VISUALIZAÇÃO COMPUTACIONAL
PROJETO FINAL

ALUNO: Sherlon Almeida da Silva **Nº USP:** 11361585 **EMAIL:** sherlon@usp.br

PROFESSORA: Rosane Minghim **DISCIPLINA:** SCC5836 **DATA:** 24/11/2019

RESUMO

Atualmente a quantidade de documentos textuais tem aumentado de maneira significativa. Com isso, a análise de coleções de documentos torna-se uma tarefa trabalhosa para o usuário. Visando auxiliar na análise destes documentos, existe a ferramenta Orange Canvas, uma ferramenta de código aberto, a qual permite explorar os dados de forma visual e interativa. No entanto, ainda existem poucas técnicas de projeção multidimensional dos dados na ferramenta. Portanto, o objetivo deste trabalho é estender as funcionalidades de visualização do Orange, implementando um módulo com a técnica de projeção multidimensional Least Square Projection (LSP), criada por Paulovich et al, em 2008.

INTRODUÇÃO AO PRÉ-PROCESSAMENTO E ANÁLISE DE DOCUMENTOS

Visto que a quantidade de documentos textuais tem aumentado significativamente, a análise de coleções de documentos torna-se uma tarefa trabalhosa e complicada, demandando soluções que auxiliem no pré-processamento, manipulação e análise destes dados.

Com o intuito de mitigar o trabalho do usuário e ao mesmo tempo facilitar a análise dos dados, ferramentas como o Orange Canvas são disponibilizadas para a comunidade. O sistema Orange é uma ferramenta de código aberto, a qual permite explorar os dados de forma visual e interativa, possibilitando utilizar diversas técnicas de aprendizado de máquina.

Uma ideia intuitiva acerca da análise de documentos textuais é agrupar aqueles documentos mais similares, gerando uma organização visual e de fácil exploração. Porém, antes disso, os documentos textuais da coleção de documentos precisam ser pré-processados, processados e representados por uma estrutura que mapeie suas características para que, então, os algoritmos de agrupamentos e projeção sejam utilizados, como visto no Figura 1.

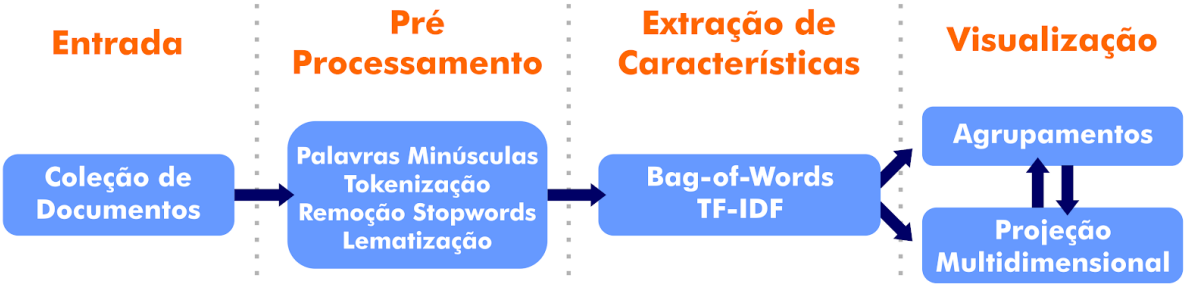


Figura 1. Análise de Coleções de Documentos Textuais.

Como mencionado acima, algumas técnicas de **Pré-Processamento** de texto devem ser utilizadas após a leitura da **Entrada** e o carregamento da coleção de documentos, como:

- Transformar palavras em minúsculo;
- **Tokenização:** Separar cada documento em termos individuais;
  - i.e Remover espaço em branco, quebras de linha, tabulações, e caracteres não alfa-numéricos;
- **Limpeza:** Filtrar os termos da coleção;
  - Remover as Stop Words, a qual é uma lista de termos não representativos para um documento, geralmente essa lista é composta por: preposições, artigos, advérbios, números, pronomes e pontuação;
  - Verificar a existência de sinônimos;
- **Lematização:** Redução da palavra ao seu radical;
  - i.e "Pens*" = Pensar, Pensando, Pensado, Pensativo, etc.

Uma vez que os documentos foram pré-processados, deverá ser realizada a **Extração de Características** que representem os documentos da coleção de documentos, os quais serão representados por um modelo que armazene as características extraídas.

Algumas técnicas utilizadas para extração de características em texto são:

- **Bag-of-Words (BoW):**
  - conta o número de ocorrências de cada palavra em um documento.
- **Term Frequency - Inverse Document Frequency (TF-IDF)**:
  - é uma medida de importância para cada palavra de um documento em relação à coleção de documentos.
  - este valor pode ser utilizado para a remoção das palavras não representativas da coleção de documentos, para melhorar os grupos.

Algumas técnicas utilizadas para representação dos documentos são:

- **Vector Space Model (VSM):**
  - é uma técnica utilizada para representação de documentos textuais, utilizando o Bag-of-Words ou o TF-IDF do vocabulário contido na coleção de documentos como sendo as suas características.
  - cada documento é descrito como um vetor multidimensional onde a dimensão é definida pelo número de palavras na coleção.
- **Word Embbedings:**
  - treina uma rede neural para obter as características intrínsecas à uma determinada língua ou à uma determinada coleção de documentos.

Por fim, a partir dos vetores de características extraídos da coleção de documentos, é possível identificar os padrões nos dados e observar a similaridade entre os documentos. No Orange, o usuário pode escolher entre projetar em 2D a relação entre os documentos apenas utilizando o espaço de características original dos dados (Alta dimensão), ou utilizando uma matriz de similaridade 2D calculada por alguma medida de similaridade, como por exemplo a distância do cosseno, a qual é comumente utilizada para representações textuais uma vez que atua sobre ângulos entre vetores multidimensionais, mitigando o ruído de dados esparsos.

A relação entre os dados em um espaço de características de alta dimensão pode ser representada em baixa dimensão, por exemplo, em um Scatterplot 2D, onde os eixos são coordenadas geradas por algoritmos de projeção multidimensional, como t-SNE, PCA, MDS, LSP, entre outros.

Para o desenvolvimento do Widget do LSP no Orange, foi utilizada a biblioteca mppy, desenvolvida por Thiago Henrique, a qual é uma biblioteca de projeção multidimensional que gera representações 2D de bases de dados de alta dimensão. Também, foi utilizada a base de dados CBR-ILP-IR-SON-INT para a validação do projeto, a qual é uma coleção de artigos composta por 682 documentos textuais, contendo como atributos o título, autores, resumo, e referências de artigos científicos em 5 diferentes áreas: Case-Based Reasoning (CBR),

Inductive Logic Programming (ILP), Information Retrieval (IR), Sonification (SON) e Intruder (INT). Sendo este último para validar a segregação das projeções entre INT e SON.

A seguir será apresentado o módulo desenvolvido para o Orange, os resultados obtidos acerca da análise da base de dados, e as conclusões. Ademais, nos anexos deste trabalho encontra-se um tutorial detalhado de como instalar e executar o módulo **Orange3-VICG-USP-Add-on** no Orange.

PROJETO DE VISUALIZAÇÃO DESENVOLVIDO

A descrição da metodologia adotada na condução deste trabalho e a apresentação do módulo Orange desenvolvido será baseada no workflow apresentado na Figura 2. Neste workflow é possível identificar o passo a passo desde a importação da coleção de documentos [1], a definição dos atributos [2], o pré-processamento dos dados [3], a criação do vetor de características e do modelo VSM [4], os agrupamentos com K-Means [5] e a visualização dos dados com técnicas de projeção multidimensional em [6] [7].

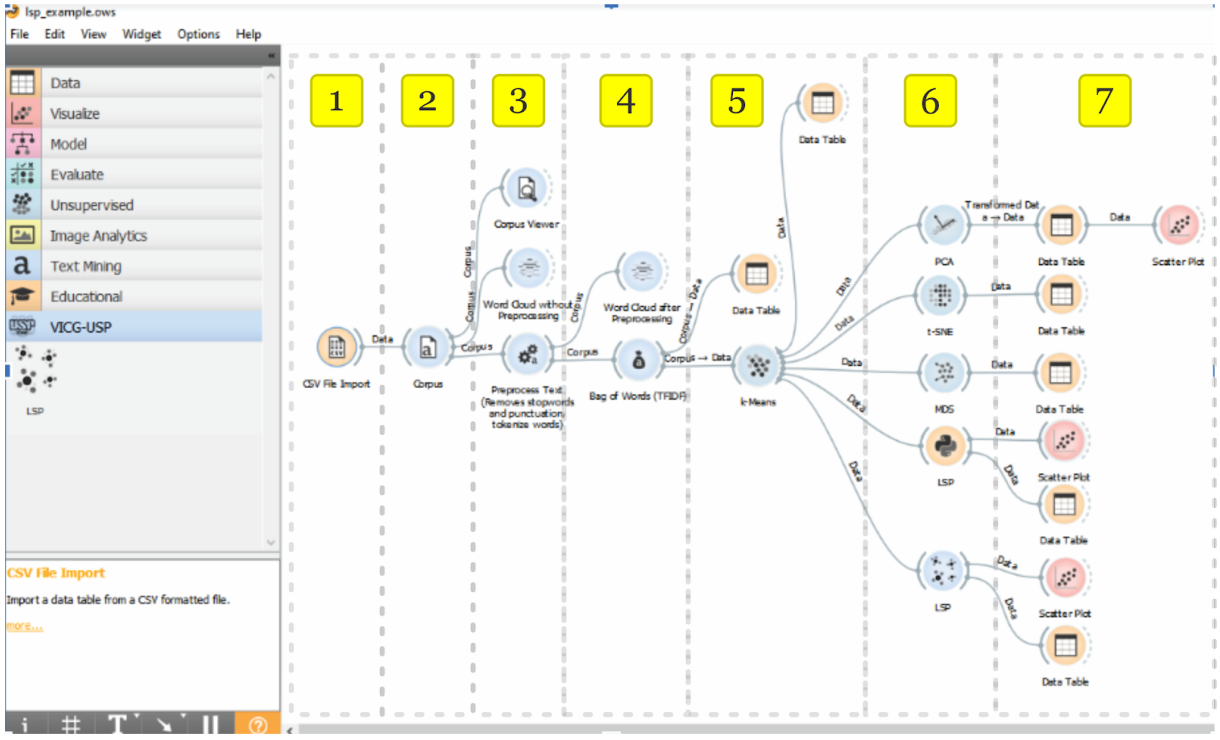
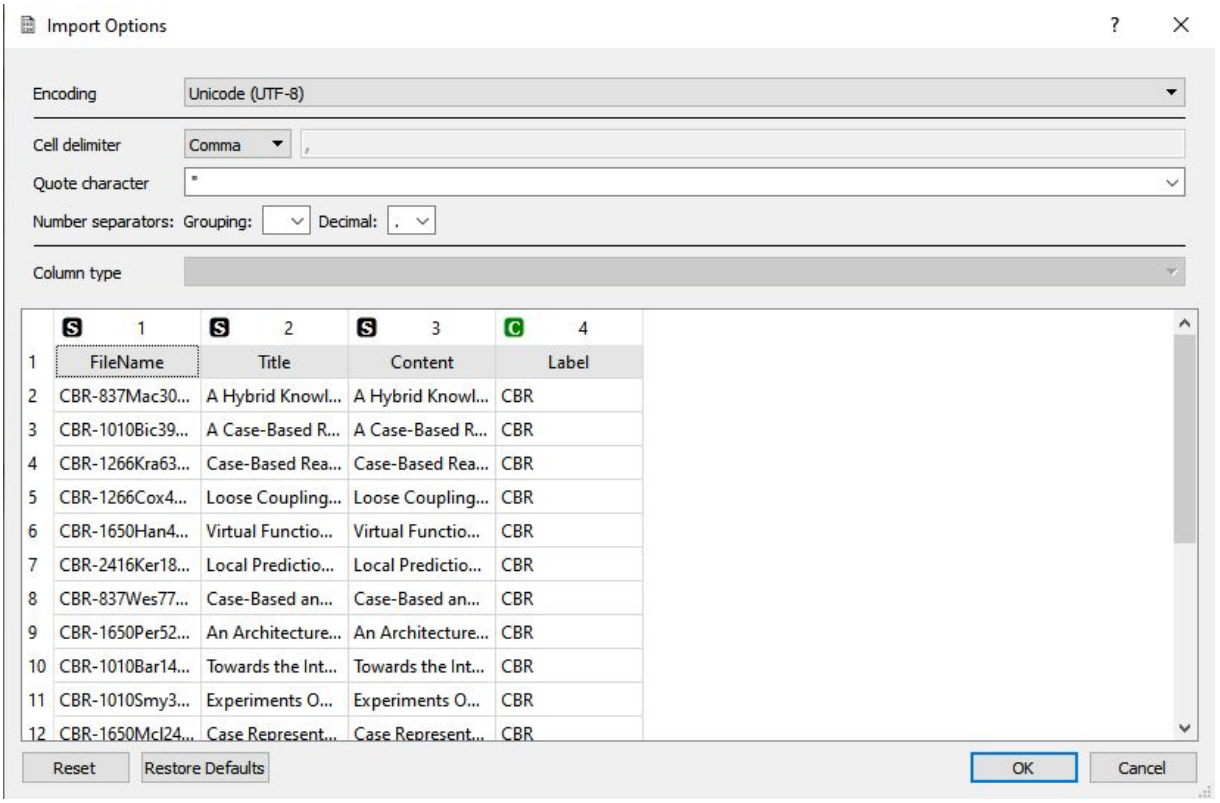


Figura 2. Workflow para Análise de Documentos Textuais.

A seguir será apresentado e discutido cada uma das etapas supracitadas. Primeiramente, a coleção de documentos é carregada com o widget **CSV File Import** [1], onde também é preciso definir os tipos de atributos encontrados durante a leitura dos dados. Como é possível ver na Figura 3, a coleção de documentos possui os atributos Filename (Textual), Title (Textual), Content (Textual), Label (Classes).



	1	2	3	4
1	FileName	Title	Content	Label
2	CBR-837Mac30...	A Hybrid Knowl...	A Hybrid Knowl...	CBR
3	CBR-1010Bic39...	A Case-Based R...	A Case-Based R...	CBR
4	CBR-1266Kra63...	Case-Based Rea...	Case-Based Rea...	CBR
5	CBR-1266Cox4...	Loose Coupling...	Loose Coupling...	CBR
6	CBR-1650Han4...	Virtual Functio...	Virtual Functio...	CBR
7	CBR-2416Ker18...	Local Predictio...	Local Predictio...	CBR
8	CBR-837Wes77...	Case-Based an...	Case-Based an...	CBR
9	CBR-1650Per52...	An Architecture...	An Architecture...	CBR
10	CBR-1010Bar14...	Towards the Int...	Towards the Int...	CBR
11	CBR-1010Smy3...	Experiments O...	Experiments O...	CBR
12	CBR-1650Mcl24...	Case Represent...	Case Represent...	CBR

Figura 3. Definição dos atributos da coleção de documentos.

Na sequência, os dados são passados para o widget **Corpus** [2], onde são definidos os atributos que serão utilizados como dados representativos da coleção de documentos (Características). Como é possível ver na Figura 4, foram identificados 682 documentos na base de dados carregada e foram selecionados os atributos *Content* e *Title* como atributos representativos para a extração de características, enquanto que o *FileName* foi ignorado, uma vez que o "Nome do Arquivo" não é capaz de descrever o conteúdo do documento.

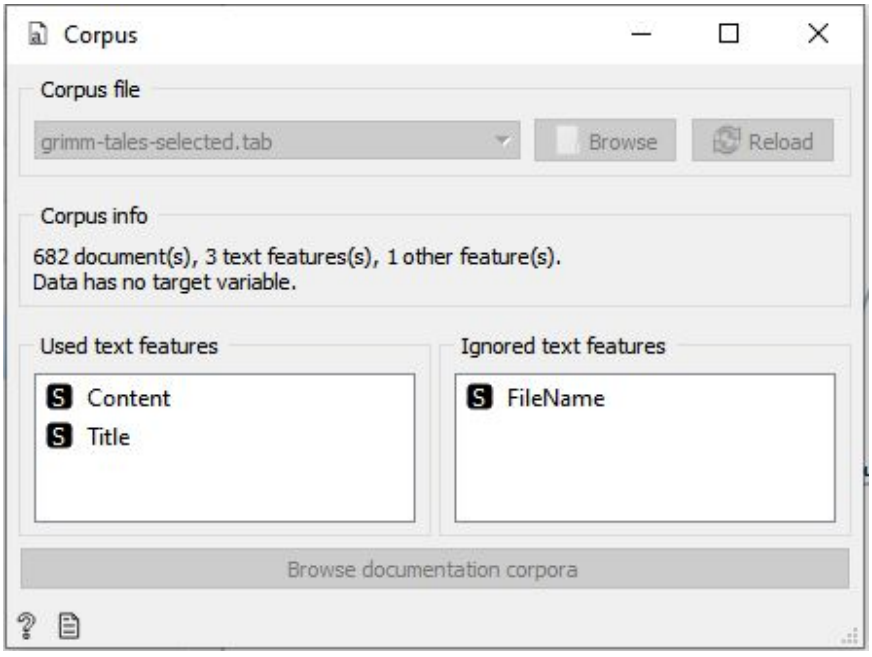


Figura 4. Definição dos atributos representativos (Características).

Na próxima etapa do workflow [3], o texto carregado da coleção de documentos é pré processado para padronização dos dados e remoção de pontuações e palavras não significativas. Primeiramente as palavras foram convertidas em minúsculas, as pontuações e caracteres não alfanuméricos foram removidos, bem como foram removidas as palavras que apareciam com menos de 10% de frequência e com mais de 50% de frequência (Figura 5).

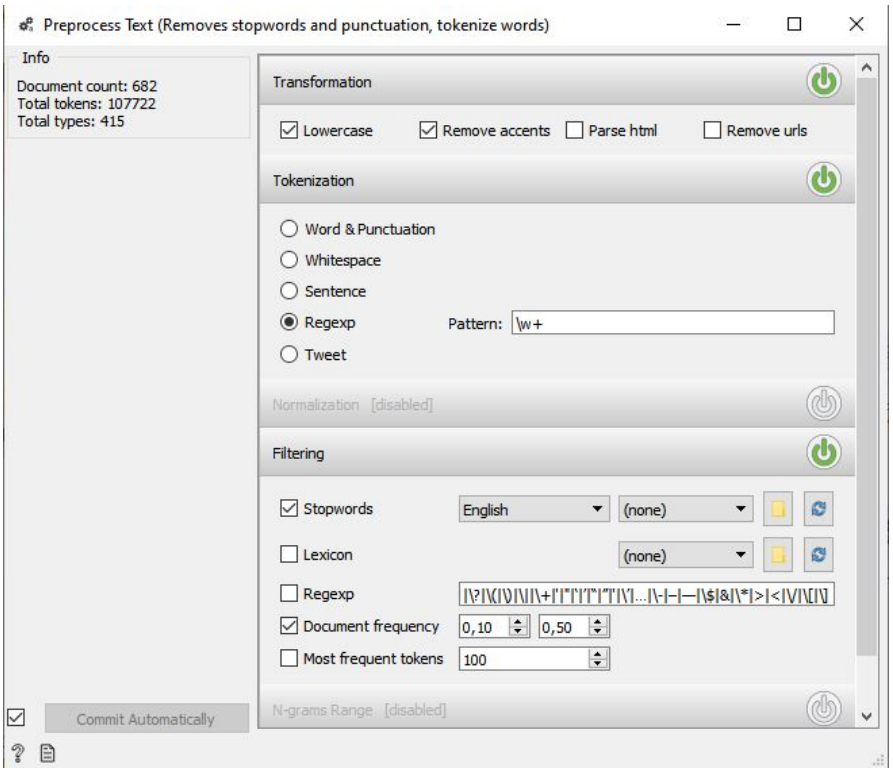
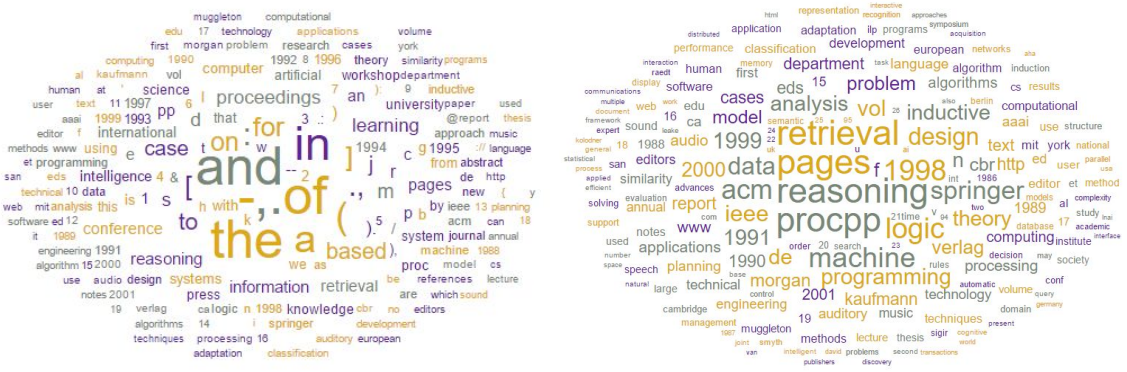


Figura 5. Pré-processamento da coleção de documentos.

Para visualizar as modificações realizadas com o pré-processamento dos dados foram criadas duas WordClouds, representadas na Figura 6, onde à esquerda (a) temos a frequência das palavras encontradas na coleção de documentos antes do pré-processamento, e em (b) após o pré-processamento. Como é possível observar nas figuras, antes do pré-processamento os dados não eram representativos uma vez que continham muita pontuação e palavras comuns (por exemplo: and, in, of, the), enquanto que com o pré-processamento a coleção de documentos contempla apenas palavras que são representativas para suas respectivas áreas do conhecimento.



a) Sem pré-processamento b) Com pré-processamento

Figura 6. WordClouds da coleção de documentos.

Uma vez que a coleção de documentos possui apenas as palavras representativas, é necessário criar uma representação numérica que reflita a quantidade de cada palavra na coleção, pois assim será possível identificar posteriormente aqueles artigos que possuem mais palavras em comum. Para isso, foi utilizada a técnica TF-IDF [4], a qual captura a importância de cada palavra em relação à coleção de documentos (Figura 7).

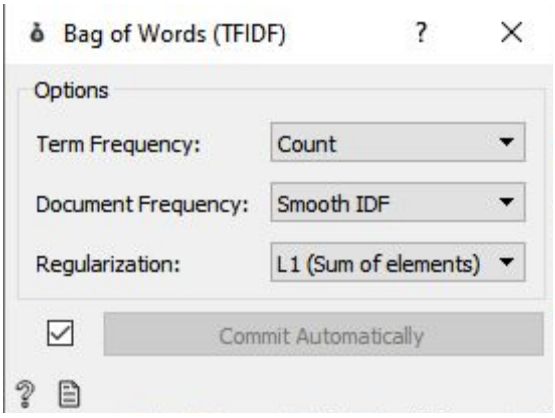


Figura 7. Cálculo de Frequência e Importância das palavras.

Com esses valores calculados, é criado um vetor de características para cada documento (VSM), o qual poderá ser usado para análise de padrões e identificação de documentos similares na etapa a seguir com o algoritmo de agrupamentos K-means [5].

Por fim, as etapas [6] e [7] apresentam a visualização dos dados utilizando as técnicas de projeção multidimensional PCA, t-SNE, MDS e LSP. Como é possível observar no workflow da Figura 2, foram desenvolvidas duas versões do LSP, sendo a primeira com um widget de **Script Python**, a qual é uma funcionalidade oferecida pela ferramenta Orange, bem como foi desenvolvido o próprio widget do LSP, o qual foi implementado no **Módulo VICG-USP** (apresentado à esquerda da Figura 2).

Como já mencionado anteriormente, foi utilizada a implementação do LSP proveniente da biblioteca mppy, onde a execução do LSP ocorre da seguinte forma. Dado uma matriz bidimensional, onde as linhas são os dados e as colunas os seus atributos numéricos, a função **mppy.lsp_2d(...)** retorna uma matriz bidimensional com duas colunas, sendo elas as coordenadas (x,y) dos dados para uma projeção no plano.

Partindo disso, foi implementado um script python adequado aos padrões de implementação da ferramenta Orange (documentação neste link), o qual recebe como entrada os dados provenientes das etapas anteriores do workflow e gera uma saída com a propagação dos dados antigos agregados com as coordenadas da projeção 2D do LSP. Assim, permitindo visualizar os dados com o widget Scatterplot.

No entanto, a utilização de um widget de **Script Python** possui limitações, uma vez que é necessário que o usuário possua conhecimento de programação para, caso desejar, alterar os argumentos utilizados no cálculo do LSP, como a vizinhança desejada. Portanto, visando contornar esta limitação e facilitar o acesso ao módulo desenvolvido e sua utilização, foi desenvolvido um módulo do Orange, o qual pode ser baixado e instalado facilmente. A seguir, na Figura 9, são apresentados os ícones criados para o módulo e para o widget do LSP.

Assim como no widget do **Script Python**, o widget do LSP recebe como entrada os dados provenientes do workflow, realiza a computação das coordenadas (x,y) com o LSP sobre o espaço de características, e por fim propaga os dados antigos agregados com as coordenadas calculadas, permitindo visualizar os resultados em um widget **Scatterplot**. Porém, neste widget é possível selecionar interativamente o número da vizinhança desejada dispensando o acesso ao código fonte do script.

a)Ícone do Módulo b)Ícone do Widget LSP

Figura 9. Ícones criados para o Add-on.

RESULTADOS

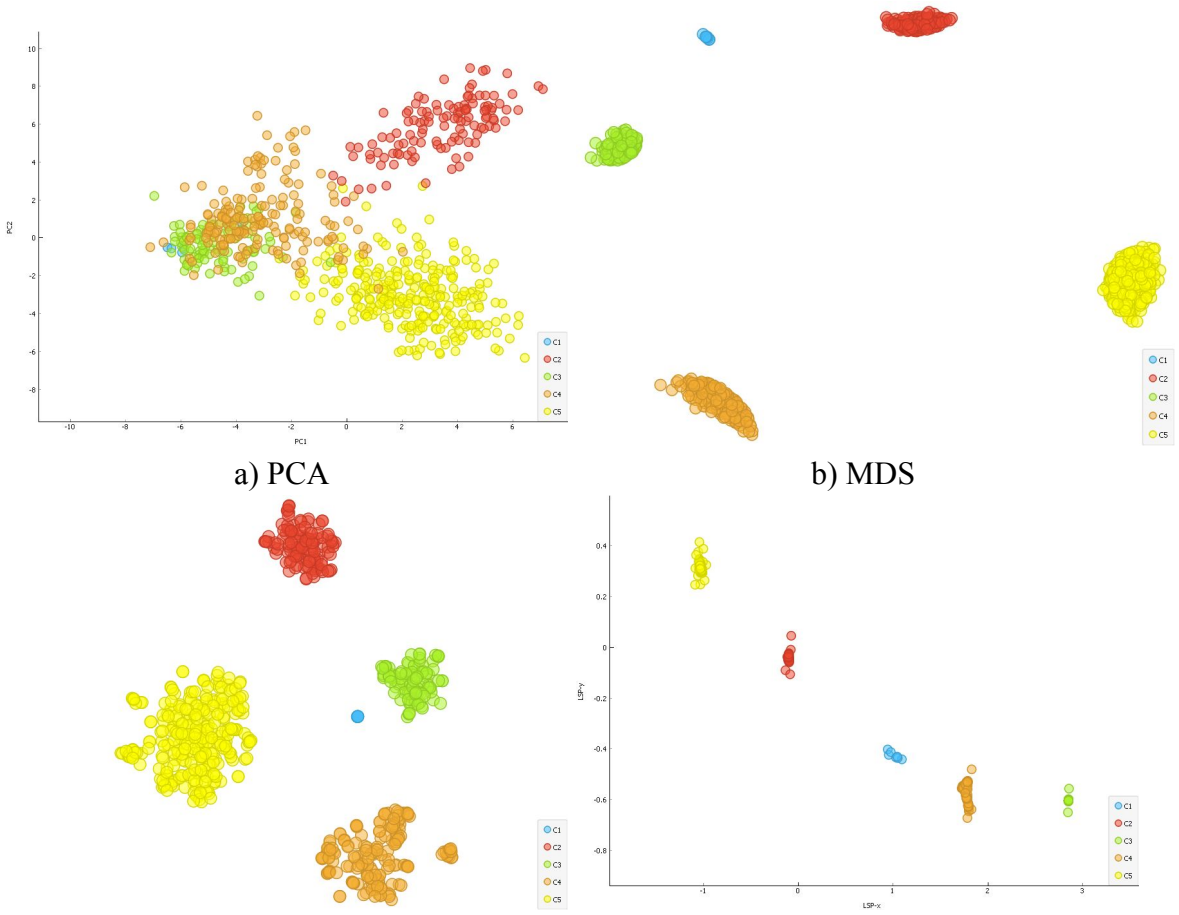
A seguir serão apresentados os resultados obtidos durante a execução do workflow proposto na Figura 2. Primeiramente, a Figura 10 apresenta a tabela de dados resultante da execução do algoritmo LSP, onde são apresentados os dados propagados anteriormente no workflow, como os dados de entrada (FileName, Title, Content), os dados do algoritmo de agrupamento (Cluster, Silhouette) e os dados calculados pelo LSP (LSP-x, LSP-y).

bow-feature hidden skip-normalizati	FileName	Title	Content	Cluster	Silhouette	LSP-x	LSP-y	Label
1	CBR-837Mac30...	A Hybrid Knowl...	A Hybrid Knowl...	C5	0.728141	-1.43696	0.879493	CBR
2	CBR-1010Bic39...	A Case-Based R...	A Case-Based R...	C5	0.727558	-1.43314	0.876783	CBR
3	CBR-1266Kra63...	Case-Based Rea...	Case-Based Rea...	C5	0.726704	-1.43734	0.863475	CBR
4	CBR-1266Cox4...	Loose Coupling...	Loose Coupling...	C5	0.727819	-1.42983	0.86723	CBR
5	CBR-1650Han4...	Virtual Functio...	Virtual Functio...	C5	0.720542	-1.43754	0.863221	CBR
6	CBR-2416Ker18...	Local Predictio...	Local Predictio...	C5	0.72463	-1.44095	0.877287	CBR
7	CBR-837Wes77...	Case-Based an...	Case-Based an...	C5	0.729166	-1.43444	0.891629	CBR
8	CBR-1650Per52...	An Architecture...	An Architecture...	C5	0.726872	-1.44731	0.879047	CBR
9	CBR-1010Bar14...	Towards the Int...	Towards the Int...	C5	0.729747	-1.43566	0.867409	CBR
10	CBR-1010Smy3...	Experiments O...	Experiments O...	C5	0.728979	-1.44254	0.849211	CBR
11	CBR-1650Mcl24...	Case Represent...	Case Represent...	C5	0.729036	-1.44092	0.887791	CBR
12	CBR-2416Dia73...	Poetry Generati...	Poetry Generati...	C5	0.729075	-1.4454	0.874833	CBR
13	CBR-1010Alu12...	A Case-Based A...	A Case-Based A...	C5	0.727789	-1.43797	0.881536	CBR
14	CBR-1898Mcs1...	Intelligent Case...	Intelligent Case...	C5	0.727104	-1.46502	0.875027	CBR
15	CBR-1010Vis33...	Reuse of Knowl...	Reuse of Knowl...	C5	0.729835	-1.43255	0.877575	CBR
16	CBR-1266Bra51...	Stratified Case-...	Stratified Case-...	C5	0.726779	-1.43868	0.864025	CBR
17	CBR-1650Pra53...	Supporting Reu...	Supporting Reu...	C5	0.72817	-1.4379	0.866845	CBR
18	CBR-1898Mil41...	Maintenance of...	Maintenance of...	C5	0.730347	-1.44129	0.878535	CBR
19	CBR-1266Tau15...	Using Case-Bas...	Using Case-Bas...	C5	0.723717	-1.43946	0.881285	CBR
20	CBR-1266Lee37...	A Case Study of...	A Case Study of...	C5	0.730495	-1.43832	0.859707	CBR

Figura 10. Tabela de dados resultante da execução do LSP.

Acima os dados são apresentados em uma matriz de documentos (Linhas) e seus respectivos atributos (Colunas), porém na Figura 11 os dados são apresentados como pontos no espaço 2D, onde suas cores refletem os grupos calculados com o algoritmo de agrupamentos K-means e a posição dos pontos sendo calculada pelos algoritmos de projeção.

Como é possível observar na Figura 11, o algoritmo PCA não conseguiu segregar bem os 5 grupos da base de dados **CBR-ILP-IR-SON-INT**, apresentando grupos sobrepostos. No entanto as demais técnicas, MDS, t-SNE e LSP foram capazes de identificar e segregar a similaridade entre os documentos de cada grupo de maneira bem definida.



a) PCA b) MDS

c) t-SNE d) LSP

Figura 11. Projeção Multidimensional da coleção de documentos.

CONCLUSÃO

Visto que o número de documentos textuais tem crescido, ferramentas visuais e interativas como o Orange se tornam imprescindíveis para a análise de documentos textuais. Com isso, foi apresentado um workflow para a análise de documentos textuais utilizando a ferramenta Orange contemplando desde a etapa de carregamento e pré-processamento dos dados até a visualização a partir de agrupamentos e projeção multidimensional.

Além do mais, foi desenvolvido um **Script Python** e um **Módulo Orange** da técnica de projeção multidimensional Least Square Projection (LSP), desenvolvida por Paulovich et al, em 2008. A qual, embora apresente bons resultados para diferentes tipos de dados, foi desenvolvida especificamente para análise de dados esparsos, como documentos textuais.

Em suma, além de oferecer um workflow para a análise de documentos textuais de forma interativa e visual, foi desenvolvida uma nova funcionalidade para a ferramenta Orange, a qual está disponível para a comunidade neste link. Ademais, a seguir em “Anexos” consta um tutorial para a instalação da ferramenta Orange e do módulo desenvolvido.

ANEXO I - Tutorial de Instalação do Add-on Orange Implementado

1. Instalação Anaconda

- Acessar o site www.anaconda.com;
- Clicar em "Downloads", ou então neste link;
- Selecionar a versão desejada (i.e 64bits e python3.7, no meu caso);
- Após o download terminar, executar o instalador do Anaconda;
- Instalar o Orange pelo anaconda: só clicar em instalar;

2. Instalação Add-ons Orange

- Abrir o prompt do Anaconda;
- Digitar os seguintes comandos:
  - conda config --add channels conda-forge
  - conda install orange3-text
- Instalação do MPPY
  - pip install mppy

3. Baixar o Add-on do LSP em:

- Download neste link: :
  - https://github.com/SherlonAlmeida/Orange3-VICG-USP-Add-on.git
- Abrir o prompt do Anaconda
- Ir até a pasta do Widget:
  - Exemplo:
    - cd Documents
    - cd Orange3-VICG-USP-Add-on
  - Instalar o Widget com acesso e edição ao código:
    - pip install -e .
  - Ou apenas instalar o Widget somente para utilização:
    - pip install .
  - Abrir o Orange:
    - python -m Orange.canvas
- Pronto!

4. Divirta-se! :)

5. Referências:

- **[1], [2], [3], [4];**