todos os outros. Contudo com todos os outros textos, e em qualquer tarefa, nunca foi capaz de se aproximar do modelo das *Redes Neuronais*.

Nos modelos de abordagem não supervisionada, os resultados são animadores na tarefa de sumarização, o mesmo não acontecendo na tarefa de *update summarization*. Parece evidente que é necessário aperfeiçoar os algoritmos para melhor incorporarem o conceito de *novidade*. As implementações feitas, apesar de até não ficarem muito mal posicionadas, quando em comparação com outros sistemas, alcançam resultados inferiores aos dos modelos baseados em aprendizagem supervisionada.

O *DensityBased* foi de todos os algoritmos baseados em aprendizagem não supervisionada o que melhor desempenho obteve. Isto é particularmente evidente na tarefa de sumarização, confirmando os resultados de Valizadeh e Brazdil (2013), mas também no *update summarization*. Aliás, tendo em consideração os resultados globais, o *DensityBased* consegue um valor médio do ROUGE-SU4 superior ao obtido pelo modelo *SVM*.

O *Reranker for Update Summarization* (*LexReRank*), desenvolvido no âmbito desta tese, consegue melhorar os resultados do *T-LexRank*, o que é particularmente gratificante.

De forma análoga, a introdução de novos atributos para o conjunto de dados de *update summarization* também foi relevante para o desempenho dos modelos de abordagem supervisionada. Tendo em consideração os vários atributos, os únicos específicos para *update summarization* que se mantiveram após o processo de seleção de atributos, foram dois pensados por nós: o *max.sim.history* que evidencia o relacionamento mais elevado que a frase tem com os textos de histórico; e o *sum.sim.history* que permite perceber o relacionamento global da frase com o histórico. Se algum destes atributos fosse retirado do conjunto de treino, o desempenho dos preditores teria um decréscimo de 2.73% (*max.sim.history*) ou de 3.39% (*sum.sim.history*) no ROUGE-2 da tarefa de *update summarization.*

Ainda da análise do conjunto de atributos, importa referir que o score do *T-LexRank*, um atributo obtido com um método de aprendizagem não supervisionada, também é relevante enquanto input dos modelos de abordagem supervisionada.

5.2. Propostas para Trabalho Futuro

Outras formas de avaliação

A comparação entre os sistemas foi realizada tendo apenas em consideração as medidas ROUGE, o que é certamente escasso para avaliar se os sumários são realmente de qualidade. Será importante realizar outro tipo de avaliação automática ou semiautomática (por exemplo, o método da Pirâmide). Mas estamos em crer que a verdadeira prova deverá ser a avaliação manual, de forma a analisar a qualidade linguística dos sumários, tendo em conta a gramática, a clareza e a coerência.

Melhorar o Pré-processamento

Estamos convencidos que a qualidade do sumário depende muito da forma como é feito o pré-processamento, em particular, a divisão das frases. Esta tarefa, em princípio simples, veio-se a revelar difícil de resolver, tendo em consideração os inúmeros casos particulares que é necessário ter em atenção. Apesar de todo o tempo que investimos nesta tarefa, estamos em crer que continuam a ocorrer situações incorretas e que devem ser corrigidas. Para se perceber a importância desta tarefa, basta ler Conroy *et al.* (2009), autores do *CLASSY*, um dos sistemas melhor classificados na TAC 2009, e verificar que uma das inovações que eles mais relevam no seu sistema é a criação de um novo algoritmo, o *F-SPLIT*, para a divisão de frases de forma a minimizar os erros ocorridos com os outros algoritmos que usavam até então.

Pós-processamento

Nos sistemas implementados, as frases são inseridas nos sumários exatamente como surgem nos textos de origem. Será importante considerar a implementação de técnicas de pós-processamento, para melhorar a qualidade dos sumários. Estas técnicas poderão visar uma forma mais aperfeiçoada de eliminar a redundância, melhorar a coerência ou a leitura dos sumários.

Uma tarefa importante será resolver o problema das anáforas[1]. Por exemplo, se num sumário é inserida a frase "*Ele confirmou-o inequivocamente*", é importante que seja fácil de perceber quem é o *ele* e o que é que ele confirmou. Nos sistemas que desenvolvemos este problema não é tido em consideração e isso nota-se em vários sumários.

Redução do tamanho das frases (*sentence reduction*)

Num cenário em que os sumários estão limitados a um número reduzido de palavras, julgamos que a redução do tamanho das frases, retirando partes irrelevantes, poderá ser muito interessante. A *simples* remoção de orações subordinadas explicativas, que não modificam a ideia original das frases, e que geralmente surgem entre vírgulas ou dentro de parêntesis, pode ser um bom começo. Mas também se pode partir para métodos mais elaborados como a indução de regras de redução do tamanho de frases, como demonstrado em Cordeiro (2010).

Utilização de métodos de otimização

Pensamos que o recurso a técnicas e métodos da área da otimização poderão conduzir a ganhos significativos nos resultados. Nos sistemas que desenvolvemos houve o cuidado de tentar ajustar os vários parâmetros de forma a obter os melhores resultados, mas seguiu-se apenas uma abordagem gananciosa (*greedy*).

Outra tarefa em que também se pode aproveitar a otimização é na seleção das frases para o sumário. Atualmente selecionam-se as frases que têm um score mais elevado (desde que caibam no limite disponível e respeitem o *threshold* de semelhança com as outras já presentes no sumário). Mas uma abordagem mais interessante seria, ao invés de apenas decidir com base na frase seguinte, usar a otimização para decidir qual o melhor conjunto de frases a selecionar, tendo em consideração as restrições (por exemplo, o tamanho máximo do sumário).

Seria portanto interessante experimentar outras técnicas, por exemplo, algoritmos genéticos ou programação linear inteira para conseguir as configurações mais vantajosas.

[1] Segundo a Infopédia (http://www.infopedia.pt/$anafora-(linguistica)), uma anáfora é um "processo sintático-semântico que consiste na retoma de uma palavra ou de uma expressão mencionadas num contexto de vizinhança linguística".

Alterações aos sistemas criados

Pensamos que há ainda vários aspetos que deverão ser experimentados nos sistemas desenvolvidos.

No caso do *SVM* parece-nos que, fazendo uma seleção de atributos diferente, poderá ser possível melhorar o seu desempenho. E acreditamos também que se conseguirmos melhorar o *SVM*, o Ensemble de Sumarizadores Heterogéneos terá consequentemente uma melhoria, que talvez seja suficiente para conseguir ter resultados mais interessantes que os três regressores sozinhos.

Ainda no Ensemble de Sumarizadores Heterogéneos poderá ser interessante experimentar outro tipo de votação que não a uniforme, dando um peso maior a algum dos algoritmos. Ou ainda experimentar um outro tipo de conjugação dos votos. Por exemplo, em vez da média dos scores atribuídos por cada um dos modelos, podíamos optar pelo maior score de todos.

Quanto ao Ensemble de Sumarizadores Homogéneos colocamos a hipótese de que se a seleção de atributos for feita de forma autónoma, para cada um dos oito modelos, talvez se consigam melhores resultados.

Como os sistemas baseados em aprendizagem não supervisionada conseguem resultados mais interessantes na tarefa de sumarização, pensamos que, investindo um pouco mais em conseguir um algoritmo capaz de detetar melhor a novidade, talvez se consigam aproximar os resultados da tarefa de *update summarization* àqueles que se obtêm na sumarização.

Isto são conjeturas que deverão ser analisadas futuramente, para averiguar se se confirmam ou não.

Exploração com outro tipo de textos

Ao longo deste trabalho, assim como acontece com a grande parte dos trabalhos disponíveis sobre sumarização e *update summarization*, apenas usamos como base para sumarização, textos noticiosos. É importante experimentar os sistemas desenvolvidos com outros textos, em particular textos maiores. Estamos a pensar especialmente em artigos académicos ou científicos, onde julgamos o *update summarization*, poderá ser uma ferramenta extremamente poderosa.

Com textos de grandes dimensões certamente surgirão novos e interessantes desafios, nomeadamente no que respeita à questão da dimensionalidade do espaço vetorial e à

exigência computacional esperada. Desafios muito interessantes, com os quais esperamos ter de lidar num futuro próximo.

Será interessante fazer a comparação dos sistemas baseados em aprendizagem supervisionada com os de aprendizagem não supervisionada. Avançamos com a hipótese de que nestes caso será ainda mais interessante a escolha dos sistemas baseados na aprendizagem supervisionada.

Processamento com fluxo contínuo de dados

Um trabalho que também poderá ser interessante futuramente, e que deverá merecer a nossa atenção, é o tratamento dos documentos em fluxo contínuo de dados, em vez de os tratar a todos em lote. Num cenário real, dificilmente os textos estarão todos disponíveis ao mesmo tempo e não se pode estar à espera de os obter a todos para fazer o processamento. Nesse cenário, os sistemas deverão ser capazes de processar os documentos conforme eles forem chegando e ter a capacidade de se ir adaptando às novas características que forem surgindo.

5.3. Considerações Finais

Julgamos que todos os objetivos a que nos tínhamos proposto foram alcançados.

Ao longo deste trabalho foram criados e avaliados muitos milhares de sumários, contando os vários sistemas, para os três conjuntos de dados, considerando a seleção de atributos e os diversos sumários de comparação (*baselines* e *upper bounds*) e o ajuste dos vários parâmetros.

Implementamos e comparamos diversos sistemas capazes de efetuar sumarização e *update summarization*, e procedemos a testes com três diferentes conjuntos de dados. Ao implementarmos sistemas baseados em aprendizagem supervisionada e aprendizagem não supervisionada, tivemos oportunidade de fazer uma análise comparativa destes dois tipos de abordagens.

Que seja do nosso conhecimento, alguns dos sistemas nunca antes tinham sido implementados e testados no âmbito do *update summarization* (por exemplo, o *DensityBased* e *T-LexReRank*). Também a utilização de modelos múltiplos heterogéneos e homogéneos não é muito comum na literatura relacionada com a tarefa

de *update summarization*. Nos modelos de aprendizagem supervisionada, tivemos oportunidade de propor alguns atributos novos, específicos para esta tarefa.

Os sistemas desenvolvidos, quando em comparação com outros que participaram nas respetivas conferências, obtêm resultados bastante promissores, todos acima das *baselines* e alguns bem próximo dos que alcançaram os lugares cimeiros em termos de medidas ROUGE.

Estes resultados são particularmente animadores se se tiver em consideração que os nossos sistemas são simplesmente extrativos, sem fazerem, por exemplo, qualquer tipo de modificação às frases dos textos originais. Isto para dizer que estamos convencidos que ainda há uma grande margem de progressão.

Também conseguimos responder às questões que inicialmente propusemos, se bem que a resposta nem sempre fosse aquela que estávamos à espera. Por exemplo, esperávamos que os modelos múltiplos conseguissem superar os modelos individuais, o que não se veio a verificar.

Esperamos que este trabalho seja um contributo relevante para os temas da sumarização e do *update summarization*.

Para nós foi e é importante, pelo que já fizemos, mas também pelas perspetivas que nos abre para o futuro.

Referências

Aggarwal, G., Sumbaly, R. e Sinha, S. (2009), "Update Summarization", Course Final Project, Standford University.

Baxendale, P. B. (1958), "Machine-Made Index for Technical Literature - an Experiment", *IBM Journal of Research Development*, Vol. 2, pp. 354-361.

Blake, C. (2006), "A comparison of document, sentence, and term event spaces", in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pp. 601-608, Association for Computational Linguistics, Sydney, Australia.

Bossard, A. e Rodrigues, C. (2011), "Combining a Multi-Document Update Summarization System –CBSEAS– with a Genetic Algorithm", in *Combinations of Intelligent Methods and Applications*, I. Hatzilygeroudis et al. (editors), Vol. 8, pp. 71-87, Springer Berlin Heidelberg.

Bouchet-Vallat, M. (2014), "Snowball stemmers based on the C libstemmer UTF-8 library (Package 'SnowballC')", http://cran.r-project.org/web/packages/SnowballC/ SnowballC.pdf.

Boudin, F. e Torres-Moreno, J.-M. (2009), "A Maximization-Minimization approach for update summarization", in *Recent Advances in Natural Language Processing V Current Issues in Linguistic Theory*, N. Nicolov et al. (editors), Vol. 309, pp. 143-154.

Brin, S. e Page, L. (1998), "The anatomy of a large-scale hypertextual Web search engine", in *Proceedings of the seventh international conference on World Wide Web 7*, pp. 107-117, Elsevier Science Publishers B. V., Brisbane, Australia.

Bysani, P., Bharat, V. e Varma, V. (2009), "Modeling Novelty and Feature Combination using Support Vector Regression for Update Summarization", in *7th International Conference on Natural Language Processing*, Macmillan Publishers, India.

Carbonell, J. e Goldstein, J. (1998), "The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries", in *Proceedings of the 21st Annual International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 335-336, Melbourne, Australia.

Carenini, G., Murray, G. e Ng, R. (2011), *Methods for Mining and Summarizing Text Conversations*: Morgan & Claypool Publishers.

Conroy, J., Schlesinger, J. D. e O'Leary, D. P. (2009), "CLASSY 2009: Summarization and Metrics", in *Proceedings of the text analysis conference (TAC)*.

Cordeiro, J. (2010), "Rule Induction for Sentence Reduction", PhD em Engenharia Informárica, Universidade da Beira Interior, Covilhã.

Dang, H. T. e Owczarzak, K. (2008), "Overview of the TAC 2008 Update Summarization Task", in *TAC 2008*, National Institute of Standards and Technology.

Dang, H. T. e Owczarzak, K. (2009), "Overview of the TAC 2009 Update Summarization Track", in *TAC 2009*, National Institute of Standards and Technology.

Das, D. e Martins, A. (2007), "A Survey on Automatic Text Summarization", Literature Survey for the Language and Statistics II course, Carnegie Mellon University.

Delort, J.-Y. e Alfonseca, E. (2012), "DualSum: a topic-model based approach for update summarization", in *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 214-223, Association for Computational Linguistics, Avignon, France.

Edmundson, H. P. (1969), "New Methods in Automatic Extracting", *Journal of the Association for Computing Machinery*, Vol. 16, Nº 2, pp. 264-285.

Erkan, G. e Radev, D. R. (2004), "LexRank: Graph-based Lexical Centrality as Salience in Text Summarization", *Journal of Articial Intelligence Research*, Vol. 22, pp. 475-479.

Feinerer, I. e Hornik, K. (2014), "Text Mining Package (Package 'tm')", http://cran.r-project.org/web/packages/tm/tm.pdf.

Foong, O. M., Oxley, A. e Sukaiman, S. (2010), "Challenges and Trends of Automatic Text Summarization", *International Journal of Information and Telecommunication Technology*, Vol. 1, Nº 1, pp. 34-39.

Gama, J., Carvalho, A., Faceli, K., Lorena, A. e Oliveira, M. (2012), *Extração de Conhecimento de Dados - Data Mining*, Lisboa: Edições Sílabo.

Gupta, V. e Lehal, G. S. (2010), "A Survey of Text Summarization Extractive Techniques", *Journal of Emerging Technologies in Web Intelligence*, Vol. 2, Nº 3, pp. 258-268.

Hahn, U. e Mani, I. (2000), "The challenges of automatic summarization", *IEEE Computer*, Vol. 33, Nº 11, pp. 29-36.

Hobson, S. F. (2007), "Text Summarization Evaluation: Correlating Human Performance on an Extrinsic Task with Automatic Intrinsic Metrics", PhD Doctor of Philosophy, University of Maryland.

Hornik, K. (2014), "Basic classes and methods for Natural Language Processing (Package 'NLP')", http://cran.r-project.org/web/packages/NLP/NLP.pdf.

Hovy, E. e 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*, pp. 197-214, Association for Computational Linguistics, Baltimore, Maryland.

Jinha, A. E. (2010), "Article 50 million: an estimate of the number of scholarly articles in existence", *Learned Publishing*, Vol. 23, Nº 3, pp. 258-263.

Jones, K. S. (1972), "A statistical interpretation of term specificity and its application in retrieval", *Journal of Documentation*, Vol. 28, Nº 1, pp. 11-21.

Kogilavani, A. e Balasubramanie, P. (2012), "Update Summary Generation based on Semantically Adapted Vector Space Model", *International Journal of Computer Applications*, Vol. 42, Nº 16, pp. 32-39.

Kumar, Y. J. e Salim, N. (2012), "Automatic Multi Document Summarization Approaches", *Journal of Computer Science*, Vol. 8, Nº 1, pp. 133-140.

Kupiec, J., Pedersen, J. e Chen, F. (1995), "A trainable document summarizer", in *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 68-73, ACM, Seattle, Washington, USA.

Lapalme, G. P., Nerima, L. e Wehrli, E. (2008), "A symbolic summarizer for the update task of TAC 2008", in *Proceedings of the First Text Analysis Conference*, Maryland, USA.

Larsen, P. O. e Ins, M. v. (2010), "The rate of growth in scientific publication and the decline in coverage provided by Science Citation Index", *Scientometrics*, Vol. 84, Nº 3, pp. 575-603.

Li, W., Wei, F., Lu, Q. e He, Y. (2008), "PNR^2: ranking sentences with positive and negative reinforcement for query-oriented update summarization", in *Proceedings of the 22nd International Conference on Computational Linguistics*, Vol. 1, pp. 489-496, Association for Computational Linguistics, Manchester, United Kingdom.

Li, X., Du, L. e Shen, Y.-D. (2013), "Update Summarization via Graph-Based Sentence Ranking", *IEEE Transactions on Knowledge anda Data Engineering*, Vol. 25, Nº 5, pp. 1162-1174.

Liaw, A. e Wiener, M. (2014), "Breiman and Cutler's random forests for classification and regression (Package 'randomForest')", http://cran.r-project.org/web/packages/ randomForest/ randomForest.pdf.

Limas, M. C., Mere, J. B. O., Marcos, A. G. e Ascacibar, F. J. M. d. P. (2014), "A MORE flexible neural network package (Package 'AMORE')", http://cran.r-project.org/web/packages/AMORE/AMORE.pdf.

Lin, C.-Y. (1999), "Training a Selection Function for Extraction", in *Proceedings of the Eighteenth Annual International ACM Conference on Information and Knowledge Management*, Kansas City, Kansas.

Lin, C.-Y. (2004a), "Looking for a Few Good Metrics: Automatic Summarization Evaluation - How Many Samples Are Enough?", in *Proceedings of NTCIR Workshop*, Tokyo, Japan.

Lin, C.-Y. (2004b), "Rouge: A package for automatic evaluation of summaries", in *Proceedings of the ACL-04Workshop: Text Summarization Branches Out*, pp. 74-81, Barcelona, Spain.

Lin, C.-Y. (2012), "ROUGE: Recall-Oriented Understudy of Gisting Evaluation - A software package for automated evaluation of summaries", http://www.berouge.com/Pages/default.aspx.

Lin, C.-Y. e Hovy, E. (1997), "Identifying topics by position", in *Proceedings of the fifth conference on Applied natural language processing*, pp. 283-290, Morgan Kaufmann Publishers Inc.

Lin, C.-Y. e Hovy, E. (2003), "Automatic Evaluation of Summaries Using N-gram Co-Occurrence Statistics ", in *Proceedings of the Human Technology Conference*, Edmonton, Canada.

Louis, A. e Nenkova, A. (2013), "Automatically Assessing Machine Summary ContentWithout a Gold Standard", *Computational Linguistics*, Vol. 39, Nº 2, pp. 267-300.

Luhn, H. P. (1958), "The automatic creation of literature abstracts", *IBM Journal of Research Development*, Vol. 2, Nº 2, pp. 159-165.

Manning, C. D., Raghavan, P. e Schütze, H. (2008), *Introduction to Information Retrieval*: Cambridge University Press.

Meyer, D., Dimitriadou, E., Hornik, K. e Weingessel, A. (2014), "Misc Functions of the Department of Statistics (e1071), TU Wien (Package 'e1071')", http://cran.r-project.org/web/packages/e1071/e1071.pdf.

Mihalcea, R. (2004), "Graph-based ranking algorithms for sentence extraction, applied to text summarization", in *Proceedings of the ACL 2004 on Interactive poster*

and demonstration sessions, Association for Computational Linguistics, Barcelona, Spain.

Mihalcea, R. e Tarau, P. (2004), "Textrank: Bringing order into texts", in *Proceedings of EMNLP 2004* (D. Lin et al., eds.), Association for Computational Linguistics, Barcelona, Spain.

Nenkova, A. e Passoneau, R. (2004), "Evaluating content selection in summarization: The pyramid method", in *Proceedings of HLT-NAACL*, pp. 145-152, Association for Computational Linguistics, Boston.

Nenkova, A. e Vanderwende, L. (2005), "The impact of frequency on summarization", Technical report, Microsoft Research.

Nomoto, T. e Matsumoto, Y. (2001), "A New Approach to Unsupervised Text Summarization", in *SIGIR'01*, pp. 26-34, ACM, New Orleans, USA.

Otterbacher, J., Erkan, G. e Radev, D. R. (2005), "Using Random Walks for Question-focused Sentence Retrieval", in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 915-922, Association for Computational Linguistics, Vancouver.

Otterbacher, J., Erkan, G. e Radev, R. (2009), "Biased lexrank:passage retrieval using random walks with question-based priors", in *Information Processing and Management*, Vol. 45, pp. 42-54.

Patil, K. e Brazdil, P. (2007), "Sumgraph: text summarization using centrality in the pathfinder network", *IADIS International Journal on Computer Science and Information Systems*, Vol. 2, pp. 18-32.

Princeton University (2010), "About WordNet".

Radev, D. R., Jing, H., Stys, M. e Tam, D. (2004), "Centroid-based summarization of multiple documents", *Information Processing and Management*, Vol. 40, pp. 919-938.

Rath, G. J., Resnick, A. e Savage, T. R. (1961), "The formation of abstracts by the selection of sentences. Part I. Sentence selection by men and machines", *American Documentation*, Vol. 12, Nº 2, pp. 139-141.

Robertson, S. (2004), "Understanding inverse document frequency: On theoretical arguments for IDF", *Journal of Documentation*, Vol. 60, Nº 5, pp. 503-520.

Salton, G., Wong, A. e Yang, C. S. (1975), "A vector space model for automatic indexing", *Commun. ACM*, Vol. 18, Nº 11, pp. 613-620.

Scirus (2014), "Scirus - About us", http://www.scirus.com/srsapp/aboutus/#range, acedido em Janeiro, 2014.

Seki, Y., Eguchi, K. e Kando, N. (2005), "Multidocument viewpoint summarization focused on facts, opinion and knowledge", in *Computing Attitude and Affect in Text: Theory and Applications*, J. G. Shanahan et al. (editors), Vol. 20, pp. 317-336, Springer-Verlag, New York.

Soboroff, I. e Harman, D. (2005), "Novelty Detection: The TREC Experience", in *Proceedings of Human Language Technoloy Conference and Conference on Empirical Methods in Natural Language Processing*, pp. 105-112, Association for Computational Linguistics.

Team, R. C. (2014), "R: A Language and Environment for Statistical Computing", http://www.R-project.org/.

Valizadeh, M. e Brazdil, P. (2013), "Density-Based Graph Model for Multi-Document Summarization", in *EPIA2013*, Açores, Portugal.

Valizadeh, M. e Brazdil, P. (2014a), "Exploring Actor-Object Relationships for Query-focused Multi-Document Summarization", *Soft Computing*.

Valizadeh, M. e Brazdil, P. (2014b), "User-Based Models for Query-focused Multi-Document Summarization", *Unpublished manuscript*.

van Rijsbergen, C. J., Robertson, S. E. e Porter, M. F. (1980), "New models in probabilistic information retrieval".

Wan, X. (2007), "TimedTextRank: Adding the Temporal Dimension to Multi-Document Summarization", in *SIGIR'07*, pp. 867-868, ACM, Amsterdam, The Netherlands.

Wang, D. e Li, T. (2010), "Document Update Summarization Using Incremental Hierarchical Clustering", in *CIKM'10*, pp. 279-287, ACM, Toronto, Canada.

Web of Knowledge (2014), "The Citaton Connection - Real Facts", http://wokinfo.com/citationconnection/, acedido em Janeiro, 2014.

Wong, K.-F., Wu, M. e Li, W. (2008), "Extractive summarization using supervised and semi-supervised learning", in *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, pp. 985-992, Association for Computational Linguistics, Manchester, United Kingdom.

ANEXOS

Anexo A – Tabelas com os Resultados ROUGE

Nas tabelas seguintes são apresentados os resultados das experiências realizadas em todas as tarefas, com todos os conjuntos de dados. Os sistemas marcados com um asterisco são sistemas implementados no âmbito desta tese. Na posição, os sistemas de comparação *lower bounds* estão assinalados com LB e os *upper bounds* com UB. Todos os sistemas de comparação, implementados por nós ou fornecidos nos conjuntos de textos, estão em itálico. Os modelos explorados e desenvolvidos ao longo da tese estão a negrito, sendo que os modelos de aprendizagem supervisionada têm o prefixo *Sup*.

TAC 2007: Sumarização

	Pos	Sistema	ROUGE-2	ROUGE-SU4
*	UB	*Oráculo 2*	*0,17276*	*0,18513*
*	UB	*Oráculo 1*	*0,15648*	*0,17975*
	1	S40	0,12724	0,15642
	UB	*Humanos*	*0,11896*	*0,15259*
	2	S55	0,11519	0,13932
	3	S48	0,11011	0,14157
*	4	**Sup-RN**	**0,10936**	**0,13467**
*	5	**Sup-EnsHetero**	**0,10602**	**0,13307**
*	6	**Sup-RF**	**0,10531**	**0,13508**
	7	S47	0,10245	0,13358
*	8	**DensityBased**	**0,10094**	**0,12604**
	9	S51	0,09966	0,12851
	10	S42	0,09803	0,12854
	11	S45	0,09743	0,12582
	12	S44	0,09702	0,13480
*	13	**Sup-SVM**	**0,09690**	**0,12356**
*	14	**PNR2**	**0,09616**	**0,12504**
	15	S36	0,09554	0,13216
*	16	**T-LexRank**	**0,09526**	**0,12002**
*	17	**T-LexReRank**	**0,09526**	**0,12002**
	18	S53	0,09451	0,12424
*	19	**Sup-EnsHomo**	**0,09370**	**0,11910**
	20	S46	0,09346	0,12645
	21	S43	0,09129	0,12640
	22	S38	0,09014	0,12308
	23	S49	0,08781	0,12301
	LB	*S58*	*0,08679*	*0,11611*
	24	S41	0,08334	0,11423
	25	S52	0,08039	0,11536
*	LB	*Topo*	*0,06199*	*0,09400*
	26	S56	0,06007	0,09689
	27	S39	0,05893	0,09743
	28	S37	0,05222	0,08146
	LB	*S35*	*0,04956*	*0,08242*
	29	S50	0,04933	0,08447
*	LB	*Aleatório*	*0,04435*	*0,08298*
	30	S54	0,03926	0,08494
	31	S57	0,03563	0,06876

TAC 2007: *Update Summarization*

	Pos	Sistema	ROUGE-2	ROUGE-SU4
*	UB	*Oráculo 2*	*0,19299*	*0,20740*
*	UB	*Oráculo 1*	*0,16594*	*0,19249*
	UB	*Humanos*	*0,13498*	*0,16436*
	1	S40	0,10550	0,13679
	2	S45	0,09765	0,13727
	3	S44	0,09722	0,13911
	4	S46	0,09614	0,13531
*	5	**Sup-RN**	**0,09299**	**0,12655**
	6	S55	0,09270	0,13383
	LB	*S58*	*0,09107*	*0,12998*
*	7	**Sup-EnsHomo**	**0,09104**	**0,12576**
	8	S47	0,08999	0,13001
	9	S38	0,08989	0,13341
	10	S36	0,08683	0,12449
*	11	**T-LexRank**	**0,08656**	**0,12451**
*	12	**T-LexReRank**	**0,08595**	**0,12569**
*	13	**DensityBased**	**0,08482**	**0,12525**
*	14	**Sup-EnsHetero**	**0,08231**	**0,12092**
	15	S52	0,08222	0,11576
*	16	**Sup-RF**	**0,08064**	**0,11830**
*	17	**PNR2**	**0,07968**	**0,12115**
	18	S48	0,07859	0,11334
*	19	**Sup-SVM**	**0,07717**	**0,11879**
	20	S49	0,07018	0,10910
	21	S51	0,07008	0,11142
	22	S53	0,06696	0,10884
	23	S54	0,06547	0,10906
	24	S56	0,06413	0,10482
	25	S42	0,06042	0,10505
*	LB	*Topo*	*0,05916*	*0,09557*
	26	S43	0,05891	0,10527
	27	S50	0,05832	0,09516
	28	S39	0,05125	0,09546
	29	S41	0,05044	0,08862
*	LB	*Aleatório*	*0,04933*	*0,09058*
	LB	*S35*	*0,04669*	*0,08665*
	30	S57	0,04192	0,08173
	31	S37	0,03832	0,07775

TAC 2007: Global

	Pos	Sistema	ROUGE-2	ROUGE-SU4
*	UB	*Oráculo 2*	*0,18049*	*0,19766*
*	UB	*Oráculo 1*	*0,15982*	*0,18886*
	UB	*Humanos*	*0,12757*	*0,15957*
	1	S40	0,11207	0,14299
	2	S55	0,09931	0,13529
*	3	**Sup-RN**	**0,09785**	**0,12876**
	4	S45	0,09657	0,13272
	5	S44	0,09564	0,13701
	6	S46	0,09376	0,13124
	7	S47	0,09372	0,13069
*	8	**Sup-EnsHomo**	**0,09097**	**0,12280**
*	9	**DensityBased**	**0,08964**	**0,12529**
*	10	**Sup-EnsHetero**	**0,08942**	**0,12428**
*	11	**T-LexRank**	**0,08906**	**0,12295**
	12	S38	0,08869	0,12921
*	13	**T-LexReRank**	**0,08863**	**0,12358**
	14	S48	0,08836	0,12242
	15	S36	0,08831	0,12620
*	16	**Sup-RF**	**0,08808**	**0,12332**
	LB	*S58*	*0,08782*	*0,12433*
*	17	**PNR2**	**0,08497**	**0,12231**
*	18	**Sup-SVM**	**0,08266**	**0,11933**
	19	S52	0,08085	0,11546
	20	S51	0,07993	0,11750
	21	S49	0,07541	0,11339
	22	S53	0,07452	0,11263
	23	S42	0,07181	0,11201
	24	S43	0,06850	0,11170
	25	S56	0,06259	0,10192
*	LB	*Topo*	*0,06200*	*0,09808*
	26	S41	0,06151	0,09771
	27	S54	0,05606	0,10034
	28	S50	0,05348	0,09036
	29	S39	0,05342	0,09588
*	LB	*Aleatório*	*0,04748*	*0,08833*
	LB	*S35*	*0,04654*	*0,08388*
	30	S37	0,04251	0,07870
	31	S57	0,03841	0,07623

TAC 2008: Sumarização

Pos	Sistema	ROUGE-2	ROUGE-SU4
* UB	*Oráculo 2*	*0,16314*	*0,18825*
* UB	*Oráculo 1*	*0,14447*	*0,17515*
UB	*Humanos*	*0,11606*	*0,15283*
1	S43	0,11089	0,14262
2	S13	0,10978	0,13929
3	S60	0,10305	0,14163
4	S37	0,10275	0,14242
5	S6	0,10073	0,13953
6	S2	0,09952	0,13639
7	S64	0,09862	0,12887
* 8	**Sup-EnsHomo**	**0,09562**	**0,12954**
* 9	**T-LexRank**	**0,09518**	**0,12920**
* 10	**T-LexReRank**	**0,09518**	**0,12920**
* 11	**DensityBased**	**0,09465**	**0,12872**
12	S45	0,09435	0,13324
* 13	**Sup-RN**	**0,09432**	**0,12794**
14	S12	0,09414	0,12659
* 15	**Sup-SVM**	**0,09351**	**0,12567**
16	S14	0,09347	0,12855
17	S65	0,09347	0,12855
18	S49	0,09265	0,13045
19	S42	0,09193	0,12540
20	S63	0,09193	0,12540
21	S23	0,09093	0,12599
22	S50	0,08978	0,12207
* 23	**Sup-RFs**	**0,08978**	**0,12198**
24	S44	0,08970	0,12690
* 25	**Sup-EnsHetero**	**0,08909**	**0,12301**
26	S25	0,08811	0,12353
27	S51	0,08764	0,12341
28	S11	0,08747	0,12234
29	S41	0,08725	0,12269
30	S69	0,08724	0,12361
31	S70	0,08513	0,12030
32	S54	0,08501	0,12108
33	S26	0,08490	0,12013
34	S62	0,08406	0,11839
35	S52	0,08398	0,11965
36	S22	0,08378	0,12287
37	S58	0,08357	0,12312
38	S17	0,08342	0,11857
39	S35	0,08229	0,12244
40	S48	0,08216	0,11753
41	S24	0,08170	0,11706
42	S68	0,08170	0,11706
43	S19	0,08108	0,11736
44	S46	0,08076	0,11925
45	S15	0,08069	0,11744
46	S3	0,08022	0,12155

Pos	Sistema	ROUGE-2	ROUGE-SU4
47	S10	0,07952	0,11833
48	S61	0,07952	0,11833
49	S34	0,07901	0,11565
50	S1	0,07893	0,11544
51	S36	0,07790	0,11347
52	S30	0,07777	0,10704
53	S67	0,07632	0,11284
54	S27	0,07430	0,11011
55	S20	0,07260	0,11442
56	S7	0,07237	0,10773
* 57	**PNR2**	**0,07214**	**0,10810**
58	S66	0,07113	0,11246
59	S16	0,07082	0,11205
60	S40	0,06936	0,10806
61	S4	0,06927	0,10543
62	S53	0,06843	0,10390
63	S21	0,06840	0,11023
64	S57	0,06791	0,10009
65	S29	0,06730	0,10470
66	S33	0,06729	0,10218
67	S71	0,06728	0,10290
68	S32	0,06709	0,10478
69	S8	0,06669	0,10530
70	S28	0,06534	0,09897
71	S56	0,06503	0,10128
72	S38	0,06450	0,10371
73	S5	0,06401	0,10210
* LB	*Topo*	*0,06346*	*0,09926*
74	S55	0,06210	0,10101
LB	*S0*	*0,05762*	*0,09232*
75	S47	0,05758	0,09334
76	S59	0,05542	0,09415
77	S31	0,05222	0,09231
78	S39	0,05039	0,08610
* LB	*Aleatório*	*0,04360*	*0,08554*
79	S9	0,04225	0,08252
80	S18	0,03897	0,07338

TAC 2008: *Update Summarization*

Pos	Sistema	ROUGE-2	ROUGE-SU4
* UB	*Oráculo 2*	*0,17804*	*0,19588*
* UB	*Oráculo 1*	*0,14686*	*0,17677*
UB	*Humanos*	*0,11482*	*0,14860*
1	S14	0,10015	0,13587
2	S65	0,09579	0,13291
3	S43	0,09574	0,12937
* 4	**Sup-RFs**	**0,09548**	**0,13086**
* 5	**Sup-RN**	**0,09357**	**0,12808**
6	S2	0,09146	0,13087
* 7	**Sup-EnsHetero**	**0,09141**	**0,12754**
8	S49	0,09080	0,13084
* 9	**Sup-SVM**	**0,08928**	**0,12732**
10	S44	0,08910	0,12960
11	S62	0,08908	0,12539
* 12	**Sup-EnsHomo**	**0,08904**	**0,12610**
13	S23	0,08881	0,12473
14	S11	0,08805	0,12588
15	S69	0,08754	0,12574
16	S13	0,08692	0,12475
17	S64	0,08491	0,12000
18	S60	0,08441	0,12890
19	S37	0,08438	0,12820
* 20	**T-LexReRank**	**0,08383**	**0,12234**
* 21	**T-LexRank**	**0,08196**	**0,12215**
22	S1	0,08165	0,12068
23	S34	0,08145	0,11780
24	S25	0,08091	0,12022
25	S24	0,08086	0,11932
* 26	**DensityBased**	**0,08070**	**0,12000**
27	S68	0,07973	0,11837
28	S41	0,07826	0,11751
29	S45	0,07826	0,11908
30	S19	0,07780	0,11804
31	S50	0,07752	0,11732
32	S6	0,07741	0,12311
33	S51	0,07705	0,11849
34	S52	0,07428	0,11427
35	S20	0,07301	0,11337
36	S29	0,07296	0,11205
37	S48	0,07296	0,11254
38	S63	0,07264	0,11173
39	S15	0,07166	0,11276
40	S26	0,07159	0,10996
41	S12	0,07068	0,10681
42	S61	0,06963	0,11015
43	S17	0,06930	0,11313
44	S67	0,06887	0,11023
45	S36	0,06876	0,10879
46	S4	0,06827	0,10700

Pos	Sistema	ROUGE-2	ROUGE-SU4
47	S42	0,06814	0,10637
48	S10	0,06813	0,10957
49	S16	0,06749	0,10928
50	S22	0,06711	0,11085
51	S46	0,06703	0,11132
* 52	**PNR2**	**0,06692**	**0,10693**
53	S58	0,06656	0,10630
54	S35	0,06652	0,10747
55	S21	0,06563	0,10603
56	S32	0,06528	0,10592
57	S40	0,06427	0,10688
58	S27	0,06292	0,10281
59	S55	0,06280	0,10259
60	S3	0,06150	0,10301
61	S66	0,05974	0,10272
LB	*S0*	*0,05961*	*0,09357*
62	S70	0,05956	0,09925
63	S54	0,05935	0,09807
64	S5	0,05668	0,10085
65	S57	0,05560	0,09179
66	S59	0,05465	0,09609
67	S53	0,05449	0,09464
68	S33	0,05364	0,09307
* LB	*Topo*	*0,05354*	*0,09139*
69	S8	0,05143	0,09472
70	S38	0,04889	0,09059
71	S31	0,04798	0,08939
72	S28	0,04615	0,08275
* LB	*Aleatório*	*0,04534*	*0,08742*
73	S47	0,03579	0,06408
74	S56	0,03445	0,06815
75	S39	0,03420	0,07310
76	S30	0,03395	0,06884
77	S71	0,03215	0,06636
78	S18	0,02798	0,05681
79	S9	0,02712	0,07042
80	S7	0,01760	0,04840

TAC 2008: Global

Pos	Sistema	ROUGE-2	ROUGE-SU4
* UB	*Oráculo 2*	*0,17112*	*0,19254*
* UB	*Oráculo 1*	*0,14626*	*0,17648*
UB	*Humanos*	*0,11635*	*0,15148*
1	S43	0,10361	0,13622
2	S13	0,09862	0,13224
3	S14	0,09717	0,13247
4	S2	0,09580	0,13394
5	S65	0,09497	0,13100
* 6	**Sup-RN**	**0,09422**	**0,12824**
7	S60	0,09406	0,13554
8	S37	0,09391	0,13561
* 9	**Sup-RFs**	**0,09287**	**0,12662**
* 10	**Sup-EnsHomo**	**0,09280**	**0,12820**
11	S49	0,09212	0,13101
12	S64	0,09208	0,12467
* 13	**Sup-SVM**	**0,09173**	**0,12679**
* 14	**Sup-EnsHetero**	**0,09048**	**0,12548**
15	S23	0,09022	0,12566
* 16	**T-LexReRank**	**0,08983**	**0,12608**
17	S44	0,08970	0,12849
18	S6	0,08933	0,13150
* 19	**T-LexRank**	**0,08886**	**0,12594**
20	S11	0,08808	0,12440
* 21	**DensityBased**	**0,08794**	**0,12461**
22	S69	0,08774	0,12499
23	S62	0,08685	0,12211
24	S45	0,08665	0,12648
25	S25	0,08470	0,12200
26	S50	0,08392	0,11990
27	S41	0,08305	0,12035
28	S12	0,08261	0,11694
29	S51	0,08255	0,12109
30	S63	0,08246	0,11872
31	S24	0,08158	0,11844
32	S68	0,08098	0,11794
33	S1	0,08060	0,11829
34	S34	0,08050	0,11689
35	S42	0,08019	0,11602
36	S19	0,07968	0,11788
37	S52	0,07944	0,11725
38	S26	0,07854	0,11532
39	S48	0,07773	0,11516
40	S17	0,07666	0,11607
41	S15	0,07638	0,11522
42	S22	0,07571	0,11703
43	S58	0,07552	0,11511
44	S35	0,07478	0,11529
45	S61	0,07477	0,11435
46	S46	0,07416	0,11552

Pos	Sistema	ROUGE-2	ROUGE-SU4
47	S10	0,07403	0,11406
48	S36	0,07354	0,11134
49	S20	0,07303	0,11408
50	S67	0,07275	0,11165
51	S70	0,07246	0,10989
52	S54	0,07231	0,10971
53	S3	0,07125	0,11264
54	S29	0,07024	0,10846
* 55	**PNR2**	**0,06972**	**0,10770**
56	S16	0,06923	0,11072
57	S4	0,06905	0,10643
58	S27	0,06859	0,10642
59	S21	0,06721	0,10829
60	S40	0,06694	0,10755
61	S32	0,06637	0,10558
62	S66	0,06563	0,10778
63	S55	0,06260	0,10196
64	S57	0,06194	0,09607
65	S53	0,06168	0,09946
66	S33	0,06066	0,09780
67	S5	0,06047	0,10166
68	S8	0,05914	0,10010
LB	*S0*	*0,05879*	*0,09311*
* LB	*Topo*	*0,05871*	*0,09550*
69	S38	0,05684	0,09730
70	S30	0,05599	0,08805
71	S28	0,05580	0,09098
72	S59	0,05531	0,09532
73	S31	0,05038	0,09107
74	S56	0,04989	0,08487
75	S71	0,04984	0,08475
76	S47	0,04675	0,07884
77	S7	0,04517	0,07827
* LB	*Aleatório*	*0,04465*	*0,08664*
78	S39	0,04239	0,07969
79	S9	0,03477	0,07652
80	S18	0,03354	0,06514

TAC 2009: Sumarização

Pos		Sistema	ROUGE-2	ROUGE-SU4
	UB	*S2*	*0,33082*	*0,34329*
*	UB	*Oráculo 2*	*0,17888*	*0,19739*
*	UB	*Oráculo 1*	*0,14709*	*0,17648*
	UB	*Humanos*	*0,12553*	*0,16285*
	1	S34	0,12102	0,14972
	2	S40	0,12049	0,15062
	3	S35	0,10751	0,14407
	UB	*S3*	*0,10573*	*0,13758*
	4	S45	0,10566	0,13916
	5	S51	0,10414	0,14105
	6	S23	0,10353	0,13889
	7	S4	0,10265	0,13788
	8	S42	0,10247	0,13845
	9	S10	0,10123	0,13480
	10	S32	0,10095	0,13726
	11	S55	0,10095	0,13726
	12	S54	0,09964	0,13770
	13	S24	0,09747	0,13217
*	14	**Sup-RN**	**0,09735**	**0,13089**
*	15	**Sup-RFs**	**0,09649**	**0,13128**
*	16	**DensityBased**	**0,09634**	**0,13480**
	17	S6	0,09549	0,13494
*	18	**Sup-EnsHetero**	**0,09526**	**0,12877**
*	19	**T-LexRank**	**0,09523**	**0,13062**
*	20	**T-LexReRank**	**0,09523**	**0,13062**
	21	S11	0,09498	0,12947
	22	S41	0,09477	0,13349
	23	S12	0,09442	0,13788
	24	S53	0,09411	0,12924
	25	S8	0,09375	0,12818
*	26	**Sup-EnsHomo**	**0,09361**	**0,12925**
	27	S49	0,09334	0,12895
	28	S19	0,09326	0,12792
	29	S27	0,09267	0,13004
	30	S38	0,09256	0,12938
*	31	**Sup-SVM**	**0,09014**	**0,12539**
	32	S7	0,08986	0,12731
	33	S14	0,08791	0,12351
	34	S37	0,08771	0,12741
	35	S15	0,08748	0,12579
	36	S48	0,08738	0,12641
	37	S36	0,08734	0,12569
	38	S13	0,08695	0,12304
	39	S50	0,08695	0,12304
	40	S20	0,08508	0,11973
	41	S26	0,08437	0,12217
	42	S33	0,08387	0,12232
	43	S52	0,08279	0,11757
	44	S22	0,08263	0,12010

Pos		Sistema	ROUGE-2	ROUGE-SU4
	45	S44	0,08125	0,12065
	46	S18	0,07967	0,11996
	47	S5	0,07872	0,11442
	48	S21	0,07845	0,11695
*	49	**PNR2**	**0,07745**	**0,11430**
	50	S29	0,07627	0,11588
	51	S30	0,07527	0,11373
	52	S16	0,07466	0,11381
	53	S43	0,07294	0,10691
*	LB	*Topo*	*0,06822*	*0,10516*
	LB	*S1*	*0,06268*	*0,09853*
	54	S25	0,06095	0,09620
	55	S46	0,05293	0,09116
	56	S9	0,05183	0,09075
	57	S31	0,04811	0,08967
*	LB	*Aleatório*	*0,04793*	*0,08907*
	58	S47	0,04302	0,06422
	59	S39	0,03894	0,07726
	60	S17	0,03451	0,05885
	61	S28	0,02814	0,06119

TAC 2009: *Update Summarization*

	Pos	Sistema	ROUGE-2	ROUGE-SU4
	UB	*S2*	*0.31866*	*0.33610*
*	UB	*Oráculo 2*	*0.18121*	*0.20038*
*	UB	*Oráculo 1*	*0.13516*	*0.16965*
	UB	*Humanos*	*0.10643*	*0.14795*
	1	S34	0.10433	0.13862
	2	S40	0.10403	0.13944
	3	S35	0.10070	0.13807
	UB	*S3*	*0.09780*	*0.13576*
	4	S24	0.09621	0.13493
	5	S51	0.09481	0.13565
*	6	**Sup-RN**	**0.09180**	**0.13055**
	7	S23	0.09142	0.13033
	8	S4	0.09114	0.13281
*	9	**Sup-SVM**	**0.08876**	**0.13023**
*	10	**Sup-RFs**	**0.08787**	**0.12680**
	11	S38	0.08780	0.12464
*	12	**Sup-EnsHetero**	**0.08664**	**0.12819**
	13	S12	0.08586	0.13001
*	14	**Sup-EnsHomo**	**0.08570**	**0.12560**
	15	S45	0.08485	0.12528
*	16	**DensityBased**	**0.08457**	**0.12470**
	17	S53	0.08444	0.12377
	18	S7	0.08413	0.12431
	19	S11	0.08308	0.12037
	20	S36	0.08268	0.12410
	21	S8	0.08106	0.11843
	22	S42	0.08096	0.12208
	23	S41	0.07946	0.11923
	24	S6	0.07942	0.12201
*	25	**T-LexReRank**	**0.07871**	**0.12127**
	26	S10	0.07809	0.11886
	27	S22	0.07747	0.11590
	28	S15	0.07714	0.11994
	29	S19	0.07654	0.11547
	30	S27	0.07647	0.11872
	31	S54	0.07544	0.11984
	32	S49	0.07540	0.11761
	33	S5	0.07477	0.11255
*	34	**T-LexRank**	**0.07438**	**0.11566**
	35	S26	0.07417	0.11230
	36	S55	0.07387	0.11674
	37	S21	0.07282	0.11729
	38	S50	0.07246	0.11356
	39	S13	0.07222	0.11301
	40	S20	0.07057	0.11063
	41	S48	0.07038	0.11150
	42	S33	0.06979	0.11581
	43	S52	0.06967	0.10707
	44	S29	0.06891	0.11188
	45	S16	0.06856	0.11135
	46	S37	0.06851	0.11331
	47	S14	0.06351	0.10361
	48	S32	0.06236	0.10388
*	49	**PNR2**	**0.06127**	**0.10264**
	50	S43	0.06113	0.10093
*	LB	*Topo*	*0.06048*	*0.09942*
	51	S30	0.05944	0.10162
	52	S25	0.05322	0.09532
	53	S18	0.05223	0.09617
	LB	*S1*	*0.05043*	*0.09006*
	54	S31	0.04860	0.08998
	55	S46	0.04727	0.09002
*	LB	*Aleatório*	*0.04651*	*0.09037*
	56	S9	0.04281	0.08462
	57	S44	0.04210	0.08573
	58	S47	0.03947	0.06567
	59	S39	0.03793	0.07846
	60	S17	0.03053	0.05703
	61	S28	0.02573	0.06506

TAC 2009: Global

	Pos	Sistema	ROUGE-2	ROUGE-SU4
	UB	*S2*	*0,32496*	*0,33993*
*	UB	*Oráculo 2*	*0,18016*	*0,19912*
*	UB	*Oráculo 1*	*0,14116*	*0,17320*
	UB	*Humanos*	*0,11636*	*0,23742*
	1	S34	0,11289	0,14440
	2	S40	0,11244	0,14523
	3	S35	0,10433	0,14124
	UB	*S3*	*0,10181*	*0,13679*
	4	S51	0,09984	0,13863
	5	S23	0,09756	0,13467
	6	S4	0,09701	0,13549
	7	S24	0,09691	0,13365
	8	S45	0,09554	0,13248
*	9	**Sup-RN**	**0,09496**	**0,13106**
*	10	**Sup-RFs**	**0,09260**	**0,12937**
	11	S42	0,09183	0,13037
*	12	**Sup-EnsHetero**	**0,09125**	**0,12870**
*	13	**DensityBased**	**0,09073**	**0,13001**
	14	S12	0,09047	0,13422
	15	S38	0,09030	0,12717
	16	S10	0,08997	0,12712
*	17	**Sup-EnsHomo**	**0,08993**	**0,12767**
*	18	**Sup-SVM**	**0,08966**	**0,12798**
	19	S53	0,08938	0,12665
	20	S11	0,08935	0,12522
	21	S54	0,08774	0,12897
	22	S6	0,08769	0,12867
	23	S8	0,08762	0,12353
	24	S55	0,08759	0,12715
	25	S7	0,08742	0,12621
	26	S41	0,08740	0,12659
*	27	**T-LexReRank**	**0,08723**	**0,12617**
	28	S19	0,08528	0,12204
	29	S36	0,08514	0,12502
*	30	**T-LexRank**	**0,08512**	**0,12340**
	31	S27	0,08471	0,12449
	32	S49	0,08454	0,12348
	33	S15	0,08227	0,12285
	34	S32	0,08190	0,12077
	35	S22	0,08031	0,11823
	36	S13	0,07973	0,11820
	37	S50	0,07959	0,11834
	38	S26	0,07947	0,11739
	39	S48	0,07903	0,11914
	40	S37	0,07838	0,12061
	41	S20	0,07801	0,11535
	42	S5	0,07705	0,11376
	43	S33	0,07702	0,11929
	44	S52	0,07643	0,11247

	Pos	Sistema	ROUGE-2	ROUGE-SU4
	45	S21	0,07580	0,11724
	46	S14	0,07577	0,11366
	47	S29	0,07266	0,11397
	48	S16	0,07166	0,11264
*	49	**PNR2**	**0,06962**	**0,10859**
	50	S30	0,06764	0,10803
	51	S43	0,06696	0,10386
	52	S18	0,06610	0,10821
*	LB	*Topo*	*0,06457*	*0,10251*
	53	S44	0,06198	0,10348
	54	S25	0,05721	0,09592
	LB	*S1*	*0,05675*	*0,09446*
	55	S46	0,05026	0,09074
	56	S31	0,04841	0,08990
*	LB	*Aleatório*	*0,04735*	*0,08987*
	57	S9	0,04735	0,08768
	58	S47	0,04116	0,06480
	59	S39	0,03858	0,07803
	60	S17	0,03266	0,05800
	61	S28	0,02706	0,06320

Anexo B – Exemplos de Sumários

Sumarização

Humanos

In August 2003, former Liberian President, Charles Taylor, accepted asylum in Nigeria from 17 counts of war crimes in Sierra Leone.
He has been accused of violating the asylum agreement by harboring Al-Qaeda terrorists, interfering with Liberian presidential elections and an assassination attempt on the Guinea President.
The U.N.-backed Special Court for Sierra Leone is pressuring the Nigerian to send Taylor home to Liberia.
U.N. peacekeepers would transfer him to Sierra Leone for prosecution.
Reversing an earlier statement, Liberia's new President states that Taylor should go directly to Sierra Leone from Nigeria since he has not been indicted in Liberia.

Redes Neuronais

Liberia's Taylor a "menace" to west Africa: Guinean minister.
Liberia's former president Charles Taylor, accused of war crimes for his role in Sierra Leone's civil war, is a "menace to west Africa," Guinea's territorial administration minister said Wednesday.
Human Rights Watch warns Nigeria to secure Taylor.
Spokesman: Liberia's Taylor not involved in attack against Guinea president.
Nigeria will end asylum for deposed Liberian ruler.
Nigeria said Saturday that it would end the asylum of deposed Liberian dictator Charles G. Taylor and hand him over to the Liberian government for trial.

Random Forests

Exiled former Liberian President Charles Taylor was not behind a failed January assassination attempt against Guinea President Lansana Conte, his spokesman said Tuesday, dismissing allegations by prosecutors for Sierra Leone's U.N.-backed war crimes tribunal.
"Taylor is not in a position to do that."
Liberia's former president Charles Taylor, accused of war crimes for his role in Sierra Leone's civil war, is a "menace to west Africa," Guinea's territorial administration minister said Wednesday.
Nigeria said Saturday that it would end the asylum of deposed Liberian dictator Charles G. Taylor and hand him over to the Liberian government for trial.

SVM

Exiled former Liberian President Charles Taylor was not behind a failed January assassination attempt against Guinea President Lansana Conte, his spokesman said Tuesday, dismissing allegations by prosecutors for Sierra Leone's U.N.-backed war crimes tribunal.
Liberia's former president Charles Taylor, accused of war crimes for his role in Sierra Leone's civil war, is a "menace to west Africa," Guinea's territorial administration minister said Wednesday.
Now, the United States is among those who say Taylor should be extradited.
But Liberia's President Ellen Sirleaf Johnson said: "Taylor should rather go to Sierra Leone rather than coming to Liberia".

Ensemble Heterogéneo

Exiled former Liberian President Charles Taylor was not behind a failed January assassination attempt against Guinea President Lansana Conte, his spokesman said Tuesday, dismissing allegations by prosecutors for Sierra Leone's U.N.-backed war crimes tribunal.

"Taylor is not in a position to do that."
Liberia's former president Charles Taylor, accused of war crimes for his role in Sierra Leone's civil war, is a "menace to west Africa," Guinea's territorial administration minister said Wednesday.
Nigeria said Saturday that it would end the asylum of deposed Liberian dictator Charles G. Taylor and hand him over to the Liberian government for trial.

Ensemble Homogéneo

Exiled former Liberian President Charles Taylor was not behind a failed January assassination attempt against Guinea President Lansana Conte, his spokesman said Tuesday, dismissing allegations by prosecutors for Sierra Leone's U.N.-backed war crimes tribunal.
Liberia's former president Charles Taylor, accused of war crimes for his role in Sierra Leone's civil war, is a "menace to west Africa," Guinea's territorial administration minister said Wednesday.
Earlier Nigeria's President Olusegun Obasanjo had said in a statement that Nigeria would honour a request from Liberia that the former president and war crimes suspect be handed over, but did not say when this would happen.

T-LexRank

Exiled former Liberian President Charles Taylor was not behind a failed January assassination attempt against Guinea President Lansana Conte, his spokesman said Tuesday, dismissing allegations by prosecutors for Sierra Leone's U.N.-backed war crimes tribunal.
Liberia's former president Charles Taylor, accused of war crimes for his role in Sierra Leone's civil war, is a "menace to west Africa," Guinea's territorial administration minister said Wednesday.
Earlier Nigeria's President Olusegun Obasanjo had said in a statement that Nigeria would honour a request from Liberia that the former president and war crimes suspect be handed over, but did not say when this would happen.

T-LexReRank

Exiled former Liberian President Charles Taylor was not behind a failed January assassination attempt against Guinea President Lansana Conte, his spokesman said Tuesday, dismissing allegations by prosecutors for Sierra Leone's U.N.-backed war crimes tribunal.
Liberia's former president Charles Taylor, accused of war crimes for his role in Sierra Leone's civil war, is a "menace to west Africa," Guinea's territorial administration minister said Wednesday.
Earlier Nigeria's President Olusegun Obasanjo had said in a statement that Nigeria would honour a request from Liberia that the former president and war crimes suspect be handed over, but did not say when this would happen.

DensityBased

Exiled former Liberian President Charles Taylor was not behind a failed January assassination attempt against Guinea President Lansana Conte, his spokesman said Tuesday, dismissing allegations by prosecutors for Sierra Leone's U.N.-backed war crimes tribunal.
Liberia's former president Charles Taylor, accused of war crimes for his role in Sierra Leone's civil war, is a "menace to west Africa," Guinea's territorial administration minister said Wednesday.
"Taylor can now face trial at the war crimes court in Sierra Leone where he is indicted on 17 counts of war crimes and crimes against humanity for his role in the armed conflict there," she added.

PNR^2

I am speaking of former Liberian President Charles Taylor, who has not only escaped answering for his crimes so far but who may be given an opportunity to repeat them if the United States does not act.
The government has denied the accusations.
Al-Qaeda working with ex Liberia leader Taylor: UN court. Al-Qaeda's network is actively seeking to destabilize West Africa, partly through its links with ex-Liberian president Charles Taylor, who has given sanctuary to its operatives, a UN-backed court said Tuesday.
Even in exile, Charles Taylor's presence felt in Liberia by Lauren Gelfand.

Update Summarization

Humanos

Charles Taylor fled Liberia in August 2003 accepting an invitation from Nigeria for exile.
He lived in a luxury villa near the town of Calibar near the border with Cameroon.
He was accused of continuing to meddle in Liberian politics, of funding and planning the assassination of the President of Guinea and of backing Sierra Leone rebels.
He was also said to give support to Al-Qaeda in its efforts to destabilize West Africa.
In March 2006, Nigeria offered to return Taylor to Liberia for trial, but Liberia didn't want him so he remained in his luxury villa.

Redes Neuronais

The former president of Liberia, Charles G. Taylor, vanished Monday night, two days after the Nigerian government said it would end his asylum and allow him to face indictment by an international court here.
Taylor leaves Liberia for Sierra Leone. Former Liberian president and war crimes suspect Charles Taylor, captured in Nigeria, left Liberia for Sierra Leone aboard a United Nations helicopter on Wednesday afternoon, AFP correspondents at Monrovia airport said.
Liberia's warlord turned president Charles Taylor is due make his first court appearance in Sierra Leone Monday, the UN-backed Special Court announced Friday.

Random Forests

The former president of Liberia, Charles G. Taylor, vanished Monday night, two days after the Nigerian government said it would end his asylum and allow him to face indictment by an international court here.
"Ellen has created a problem for Liberia. "
Taylor leaves Liberia for Sierra Leone. Former Liberian president and war crimes suspect Charles Taylor, captured in Nigeria, left Liberia for Sierra Leone aboard a United Nations helicopter on Wednesday afternoon, AFP correspondents at Monrovia airport said.
Liberia's Charles Taylor faces judgment after years of mayhem.
Taylor to make first court appearance Monday.

SVM

The former president of Liberia, Charles G. Taylor, vanished Monday night, two days after the Nigerian government said it would end his asylum and allow him to face indictment by an international court here.
UN forces ready to arrest Taylor in Liberia. The United Nations peacekeeping force in Liberia (UNMIL) is ready to arrest and transfer to Freetown war crimes suspect Charles Taylor, who left Nigeria Wednesday, a UNMIL spokesman said.
Liberia's warlord turned president Charles Taylor is due make his first court appearance in Sierra Leone Monday, the UN-backed Special Court announced Friday.

Ensemble Heterogéneo

The former president of Liberia, Charles G. Taylor, vanished Monday night, two days after the Nigerian government said it would end his asylum and allow him to face indictment by an international court here.
UN forces ready to arrest Taylor in Liberia. The United Nations peacekeeping force in Liberia (UNMIL) is ready to arrest and transfer to Freetown war crimes suspect Charles Taylor, who left Nigeria Wednesday, a UNMIL spokesman said.
Liberia's warlord turned president Charles Taylor is due make his first court appearance in Sierra Leone Monday, the UN-backed Special Court announced Friday.

Ensemble Homogéneo

A close ally of former Liberian leader Charles Taylor Tuesday charged Tuesday that President Ellen Johnson Sirleaf knows the whereabouts of the war crimes suspect who went missing from his exile home in Nigeria.

Taylor leaves Liberia for Sierra Leone. Former Liberian president and war crimes suspect Charles Taylor, captured in Nigeria, left Liberia for Sierra Leone aboard a United Nations helicopter on Wednesday afternoon, AFP correspondents at Monrovia airport said.
Liberia's warlord turned president Charles Taylor is due make his first court appearance in Sierra Leone Monday, the UN-backed Special Court announced Friday.

T-LexRank

The former president of Liberia, Charles G. Taylor, vanished Monday night, two days after the Nigerian government said it would end his asylum and allow him to face indictment by an international court here.
Taylor leaves Liberia for Sierra Leone. Former Liberian president and war crimes suspect Charles Taylor, captured in Nigeria, left Liberia for Sierra Leone aboard a United Nations helicopter on Wednesday afternoon, AFP correspondents at Monrovia airport said.
Liberia's warlord turned president Charles Taylor is due make his first court appearance in Sierra Leone Monday, the UN-backed Special Court announced Friday.

T-LexReRank

The former president of Liberia, Charles G. Taylor, vanished Monday night, two days after the Nigerian government said it would end his asylum and allow him to face indictment by an international court here.
Taylor leaves Liberia for Sierra Leone. Former Liberian president and war crimes suspect Charles Taylor, captured in Nigeria, left Liberia for Sierra Leone aboard a United Nations helicopter on Wednesday afternoon, AFP correspondents at Monrovia airport said.
Liberia's warlord turned president Charles Taylor is due make his first court appearance in Sierra Leone Monday, the UN-backed Special Court announced Friday.

DensityBased

A close ally of former Liberian leader Charles Taylor Tuesday charged Tuesday that President Ellen Johnson Sirleaf knows the whereabouts of the war crimes suspect who went missing from his exile home in Nigeria.
Taylor leaves Liberia for Sierra Leone. Former Liberian president and war crimes suspect Charles Taylor, captured in Nigeria, left Liberia for Sierra Leone aboard a United Nations helicopter on Wednesday afternoon, AFP correspondents at Monrovia airport said.
Liberia's warlord turned president Charles Taylor is due make his first court appearance in Sierra Leone Monday, the UN-backed Special Court announced Friday.

PNR^2

UN forces ready to arrest Taylor in Liberia. The United Nations peacekeeping force in Liberia (UNMIL) is ready to arrest and transfer to Freetown war crimes suspect Charles Taylor, who left Nigeria Wednesday, a UNMIL spokesman said.
Taylor leaves Liberia for Sierra Leone. Former Liberian president and war crimes suspect Charles Taylor, captured in Nigeria, left Liberia for Sierra Leone aboard a United Nations helicopter on Wednesday afternoon, AFP correspondents at Monrovia airport said.
Ex-Liberian President Charles Taylor behind bars at Sierra Leone war-crimes tribunal.

Anexo C – Implementação de Algumas Funcionalidades em R

De seguida apresentamos a implementação em R de algumas funcionalidades. Incluir aqui o código completo não seria possível, dado o elevado número de linhas de código.

```
processaTopico <- function(my.folder){

	my.path <- paste0(main.folder,my.folder)

	texto <- readLines(paste0(my.path,'/topic.txt'))

	f <- scan(paste0(my.path,'/topic.txt'), character(0))

	f <- unique(gsub("[[:punct:]]", "", tolower(f)))

	sinonimos <- c()
	for(word in f){
		w <- gsub("[[:punct:]]", "", tolower(word)) #Remover a pontuação
		if (!w %in% stopwords()){ # Se nao for stopword
			filter <- getTermFilter("ExactMatchFilter", w, TRUE)
			terms <- getIndexTerms(c("NOUN","ADJECTIVE"), 1, filter)
			if (!is.null(terms)){
				sinonimos <- c(sinonimos, getSynonyms(terms[[1]]))
			}
		}
	}

	sinonimos <- unique(sinonimos[!sinonimos %in% f])
	frase.sinonimos <- paste(sinonimos, collapse=' ')

	writeLines(paste(texto, frase.sinonimos, sep=' '),file(paste0(my.path,'\\topic_wn.txt')))

	return(length(sinonimos))

}

preprocessing <- function (corpus, removeStopWords=T){

	corpus <- tm_map(corpus, removePunctuation)
	corpus <- tm_map(corpus, tolower)
	if (removeStopWords)
		corpus <- tm_map(corpus, removeWords, stopwords("english"))
	corpus <- tm_map(corpus, stemDocument,language="english") #Requer package snowballC
	corpus <- tm_map(corpus, stripWhitespace)
	#corpus <- tm_map(corpus, removeNumbers)

	corpus <- tm_map(corpus, PlainTextDocument)

}

calc.tf.idf <- function(tf.docs,tf.sentences){
	idf <- log10(nrow(tf.docs)/colSums(tf.docs>0))
	tf.idf <- tf.sentences %*% diag(idf)

	tf.idf
}

calc.tf.idf1 <- function(tf.sentences){
	idf <- log10((nrow(tf.sentences)+1)/(0.5+colSums(tf.sentences>0)))
	tf.idf <- tf.sentences %*% diag(idf)

	tf.idf
}

rel.sent.query <- function(tf.docs,tf.sentences,tf.topic){
	idf <- log10((nrow(tf.docs)+1)/(0.5+colSums(tf.docs>0)))
	rel <- rowSums((log10(tf.sentences + 1) %*% diag(log10(tf.topic + 1))) %*% diag(idf))

	rel
}

rel.sent.query1 <- function(tf.sentences, tf.topic){
```

```
	idf <- log10((nrow(tf.sentences)+1)/(0.5+colSums(tf.sentences>0)))
	rel <- rowSums((log10(tf.sentences + 1) %*% diag(log10(tf.topic + 1))) %*% diag(idf))

	rel
}

calc.cosine <- function(mat){

	cos.sim <- function(ix) {
		A <- mat[ix[1],]
		B <- mat[ix[2],]
		return( sum(A*B)/sqrt(sum(A^2)*sum(B^2)) )
	}

	n <- nrow(mat)
	cmb <- expand.grid(i=1:n, j=1:n)
	m <- matrix(apply(cmb,1,cos.sim),n,n)

	m[is.na(m)] <- 0
	m
}

create.summary <- function (s, df, s.sim, size, sim.threshold, len.historical, sim.historical.threshold=1, sent.min.words=6){

	df <- df[order(df$score, decreasing = TRUE), ]

	n.sentences <- nrow(df)
	sentences.selected <- c(rep(F,n.sentences - len.historical), rep(T,len.historical))
	ini.historical <- (n.sentences - len.historical)+1

	nw <- 0
	for (i in 1:n.sentences){
		if (!sentences.selected[df[i,]$pos]){

			if (nw + df[i,]$words <=size && max(s.sim[df[i,]$pos,sentences.selected],0) <= sim.threshold && df[i,]$words
>= sent.min.words){

				if (len.historical>0 && sim.historical.threshold<1){
					if (max(s.sim[df[i,]$pos,ini.historical : (ini.historical+len.historical-1)],0) <=
sim.historical.threshold){
						sentences.selected[df[i,]$pos] <- T
						nw <- nw + df[i,]$words
					}
					else
						#print("Frase muito semelhante ao histórico!")
				}
				else{
					sentences.selected[df[i,]$pos] <- T
					nw <- nw + df[i,]$words
				}
			}
		}
	}
	sentences.selected[(n.sentences-len.historical+1):(n.sentences)] <- F

	return(s[sentences.selected])
}

summarize.super <- function (cluster, texts, historical, scores, max.len, max.sim, max.sim.hist=1){

	txt <- c(texts,historical)
	txt.sentences <- as.vector(unlist(txt))
	n.txt <- length(txt.sentences)
	sentences <- as.vector(unlist(texts))
	n.sentences <- length(sentences)
	sentences.historical <- as.vector(unlist(historical))
	n.sent.historical <- length(sentences.historical)
	n.texts <- length(texts)
	n.historical <- length(historical)

	df <- data.frame(pos = 1:n.txt, sentence = txt, score = c(scores,rep(0,n.sent.historical)), words = unlist(lapply(strsplit(as.vector(txt),"
"),length)))
	sum <- create.summary(sentences, df, cluster$cosine, max.len, max.sim, n.sent.historical, max.sim.hist)

}

Density.BasedQ <- function (sim,tf.idf,bias,radius,d,t,cont=T)
{

	PowerMethod <- function (m, n, tf.idf, b, radius, d, epsilon=0.0001, maxIteration=1000){

		density <- 1/(1 + radius)
```

```
                m <- m * density
                diag(m)<-0
                m[is.nan(m)] <- 0
                m[which(rowSums(m)==0),]=1/n
                degree <- rowSums(m)
                m <- (m / degree)
                matTransp <- t(matrix((1-d) * (b/sum(b)), n,n, byrow=T)+ d*m)
                currVec <- matrix(1/n,n)

                for (i in 1:maxIteration){
                        prevVec <- currVec
                        currVec <- matTransp %*% currVec
                        error <- sum((currVec-prevVec)^2)
                        if (error < epsilon^2)
                                break
                }
                return(currVec)
        }
        n <- nrow(sim)
        cosineMatrix <- sim
        cosineMatrix[sim<=t] <- 0

        return(PowerMethod(cosineMatrix,n,tf.idf,bias,radius,d))
}

PNR2 <- function (sim,bias,d,n.sent,n.sent.hist,t=0.05,alfa=1,beta=-0.5,teta=0.5)
{

        my.GaussSeidel <- function (A, b, x0 = NULL, nmax = 1000, tol = .Machine$double.eps^(0.5))
        {
                my.norm <- function (x) sum(abs(x)^2)^(1/2)

                stopifnot(is.numeric(A), is.numeric(b))
                n <- nrow(A)

                if (is.null(x0)) {
                        x0 <- rep(1, n)
                }
                else {
                        x0 <- c(x0)
                }
                L <- tril(A)
                U <- eye(n)
                b <- as.matrix(b)
                x <- x0 <- as.matrix(x0)
                r <- b - A %*% x0
                r0 <- err <- my.norm(r)
                iter <- 0
                while (err > tol && iter < nmax) {
                        iter <- iter + 1
                        z <- qr.solve(L, r)
                        z <- qr.solve(U, z)
                        x <- x + z
                        r <- b - A %*% x
                        err <- my.norm(r)/r0
                }
                return(x)
        }

        n <- nrow(sim)
        M <- sim

        diag(M) <- 0
        degree <- colSums(M)
        M <- M / degree
        M[is.nan(M)] <- 0
        M[which(rowSums(M)==0),]=1/n

        M[1:n.sent,1:n.sent] <- M[1:n.sent,1:n.sent] * alfa

        if (n.sent.hist>0){

                M[(n.sent+1):(n.sent+n.sent.hist),(n.sent+1):(n.sent+n.sent.hist)] <-
M[(n.sent+1):(n.sent+n.sent.hist),(n.sent+1):(n.sent+n.sent.hist)] * alfa

                M[(n.sent+1):(n.sent+n.sent.hist),1:n.sent] <- M[(n.sent+1):(n.sent+n.sent.hist),1:n.sent] * beta
                M[1:n.sent,(n.sent+1):(n.sent+n.sent.hist)] <- M[1:n.sent,(n.sent+1):(n.sent+n.sent.hist)] * beta
        }

        b <- d * bias
        A <- diag(1,n) - teta * M
        R <- my.GaussSeidel(A, b, tol=0.000001)
```

```
	return(R)
}

create.data.frame <- function(df, txts, t, ident, lr.d=0.90, lr.t=0.05){
	n.sentences <- length(t$sentences) # número de frases
	n.sentences.models <- length(t$sentences.models) # número de frases dos modelos
	n.sentences.historical <- length(t$sentences.historical) # número de frases dos textos já conhecidos (histórico)
	num.linhas <- unlist(lapply(txts,length))

	# Identificador (este atributo não irá entrar na aprendizagem)
	ident <- rep(ident, n.sentences)

	###################################
	#### Atributos
	# Posicao da linha no texto
	pos.vec <- c()
	for(i in num.linhas){
		pos.vec <- c(pos.vec,1:i)
	}

	# Posicao da linha, conforme Valizadeh e Brazdil (2014): fPos(Si) = 1 - (i-1) / n
	pos1.vec <- c()
	for(i in num.linhas){
		pos1.vec <- c(pos1.vec,1-(0:(i-1))/i)
	}

	# Posicao da linha, conforme Bysani et al. (2009): fPos(Si) = 1 - i / 1000, se i<=3 ou n/1000, se i>3 (assumindo que o numero de linhas e inferior a 1000)
	pos2.vec <- c()
	for(i in num.linhas){
		pos2.vec <- c(pos2.vec,1-1:min(i,3)/1000)
		if (i> 3){
			pos2.vec <- c(pos2.vec,4:i/1000)
		}
	}

	# Numero de palavras
	sent.length <- unlist(lapply(strsplit(as.vector(t$sentences)," "),length))
	# Numero de palavras (sem stop words)
	sent.length.wsw <- rowSums(t$sentences.mat>0)

	# Sentence Frequency score (SFS), conforme Bysani et al. (2009)
	sfs <- rowSums(t$sentences.mat %*% diag(colSums(t$sentences.mat>0)/nrow(t$sentences.mat))) / rowSums(t$sentences.mat>0)

	# Document Frequency score (DFS), Schilder and Kondadandi (2008), citado em Bysani et al. (2009)
	dfs <- rowSums(t$sentences.mat %*% diag(colSums(t$doc.mat>0)/nrow(t$doc.mat))) / rowSums(t$sentences.mat>0)

	# TF-IDF, conforme Bysani et al. (2009)
	tf.idf <- apply(t$tf.idf,1,function(x) mean(x[x>0]))

	# Sentence Radius, conforme Valizadeh e Brazdil (2014)
	#centroid <- colMeans(t$tf.idf)
	#radius <- sqrt(rowSums(t(apply(t$tf.idf,1,function(x) (x-centroid))^2))/nrow(t$tf.idf))
	radius <- c()
	pos.ini <- 1
	for(i in 1:length(num.linhas)){
		t.tf.idf <- t$tf.idf[pos.ini:(pos.ini + num.linhas[i]-1),]
		if (num.linhas[i]>1){
			centroid <- colMeans(t.tf.idf)
			radius <- c(radius, sqrt(rowSums(t(apply(t.tf.idf,1,function(x) (x-centroid))^2))/nrow(t.tf.idf)))
		}
		else{
			radius <- c(radius, 0)
		}

		pos.ini <- pos.ini + num.linhas[i]
	}

	# Soma das semelhanças com as outras frases, conforme Valizadeh e Brazdil (2014)
	# threshold de 0.05 para evitar o problema de "link by chance"
	sim.to.others <- apply(t$cosine[1:n.sentences,1:n.sentences],1,function(x) sum(x[x>0.05])-1) # Temos de subtrair 1 para não contar com a semelhança da frase com ela própria

	# Soma das semelhanças com as frases top 5, conforme Valizadeh e Brazdil (2014)
	sim.to.top5 <- apply(t$cosine[1:n.sentences,1:n.sentences],1,function(x) sum(sort(x,dec=T)[2:6])) # A posicao 1 sera sempre 1 (semelhanca da frase com ela própria), por isso somamos do 2 ao 6

	# Relevancia e semelhanca com o topico
	topic.rel <- t$rel.topic[1:n.sentences]
	topic.sim <- t$cos.topic[1:n.sentences]

	# Valor do Biased-LexRak
	lex.rank <- Biased.LexRank (t$cosine, bias=t$rel.topic, lr.d, lr.t,T)[1:n.sentences]
```

```
	##################################


	##################################
	#### Atributos do update summarization
	if (n.sentences.historical>0){
		# Novelty Factor (NF), conforme Bysani et al. (2009)
		nf <- rowSums(t$sentences.mat %*% diag(colSums(t$doc.mat>0)/(colSums(t$doc.hist.mat>0)+nrow(t$doc.mat)))) /
rowSums(t$sentences.mat>0)

		# Novel and Topic Relevance Measure (NTRM), conforme Kogilavani e Balasubramanie (2012)
		ntrm <- nf + topic.sim

		# Soma das semelhanças com as frases do histórico
		# threshold de 0.05 para evitar o problema de "link by chance"
		sum.sim.historical <-
apply(t$cosine[1:n.sentences,(n.sentences+n.sentences.models+1):(n.sentences+n.sentences.models+n.sentences.historical)],1,function(x)
sum(x[x>0.05]))

		# Maior semelhança com as frases do histórico
		# threshold de 0.05 para evitar o problema de "link by chance"
		max.sim.historical <-
apply(t$cosine[1:n.sentences,(n.sentences+n.sentences.models+1):(n.sentences+n.sentences.models+n.sentences.historical)],1,function(x)
max(x[x>0.05],0))


		# Percentagem da semelhança com as outras frases que diz respeito a novidade
		perc.sim.historical <- sum.sim.historical / (sim.to.others + sum.sim.historical)
		perc.sim.historical[is.nan(perc.sim.historical)] <- 0 # Para fazer o replace dos valores NaN (caso em que ha divisão por zero)
	}
	##################################

	##################################
	#### Scores das frases
	if (n.sentences.models==0){
		# Nao tem modelos: teste
		score.cosine.sums <-  rep(0, n.sentences)
		score.cosine.max <- rep(0, n.sentences)
		score.rouge <- rep(0, n.sentences)
	}
	else{

		# Somatorio das semelhancas com todos as frases dos modelos
		# score.cosine.sums <- rowSums(t$cosine[1:n.sentences,(n.sentences+1):(n.sentences+n.sentences.models)])

		# Maior semelhanca com um das frases do modelo
		# score.cosine.max <- apply(t$cosine[1:n.sentences,(n.sentences+1):(n.sentences+n.sentences.models)],1,max)

		# ROUGE-1 ou ROUGE-2
		#score.rouge1 <- rowSums(((t$sentences.mat>0)+0) %*% diag(colSums(t$doc.model.mat>0))) /
rowSums(t$sentences.mat>0)
		score.rouge <- t$scores
	}
	score.rouge <- t$scores

	##################################

	if (n.sentences.historical>0){
		# Data frame Update Summarization
		df <- rbind(df, data.frame(ident=ident, frases=t$sentences, pos=pos.vec, pos1=pos1.vec, pos2=pos2.vec,
sent.length=sent.length, sent.length.wsw=sent.length.wsw, topic.rel=topic.rel, sim.to.others=sim.to.others, sim.to.top5=sim.to.top5,
topic.sim=topic.sim, sent.freq=sfs, doc.freq=dfs, sent.tf.idf=tf.idf, sent.radius=radius, lex.rank=lex.rank, novelty.factor=nf, novel.relevant=ntrm,
sum.sim.historical=sum.sim.historical, max.sim.historical=max.sim.historical, perc.sim.historical=perc.sim.historical, score= score.rouge))
	}
	else{
		# Data frame Summarization
		df <- rbind(df, data.frame(ident=ident, frases=t$sentences, pos=pos.vec, pos1=pos1.vec, pos2=pos2.vec,
sent.length=sent.length, sent.length.wsw=sent.length.wsw, topic.rel=topic.rel, sim.to.others=sim.to.others, sim.to.top5=sim.to.top5,
topic.sim=topic.sim, sent.freq=sfs, doc.freq=dfs, sent.tf.idf=tf.idf, sent.radius=radius, lex.rank=lex.rank, score=score.rouge))
	}

	return(df)

}

cria.sumarios.tac2008 <- function (main.path, folder.eval, list.df, df.scores.sum, df.scores.upd, clusters, fld.score, tipo, max.sim.historico=1){
	for (cluster in clusters) {
		sumA <- summarize.super(list.df$corpus[[paste0(cluster,'-A')]],list.df$corpus[[paste0(cluster,'-A')]]$sentences, historical=c(),
scores=df.scores.sum[grepl(paste0('^',cluster,'.', collapse="|"),df.scores.sum$ident),fld.score], max.len=100, max.sim=0.7)
		sumB <- summarize.super(list.df$corpus[[paste0(cluster,'-B')]],list.df$corpus[[paste0(cluster,'-B')]]$sentences,
historical=c(list.df$corpus[[paste0(cluster,'-A')]]$sentences), scores=df.scores.upd[grepl(paste0('^',cluster,'-B',
collapse="|"),df.scores.upd$ident),fld.score], max.len=100, max.sim=0.7, max.sim.hist=max.sim.historico)
```

```
                save.files(main.path, folder.eval, list(sumA, sumB), c(paste0(cluster, '.', tipo, '-', c('A','B'))))
        }
}

mod.svm <- function(df.tr, df.te){
        set.seed(123456789)
        sv <- svm(score ~ ., data=df.tr)
        predict(sv,df.te)
}

mod.nn <- function(df.tr, df.te){
        set.seed(123456789)
        net<- newff(n.neurons=c(ncol(df.tr)-1,10,10,1), learning.rate.global=1e-2, momentum.global=0.5,
    error.criterium="LMS", Stao=NA, hidden.layer="sigmoid",
    output.layer="purelin", method="ADAPTgdwm")
        nn <- train(net, rescale(df.tr[,-ncol(df.tr)]), rescale(df.tr[,ncol(df.tr)]), error.criterium="LMS", report=TRUE, show.step=1000, n.shows=5)

        sim(nn$net, rescale(df.te[,-ncol(df.te)]))
}

feature.elimination <- function (list.df.tr, list.df.te, campos.ini, f){

        minimo <- Inf
        minimo.campos <- campos.ini
        continua <- T
        ret <- list()

        while (continua){

                continua <- F
                campos <- minimo.campos
                campos.te <- campos

                i <- 0
                while(i<=length(campos)){

                        df.tr <- subset(list.df.tr,select=c(campos.te,'score'))
                        df.te <- subset(list.df.te,select=c(campos.te,'score'))

                        v.sum <- f(df.tr, df.te)
                        sc <- MSE(v.sum,df.te$score)
                        ret[[length(ret)+1]] <- list(fields=campos.te, score=sc)

                        if (sc < minimo){
                                minimo <- sc
                                minimo.campos <- campos.te
                                if (i>1 && length(campos.te)>1)
                                        continua <- T
                                else
                                        continua <- F
                        }

                        i <- i+1
                        campos.te <- campos[-i]
                }
        }

        return (list(iteracoes=ret,melhor=list(minimo, campos=minimo.campos)))
}

fe.sum <- feature.elimination(list.df.2008$summary, list.df.2009$summary,
c('pos','pos1','pos2','sent.length','sent.length.wsw','topic.rel','sim.to.others','sim.to.top5','topic.sim','sent.freq','doc.freq','sent.tf.idf','sent.radius','lex.rank'),
mod.nn)
fe.upd <- feature.elimination(list.df.2008$update, list.df.2009$update,
c('pos','pos1','pos2','sent.length','sent.length.wsw','topic.rel','sim.to.others','sim.to.top5','topic.sim','sent.freq','doc.freq','sent.tf.idf','sent.radius','lex.rank','
novelty.factor','novel.relevant','sum.sim.historical','max.sim.historical','perc.sim.historical'), mod.nn)

list.df <- list.df.2008

csv.sum.2008 <- list.df$summary
df.sum.2008 <- subset(csv.sum.2008,select=-c(ident,frases,sim.to.top5,pos1))
csv.upd.2008 <- list.df$update
df.upd.2008 <- subset(csv.upd.2008,select=-c(ident,frases,sim.to.top5,pos1,novelty.factor,novel.relevant,perc.sim.historical))

csv.sum.2008.teste <- csv.sum.2008
csv.sum.2008.teste$scores.rf <- 0
csv.sum.2008.teste$scores.nn <- 0
csv.sum.2008.teste$scores.sv <- 0
csv.upd.2008.teste <- csv.upd.2008
csv.upd.2008.teste$scores.rf <- 0
csv.upd.2008.teste$scores.nn <- 0
csv.upd.2008.teste$scores.sv <- 0
```

```
k <- 12 # Folds
num.cluster.fold <- length(folder.textos.2008) / k

for (i in 1:k){

		##### Sumarization
		data<-df.sum.2008

		cl.teste <- ((i-1)*num.cluster.fold+1):(i*num.cluster.fold)
		frases.treino <- which(grepl(paste0(",clusters.2008[-cl.teste],'-A', collapse="|"),csv.sum.2008$ident))
		frases.teste <- which(grepl(paste0(",clusters.2008[cl.teste],'-A', collapse="|"),csv.sum.2008$ident))

		trainingset <- data[frases.treino,]
		testset <- data[frases.teste,]

		# # Redes Neuronais
		 set.seed(123456789)
		net <- newff(n.neurons=c(ncol(data)-1,10,10,1), learning.rate.global=1e-2, momentum.global=0.5,
	error.criterium="LMS", Stao=NA, hidden.layer="sigmoid",
	output.layer="purelin", method="ADAPTgdwm")
		 nn <- train(net, rescale(trainingset[,-ncol(data)]), rescale(trainingset[,ncol(data)]), error.criterium="LMS", report=TRUE,
show.step=1000, n.shows=5)
		 csv.sum.2008.teste[frases.teste,]$scores.nn <- sim(nn$net, rescale(testset[,-ncol(data)]))

		# Random Forest
		set.seed(123456789)
		rf.sum <- randomForest(score ~ ., data=trainingset, ntrees=250, mtry=3)
		csv.sum.2008.teste[frases.teste,]$scores.rf <- predict(rf.sum,testset)

		# SVM for Regression
		set.seed(123456789)
		sv.sum <- svm(score ~ ., data=trainingset, type='nu-regression', nu=0.2)
		csv.sum.2008.teste[frases.teste,]$scores.sv <- predict(sv.sum,testset)

		##### Update Sumarization
		data<-df.upd.2008

		cl.teste <- ((i-1)*num.cluster.fold+1):(i*num.cluster.fold)
		frases.treino <- which(grepl(paste0(",clusters.2008[-cl.teste],'-B', collapse="|"),csv.upd.2008$ident))
		frases.teste <- which(grepl(paste0(",clusters.2008[cl.teste],'-B', collapse="|"),csv.upd.2008$ident))

		trainingset <- data[frases.treino,]
		testset <- data[frases.teste,]

		# # Redes Neuronais
		 set.seed(123456789)
		net <- newff(n.neurons=c(ncol(data)-1,10,10,1), learning.rate.global=1e-2, momentum.global=0.5,
	error.criterium="LMS", Stao=NA, hidden.layer="sigmoid",
	output.layer="purelin", method="ADAPTgdwm")

		nn <- train(net, rescale(trainingset[,-ncol(data)]), rescale(trainingset[,ncol(data)]), error.criterium="LMS", report=TRUE, show.step=1000,
n.shows=5)
		csv.upd.2008.teste[frases.teste,]$scores.nn <- sim(nn$net, rescale(testset[,-ncol(data)]))

		# Random Forest
		set.seed(123456789)
		rf.upd <- randomForest(score ~ ., data=trainingset, ntrees=250, mtry=3)
		csv.upd.2008.teste[frases.teste,]$scores.rf <- predict(rf.upd,testset)

		# SVM for Regression
		set.seed(123456789)
		sv.upd <- svm(score ~ ., data=trainingset, type='nu-regression', nu=0.2)
		csv.upd.2008.teste[frases.teste,]$scores.sv <- predict(sv.upd,testset)

}

# Ensemble: RF + NN + SVM
csv.sum.2008.teste$scores.en <- (as.vector(csv.sum.2008.teste$scores.rf) + as.vector(csv.sum.2008.teste$scores.nn) +
as.vector(csv.sum.2008.teste$scores.sv)) / 3
csv.upd.2008.teste$scores.en <- (as.vector(csv.upd.2008.teste$scores.rf) + as.vector(csv.upd.2008.teste$scores.nn) +
as.vector(csv.upd.2008.teste$scores.sv)) / 3

# Ensemble Ponderado: RF + NN + SVM
csv.sum.2008.teste$scores.ep <- as.vector(csv.sum.2008.teste$scores.rf *.1) + as.vector(csv.sum.2008.teste$scores.nn *.55) +
as.vector(csv.sum.2008.teste$scores.sv *.35)
csv.upd.2008.teste$scores.ep <- as.vector(csv.upd.2008.teste$scores.rf *.1) + as.vector(csv.upd.2008.teste$scores.nn *.55) +
as.vector(csv.upd.2008.teste$scores.sv *.35)

SSE(csv.sum.2008.teste$scores.nn,csv.sum.2008.teste$score)
SSE(csv.upd.2008.teste$scores.nn,csv.upd.2008.teste$score)

# Sumários Neural Network
cria.sumarios.tac2008(main.path.2008, folder.eval.2008, list.df, csv.sum.2008.teste, csv.upd.2008.teste, clusters.2008, 'scores.nn', 'nn')
```

```
# Sumários Random Forest
cria.sumarios.tac2008(main.path.2008, folder.eval.2008, list.df, csv.sum.2008.teste, csv.upd.2008.teste, clusters.2008, 'scores.rf', 'rf')

# Sumários SVM for Regression
cria.sumarios.tac2008(main.path.2008, folder.eval.2008, list.df, csv.sum.2008.teste, csv.upd.2008.teste, clusters.2008, 'scores.sv', 'sv')

# Sumários Ensemble: RF + NN + SVM
cria.sumarios.tac2008(main.path.2008, folder.eval.2008, list.df, csv.sum.2008.teste, csv.upd.2008.teste, clusters.2008, 'scores.en', 'en')

# Sumários Ensemble: Ponderação de RF + NN + SVM
cria.sumarios.tac2008(main.path.2008, folder.eval.2008, list.df, csv.sum.2008.teste, csv.upd.2008.teste, clusters.2008, 'scores.ep', 'ep')

# Sumários LexRank
cria.sumarios.lr.tac2008(main.path.2008, folder.eval.2008, list.df, clusters.2008)

cria.rouge <- function(ficheiro, clusters, classificadores, com.peers=T, peers.a.incluir=0:71, tipo=c('A','B'), executa=T, nome.out='eval.out',
fich.prefixo='TEST'){
# Tipo A -> Sumarização
# Tipo B -> Update Summarization

                        f<-file(ficheiro)
                        linhas<-c('<ROUGE_EVAL version="1.5.5">')

                        for (cluster in clusters){
                                if (substr(cluster, 7, 7) %in% tipo){

                                        linhas<-c(linhas,paste0('<EVAL ID="',cluster,'">'))
                                        linhas<-c(linhas,'<PEER-ROOT>')
                                        linhas<-c(linhas,'C:\\_textos\\tac2008\\update\\avaliacao\\peers')
                                        linhas<-c(linhas,'</PEER-ROOT>')
                                        linhas<-c(linhas,'<MODEL-ROOT>')
                                        linhas<-c(linhas,'C:\\_textos\\tac2008\\update\\avaliacao\\models')
                                        linhas<-c(linhas,'</MODEL-ROOT>')
                                        linhas<-c(linhas,'<INPUT-FORMAT TYPE="SPL">')
                                        linhas<-c(linhas,'</INPUT-FORMAT>')
                                        linhas<-c(linhas,'<PEERS>')

                                        if (com.peers){
                                                for (i in peers.a.incluir){
                                                        linhas<-c(linhas,paste0('<P ID="',i,'">',cluster,'.',i,'</P>'))
                                                }
                                        }

                                        for (cl in classificadores){

                                                linhas<-c(linhas,paste0('<P ID="',cl,'">',substr(cluster, 1, 5),substr(cluster, 15, 15),'.',cl,'-
',substr(cluster, 7, 7),'</P>'))
                                        }

                                        linhas<-c(linhas,'</PEERS>')
                                        linhas<-c(linhas,'<MODELS>')
                                        for (m in df.models$m[df.models$c==substr(cluster, 1,7)]){
                                                linhas<-c(linhas,paste0('<M ID="',m,'">',cluster,'.',m,'</M>'))
                                        }

                                        linhas<-c(linhas,'</MODELS>')
                                        linhas<-c(linhas,'</EVAL>')


                                }

                        }
                        linhas<-c(linhas,'</ROUGE_EVAL>')

                        writeLines(linhas,f)
                        close(f)

                        if (executa){
                                executar.rouge.2008(nome.out, fich.prefixo)
                        }

}
```