UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

**Interactive keyterm-based document clustering and
visualization via neural language models**

Eric Macedo Cabral

Dissertação de Mestrado do Programa de Pós-Graduação em Ciências
de Computação e Matemática Computacional (PPG-CMC)

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Eric Macedo Cabral

Interactive keyterm-based document clustering and visualization via neural language models

Master dissertation submitted to the Institute of
Mathematics and Computer Sciences – ICMC-USP,
in partial fulfillment of the requirements for the
degree of the Master Program in Computer Science
and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Profa. Dra. Rosane Minghim

USP – São Carlos
August 2020

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

C117 Cabral, Eric Macedo
Interactive keyterm-based document clustering
and visualization via neural language models / Eric
Macedo Cabral; orientadora Rosane Minghim. -- São
Carlos, 2020.
97 p.

Dissertação (Mestrado - Programa de Pós-Graduação
em Ciências de Computação e Matemática
Computacional) -- Instituto de Ciências Matemáticas
e de Computação, Universidade de São Paulo, 2020.

1. Interactive document clustering. 2. Visual
analytics. 3. Neural language models. I. Minghim,
Rosane, orient. II. Título.

Eric Macedo Cabral

**Agrupamento interativo e visualização de documentos
baseado em termos-chave via modelos neurais de
linguagem**

Dissertação apresentada ao Instituto de Ciências
Matemáticas e de Computação – ICMC-USP,
como parte dos requisitos para obtenção do título
de Mestre em Ciências – Ciências de Computação e
Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e
Matemática Computacional

Orientadora: Profa. Dra. Rosane Minghim

USP – São Carlos
Agosto de 2020

ACKNOWLEDGEMENTS

First of all, I would like to thank my advisor Rosane Minghim for guiding, advising, and for having so much patience with me throughout such an amazing and important period in my professional career. Additionally, I thank Evangelos Milios from Dalhousie University (Canada), who consistently supported and helped me during this research. I've learned a lot from both of you.

I'm forever grateful for having such an amazing life partner. Rodolfo, thank you for always believing in me and for always making every day worth the fight. You are and will always be my only love.

I thank my mother Joana, who always believed in me, who provided so much love and caring above all. I owe you every and any achievement in this life.

I also thank the ICMC-USP and the *Conselho Nacional de Desenvolvimento Científico e Tecnológico – Brasil*, (CNPq) who funded this research (grant number 133685/2018-7).

*“ La vrai générosité envers l’avenir
consiste à tout donner au présent. ”*
(Albert Camus)

RESUMO

CABRAL, E. M. **Agrupamento interativo e visualização de documentos baseado em termos-chave via modelos neurais de linguagem**. 2020. 97 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2020.

Técnicas interativas de agrupamento de dados colocam o usuário no ciclo do algoritmo de agrupamento, permitindo não somente uma melhor qualidade de agrupamento, mas também apoiando a tarefa de descoberta de conhecimento em grandes volumes textuais. A abordagem guiada por termos-chave é sem dúvida intuitiva permitindo ao usuário a interação com palavras representativas ao invés de interagir com um grande volume de documentos ou com modelos de tópicos complexos. Mais do que tornar o algoritmo de agrupamento ajustável com pouco esforço do usuário, a abordagem de agrupamento visualmente interativo permite que o usuário foque na exploração do corpus como uma tarefa incremental. Após cada interação, o usuário pode obter novas informações sobre o corpus e expressar essas informações como *feedback* para o algoritmo de agrupamento.

O sistema *Vis-Kt* apresenta-se como um sistema de visualização analítica para agrupamento de documentos baseado em termos-chave, com técnicas que superam as técnicas considerada como estado da arte, como a *Latent Dirichlet Allocation* e a *Non-negative Matrix Factorization*. Com uma abordagem guiada pelo usuário, o sistema *Vis-Kt* permite ao usuário modelar seu discernimento sobre o corpus por meio de conjuntos de termos-chave que descrevem grupos de documentos. No entanto, o sistema *Vis-Kt* e seus algoritmos dependem do modelo *Bag-of-Words*, que possui várias limitações em relação à escalabilidade da extração de informação, à incrementalidade do processo e à representação semântica dos dados.

Com o objetivo de superar as limitações inerentes ao *Bag-of-Words*, propomos uma atualização da representação por termos-chave para uma abordagem de aprendizado de máquina baseado em modelos neurais de linguagem. Tais modelos podem extrair informações semânticas e relações das palavras que compõem o corpus.

A principal contribuição deste projeto é um novo algoritmo interativo de agrupamento de documentos guiado por termos-chave e baseado em modelos neurais de linguagem. Essa abordagem mostra uma melhoria significativa em comparação com os algoritmos considerados estado da arte. O algoritmo de agrupamento proposto permite que o sistema *Vis-Kt* funcione de forma incremental, sem a necessidade de repetir todo processo de aprendizado e agrupamento desde o início. Isso torna o sistema adequado para o uso em análises de fluxos de texto. Para contribuir com a tarefa de descoberta de conhecimento e apoiar seu aspecto incremental, foi desenvolvida uma visualização baseada no diagrama de Sankey que representa as mudanças nos agrupamentos após cada interação com o corpus.

Um conjunto de experimentos quantitativos em conjuntos de dados de texto disponíveis publicamente foi realizado para avaliar os resultados dos agrupamentos obtidos. Os resultados reportados neste trabalho mostram que, na maioria dos casos experimentados, o algoritmo proposto apresenta uma melhoria significativa nas medidas de qualidade de agrupamentos em comparação com os algoritmos previamente adotados no sistema. Em todos os casos, o algoritmo proposto apresentou um ganho em tempo de processamento, principalmente nos maiores conjuntos de dados.

Também relatamos dois cenários de uso para avaliar qualitativamente o componente visual proposto.

Palavras-chave: Agrupamento Interativo de Documentos, Visualização Analítica, Modelos Neurais de Linguagem.

ABSTRACT

CABRAL, E. M. **Interactive keyterm-based document clustering and visualization via neural language models**. 2020. 97 p. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2020.

Interactive data clustering techniques put the user in the clustering algorithm loop, allowing not only better clustering quality, but also supporting the knowledge discovery task in large textual corpora. The keyterm guided approach is arguably intuitive, allowing the user to interact with representative words instead of interacting with a large volume of full-length documents or complex topic models. More than making the clustering algorithm adjustable with little user-effort, the visual interactive clustering approach allows the user to focus on exploring the corpus as an incremental task. After each interaction, the user can obtain new information about the corpus, and expresses it as feedback to the clustering algorithm.

The visual analytics system Vis-Kt presents itself as an interactive keyterm-based document clustering system, embedded with techniques that overcome the state-of-the-art ones, such as Latent Dirichlet Allocation and the Non-negative Matrix Factorization. With a user-guided approach, Vis-Kt allows the user to draw her insights into the corpus by describing document clusters with a small set of significative terms. However, Vis-Kt and its underlying clustering algorithms depend on the Bag-of-Words model, which has several limitations concerning the information extraction's scalability, the process' incrementality, and the data's semantic representation.

In order to overcome the limitations inherent to the Bag-of-Words model, we propose an update for the keyterm-based representation model to a machine learning approach based on neural language models. Such a model can extract semantic information and relationships from the words that are included in the corpus.

This project's main contribution is a novel interactive document clustering algorithm guided by keyterms and based on neural language models. This approach shows a significant improvement compared to the baseline algorithms, considered state-of-the-art. The proposed clustering algorithm allows Vis-Kt to work incrementally, without the need to repeat the entire learning and clustering processes from the beginning. This makes the system suitable for analyzing text streams. In order to contribute to the task of knowledge discovery and to support its incremental aspect, a visual component based on the Sankey diagram was developed to depict the clustering membership changes throughout the clustering loop after each interaction with the corpus.

A set of quantitative experiments on publicly available text datasets was performed to evaluate the obtained clustering results. The results reported in this work show that, in most of the experimented cases, the proposed algorithm presents a significant improvement in clustering

quality measures in comparison with the baseline algorithms. In all cases, the proposed algorithm showed a gain in processing time, especially in the largest datasets.

We also report two usage scenarios to qualitatively evaluate the proposed visual component.

Keywords: Interactive Document clustering, Visual Analytics, Neural Language Models.

LIST OF FIGURES

Figure 1 – The framework for learning word vectors.	32
Figure 2 – PV-DM graphical illustration.	32
Figure 3 – PV-DBOW model graphical illustration.	33
Figure 4 – The outcome of the number of neighbors in the LSP technique.	35
Figure 5 – 675 PV-DM vectors projected with three different dimensionality reduction techniques.	36
Figure 6 – A graphical model of the LDA algorithm.	38
Figure 7 – An overview of the iVisClustering system exploring the Jigsaw dataset. . . .	39
Figure 8 – A graphical model of the NMF decomposition matrices.	40
Figure 9 – An overview of UTOPIAN exploring the Jigsaw dataset.	41
Figure 10 – An overview of the testbed system, displaying several modules.	42
Figure 11 – An overview of TopicLens exploring information visualization related papers. .	43
Figure 12 – An overview of Vis-Kt’s pipeline.	44
Figure 13 – TFIDF weighted document-term matrix.	45
Figure 14 – An overview of Vis-Kt.	46
Figure 15 – ForceSPIRE’s usage with high-resolution displays.	47
Figure 16 – Overview of the ThemeRiver technique.	50
Figure 17 – An overview of the TIARA system exploring approximately 8,000 emails. . .	51
Figure 18 – ParallelTopics system exploring information visualization related documents. .	52
Figure 19 – An illustration of the TextFlow visual framework.	53
Figure 20 – Vis-Kt’s overview with the new temporal component highlighted.	54
Figure 21 – The HierarchicalTopics system exploring news articles from CNN News. . .	55
Figure 22 – An overview of the OpinionFlow system exploring tweets about the PRISM scandal.	56
Figure 23 – The visual analytics system RoseRiver exploring tweets and news articles about the PRISM scandal.	57
Figure 24 – An overview of the CiteRivers system.	58
Figure 25 – An overview of the temporal visualization of the CiteSpace II system. . . .	60
Figure 26 – Pipeline for the new framework.	65
Figure 27 – Illustration of user interactions with term clusters.	68
Figure 28 – Cubic Bézier curves applied to display the Sankey diagram’s flows.	69
Figure 29 – The overview of the proposed visual component.	70

Figure 30 – The projection and average silhouette score for t-SNE technique executed 100 times in the datasets. 77

Figure 31 – Document clustering algorithms average metrics with standard deviation after 100 executions. 78

Figure 32 – The CBR-ILP-IR-SON-INT dataset after six iterations depicted on the Sankey Diagram. 80

Figure 33 – The expert’s feedback on the visual component utility. 81

Figure 34 – One insight reported by the expert. 82

LIST OF ALGORITHMS

Algorithm 1 – Vis-Kt’s general clustering framework.	64
Algorithm 2 – Proposed Framework based on NLM.	67

LIST OF SOURCE CODES

Source code 1 – Pre-processing algorithm produced by the author. Python version: 3.6. 96

LIST OF TABLES

Table 1 – A summary of the identified main aspects of IDC systems. KT stands for keyterm-based clustering algorithms.	48
Table 2 – A summary of the identified main aspects of visual analytics systems for dynamic document clustering exploration.	61
Table 3 – A comparison between the results of predicting the PV representation with a pre-trained embedding and an embedding trained with the corpus.	66
Table 4 – Clustering aspects of each metric in the project.	75
Table 5 – Document processing time (in minutes) and representation silhouette score. .	76
Table 6 – Document clustering time (in minutes) of the proposed algorithm, iKMeans, and LDC.	78

LIST OF ABBREVIATIONS AND ACRONYMS

AMI	Adjusted Mutual Information
ARS	Adjusted Rand Score
BoW	Bag-of-Words
BRT	Bayesian Rose Tree
CBR	Case-Based Reasoning
DAG	Directed Acyclic Graph
FCM	Fuzzy C-Means
FDP	Force-Directed Placement
FMI	Fowlkes-Mallows Index
HDP	Hierarchical Dirichlet Process
IDC	Interactive Document Clustering
ILP	Inductive Logic Programming
IR	Information Retrieval
IVC	Interactive Visual Clustering
LDA	Latent Dirichlet Allocation
LDC	Lexical Double Clustering
LSP	Least Square Projection
MDS	Multidimensional Scaling
NMF	Non-negative Matrix Factorization
PCA	Principal Component Analysis
PV	Paragraph Vector
PV-DM	Distributed Memory Model of Paragraph Vectors
SON	Sonification
t-SNE	t-Distributed Stochastic Neighbor Embedding
TFIDF	Term Frequency – Inverse Document Frequency
TRT	Topic Rose Tree
WV	Word Vector

CONTENTS

1	INTRODUCTION	25
1.1	Motivation	26
1.2	Contributions	27
1.3	Outline	27
2	DOCUMENT CLUSTERING WITH VISUAL SUPPORT	29
2.1	Document Representation	29
2.1.1	<i>Bag-of-Words Model (BoW)</i>	30
2.1.2	<i>Neural Language Models (NLM)</i>	31
2.2	Document Projection	33
2.2.1	<i>Principal Component Analysis (PCA)</i>	34
2.2.2	<i>Force-Directed Placement (FDP)</i>	34
2.2.3	<i>Least Square Projection (LSP)</i>	35
2.2.4	<i>t-Distributed Stochastic Neighbor Embedding (t-SNE)</i>	36
2.3	Interactive Document Clustering	37
2.3.1	<i>Interactive Topic Modeling</i>	37
2.3.1.1	<i>Latent Dirichlet Allocation (LDA)</i>	37
2.3.1.2	<i>Non-negative Matrix Factorization (NMF)</i>	40
2.3.2	<i>User-driven keyterm-based clustering</i>	42
2.3.2.1	<i>Lexical Double Clustering (LDC)</i>	43
2.3.2.2	<i>Interactive K-means (iKMeans)</i>	44
2.3.3	<i>Visual Clustering</i>	45
2.4	Discussions and final considerations	47
3	VISUALIZATION OF DYNAMIC DOCUMENT CLUSTERS	49
3.1	Initial Considerations	49
3.2	Content-based Dynamic Visualization	49
3.2.1	<i>Flat visualizations</i>	50
3.2.2	<i>Hierarchical visualization</i>	53
3.3	Citation-based Dynamics Visualizations	57
3.4	Discussion and Final Considerations	59
4	THE PROPOSED FRAMEWORK	63
4.1	Initial Considerations	63

4.2	The Vis-Kt System	63
4.3	Proposed Framework	64
4.3.1	<i>Document processing</i>	65
4.3.2	<i>Clustering Algorithm</i>	66
4.3.3	<i>Interaction History Visualization</i>	68
4.4	Discussions and limitations	70
5	EXPERIMENTS AND DISCUSSIONS	73
5.1	Initial Considerations	73
5.2	The Data	74
5.3	Quantitative Evaluation	74
5.3.1	<i>Document Representation</i>	75
5.3.2	<i>Document Clustering</i>	76
5.4	Usage Scenarios	79
5.4.1	<i>Incremental Clustering Analysis</i>	79
5.4.2	<i>Expert Study: Clustering Evolution with User Interactions</i>	80
5.5	Final Considerations	81
6	CONCLUSION AND FUTURE RESEARCH	83
	BIBLIOGRAPHY	85
APPENDIX A	DOCUMENT PRE-PROCESSING ALGORITHM	95

INTRODUCTION

Text is a strategic form of data encoding in many applications. Daily, countless volumes of documents are generated. Accessing and interpreting the content of these documents is a crucial task in the routine of many professionals. For instance, a researcher may be interested in mapping the main concepts of her literature review. In this scenario, *Text Mining* techniques seek to process and extract pieces of information of large text corpora. Among the Text Mining techniques, we highlight the clustering algorithms, where documents are grouped by their content similarity. Document clusters are generally depicted by their most relevant set of terms or topics.

As can be seen in the literature (e.g. [Alencar, Oliveira and Paulovich \(2012\)](#) and [Federico et al. \(2017\)](#)), unsupervised clustering techniques are popular and effective on the task of mining textual data. However, even a unsupervised document clustering algorithm with state-of-the-art performance may produce clusters that do not make sense to the end-user ([HU et al., 2014](#)). A reasonable solution is to introduce the user in the clustering loop. With this approach, the user can draw insight into the corpus with user-intuitive interaction tools within the clustering process.

In addition to clustering techniques, *Information Visualization* techniques play an important role in depicting complex data as abstract visual components. This field of knowledge seeks to take advantage of the human ability to identify and interpret visual patterns, reducing the cognitive overload required to make sense of large collections of complex data ([MUNZNER, 2014](#)), such as collections of textual documents. To take full advantage of the human cognitive ability, we use interactions with the visualization to allow the user to explore and gradually obtain more knowledge about the data. The analytical reasoning employed in Information Visualization techniques is known as *Visual Analytics*.

Humans find it easier to recognize a piece of information than describe something from scratch ([RUOTSALO et al., 2014](#)). Also, according to [Avons, Wright and Pammer \(1994\)](#), it is easier for the user to make sense of information relative to a reference point than in

isolation. Therefore, we can define *Information Discovery* as an incremental task where the user progressively learns about the content of the corpus by exploring it and drawing her insights.

In this context, we can identify four scenarios related to the data and the analysis of a corpus. The corpus can be static, where the document collection does not change during the analysis; Or it can be dynamic, where the document collection changes by user interference or by new streaming documents. The analysis can be static, where its goal is to visualize the data; Or it can be dynamic, where the interactions affect the corpus and the clustering algorithm.

1.1 Motivation

The visual analytics system Vis-Kt (SHERKAT; MILIOS; MINGHIM, 2019) proposes an effective keyterm-based technique where the user interacts with the top keyterms discovered by a keyterm-based clustering algorithm, and can tailor the results of the cluster as her understanding of the dataset evolves. By employing keyterms as basis for content interaction and clustering, the approach is intuitive for a varied set of end users.

One of Vis-Kt's main drawbacks, however, is the use of the Bag-of-Words (BoW) model as its document representation technique and Term Frequency - Inverse Document Frequency (TFIDF) (JONES, 1972) to capture the terms' importance to a document in a corpus. That impairs, to a certain extent, two important follow-ups in text analysis by clustering: the speed up needed to process incremental document sets and the analysis of time-related issues in the data set.

In most approaches based on BoW, the document is represented as the TF-IDF score for each term (or n-gram) in each document. Since larger datasets tend to have large vocabularies, it is common to use feature selection techniques to reduce the dimensionality of the document representation. Vis-Kt uses the mean-TDIDF as the feature selection technique (NOURASHRAFED-DIN *et al.*, 2018). However, the resulting document representation is still very sparse.

Alternatively, the vector representation of the documents using neural language models (NLM) (BENGIO *et al.*, 2003; MIKOLOV *et al.*, 2013a) does not increase with the size of the vocabulary. The document vector is dense and has a fixed dimensionality. In addition, the fixed-dimensionality representation allows incremental handling without redefining the vector representation to account for new terms introduced by new documents, when we are dealing with streaming text data.

Since information discovery is an incremental task, the system also lacks a visualization of interaction history, depicting each clustering structure sequentially. With this kind of visual component, we could depict the unsupervised clustering of temporal data and the clustering after each user interaction.

1.2 Contributions

To enhance Vis-Kt’s performance and to make it scalable to larger and incremental datasets, this study proposes a novel approach to keyterm-based visual clustering employing neural language models yet preserving its original data processing pipeline. The proposed algorithm takes advantage of the mathematical vector operations in NLM.

We report two main contributions:

- A novel interactive keyterm-based document clustering algorithm designed for NLM.
- A visual component based on the Sankey diagram to depict document evolution flows throughout several clustering iterations.

1.3 Outline

This document is organized as follows:

- In [Chapter 2](#), we describe a short background on interactive document clustering and its fundamental concepts, such as document representation models, multidimensional projection techniques, and clustering algorithms. Later, we map the literature on interactive document clustering relative to the described techniques.
- In [Chapter 3](#), we present a brief survey on related work on evolutionary visualization applied to dynamic document clustering and topics.
- In [Chapter 4](#), we present the proposed clustering algorithm with the new representation model, and the visual component employed to depict the interaction historic. Next, we discuss some known limitations and possible enhancements to future works.
- In [Chapter 5](#), to validate our proposed algorithm, we report a quantitative evaluation with publicly available datasets comparing the former against the novel clustering algorithm. We performed an expert analysis to qualitatively evaluate the visual component’s usefulness.
- In [Chapter 6](#), we present our conclusions and future work.

DOCUMENT CLUSTERING WITH VISUAL SUPPORT

The following sections provide an overview of the Interactive Document Clustering (IDC) field, describing aspects, such as the document representation models, document projection techniques, and clustering algorithms. Additionally, this chapter presents a brief literature mapping in user-driven clustering visual analytics systems, and some considerations on the literature gap in IDC.

2.1 Document Representation

In document-based machine learning or clustering algorithms, the document representation is the first concern since most algorithms use fixed-size vectors. Document representations should encode its content, context and semantic information in a way that the document representation can be used to find relevant information about the corpus, such as groups, topics, and concepts.

Several steps of preprocessing are needed to convert a document written in natural language into a computer-friendly representation, which is generally a fixed-size vector. In the first step, we clear the text by removing punctuation and numbers. Additionally, it is common to transform all terms to lower case characters, otherwise strings like “Student” and “student” will be two different representations of the same word.

The following step is generally known as stop-word removal, where we remove devoid-of-meaning words (e.g., “of”, “the” and “a”), or words without relevance to a particular context. For instance, the words “References” and “Abstract” have no relevance in the context of scientific documents since they generally occur through all the documents.

An advanced preprocessing task is related to handling every word inflection as a single

word representation. For instance, the words “studies”, “studied”, and “studying” carry the same meaning and should be conflated into a single representation. Stemming and lemmatization techniques can be employed to achieve this goal. The former is a simpler technique that removes the words’ suffixes (PORTER, 2006). The stemming of the previous example would be “studi”. Whereas lemmatization is a more elaborated technique where the words are transformed into their base or dictionary form without any inflection. The lemmatization of the previous example would be “study”.

The *Bag-of-Words model* (HARRIS, 1954) and recently the word embeddings (MIKOLOV *et al.*, 2013a; MIKOLOV *et al.*, 2013b; LE; MIKOLOV, 2014) are two of the most popular techniques for document representation. There are variations of these first two techniques but we decided to cover only the main ones, which we shall describe in the following sections.

2.1.1 *Bag-of-Words Model (BoW)*

A popular approach for a long time is the Bag-of-Words (BoW) or *bag-of-n-grams* model, which represents a document as a vector of term frequencies. BoWs became popular due to its simplicity, efficiency, and sufficient accuracy. One of the main issues with the BoW model is that only the term frequency does not reflect the real importance of a word to a document nor the corpus. For instance, the word “visual” would have a high frequency in a corpus of Information Visualization documents, and thus, has no relevance since a large percentage of documents will contain this term. Moreover, some low-frequency terms can help to discriminate a document’s content. This implies that a form of weighting is necessary to enhance the BoW model.

An approach to tackle this issue is the *Inverse Document Frequency* (IDF), which is a statistical interpretation of term specificity conceived by JONES (1972), which decreases high-frequency terms scores and increases low-frequency terms scores. This suggests that a rare term has a high IDF, whereas a frequent term has a low IDF. In Equation 2.1, let DF_t be the number of documents within the corpus that contain the term t , and N be the total number of documents in the corpus. Then, the IDF of each term is calculated as:

$$IDF_t = \log \frac{N}{DF_t} \quad (2.1)$$

Finally, in Equation 2.2 the Term Frequency – Inverse Document Frequency (TFIDF) score of each term t for each document d can be calculated as follows:

$$TFIDF_{t,d} = TF_{t,d} \times IDF_t \quad (2.2)$$

Besides this issue, this model has several disadvantages. For instance, the word order is ignored. Therefore, different documents can have the same representation as long as they contain the same words and the same frequencies. To partially address this concern, the bag-of-n-grams

model considers a short context as word n -grams of size n . The BoW model suffers from data sparsity and high dimensionality, which gets even worse in the bag-of- n -grams model. Also, the BoW model does not capture the semantics of the documents in the case when they use different words to represent similar meanings (vocabulary mismatch in similar documents). Even with these downsides, the BoW is still a very popular approach and the TFIDF is the most used weighting technique (TURNERY; PANTEL, 2010).

Several feature selection techniques based on the TFIDF score can be employed to tackle this issue. For instance, Sherkat, Milios and Minghim (2019) employed the mean-TFIDF technique to filter the terms that will be used as features to represent the documents. This dimensionality reduction technique removes the terms that have a TFIDF below the average score of all terms. In Equation 2.3, let D be the corpus. The mean-TFIDF of each term t is calculated as:

$$mean_TFIDF_{t,D} = \frac{1}{|D|} \times \sum_{d \in D} TFIDF_{t,d} \quad (2.3)$$

According to Liu *et al.* (2003) and Yang and Pedersen (1997), feature selection techniques based on term frequency are reported to be as effective as more complex methods while having linear time complexity for unsupervised selection techniques (SHERKAT, 2018).

2.1.2 Neural Language Models (NLM)

With the recent progress of machine learning techniques, Mikolov *et al.* (2013b) introduced the Word Vector (WV) model, which uses neural networks to learn vector representations of words. In this model, every word is mapped to a unique vector, represented as a column in a matrix W . The W matrix contains the unique vector of each word in the vocabulary. Let k be the maximum word distance from a center word, the concatenation or sum of the word vectors within a context of size $2 * k + 1$ is used to predict the word vector in the center position of the sliding window. The process is illustrated in Figure 1.

$$\frac{1}{T} \sum_{t=k}^{T-k} \log p(w_t | w_{t-k}, \dots, w_{t+k}) \quad (2.4)$$

$$\log p(w_t | w_{t-k}, \dots, w_{t+k}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_i}} \quad (2.5)$$

$$y = b + Uh(w_{t-k}, \dots, w_{t+k}; W) \quad (2.6)$$

Formally, given a sequence of training words $\{w_1, w_2, \dots, w_T\}$, the objective of the skip-gram model is to maximize the average log probability in Equation 2.4, where the prediction p is done by the softmax function in Equation 2.5. Finally, each of y_i is the un-normalized

log-probability for each output word i , computed by Equation 2.6, where h is constructed by the average of word vectors within the skip-gram window extracted from W . U and b are the softmax function's parameters.

Figure 1 – The framework for learning word vectors. In the figure, the sliding window is of size one and it has the context of three words (“the”, “cat” and “sat”), which is used to predict the fourth word (“on”). The input words are mapped to columns of the matrix W to predict the output word.

Source: [Le and Mikolov \(2014\)](#).

As a result, the trained model maps words into a vector space in which semantically similar words have spatially closer vector representations. Also, it encodes the inherent relationship among words, which allows the execution of mathematical operations in the word representations. For instance, $v(\text{“Madrid”}) - v(\text{“Spain”}) + v(\text{“France”}) \approx v(\text{“Paris”})$, where $v(\text{“word”})$ is the vector representation for “word”.

The studies that followed Mikolov’s word embeddings have tried to generalize it to sentences, paragraphs, and even full-length documents.

Figure 2 – The paragraph vector acts as a memory of the missing information in the current context (paragraph). And similar to Figure 1, this framework’s goal is to predict the fourth word by average or concatenation of the vector representation of the contained in the given sliding window context.

Source: [Le and Mikolov \(2014\)](#).

The Paragraph Vector (PV) model (LE; MIKOLOV, 2014) is an extension of the WV. The Distributed Memory Model of Paragraph Vectors (PV-DM) maps each paragraph to a unique vector in matrix D and each word is mapped also to a unique vector in matrix W . Next, PV and WV are averaged or concatenated to predict the next word in a context, Le and Mikolov (2014) report that concatenating the vectors results in better predictions. As the model slides through the text, each PV acts as a WV that memorizes the missing words in the current context. It constructs vector representations of variable-length texts (see Figure 2). The PV method has the same advantages as its predecessor, semantically similar documents are represented closer in the vector representation.

Figure 3 – In the PV-DBOW model, the PV is trained to predict the words in a small context.

Source: Le and Mikolov (2014).

The PV also has a simpler second model called *Distributed Bag of Words* (PV-DBOW) and its goal is to predict words in the current paragraph (see Figure 3). This model ignores the context words and tries to predict words randomly sampled from the paragraph. Since it only stores the softmax weights, this approach requires less memory than the PV-DM model which stores both the softmax weights and the word vectors. As reported in Le and Mikolov (2014), PV-DM alone usually has a state-of-art performance by its publishing date, but its combination with the PV-DBOW is usually more consistent in the experiments executed.

2.2 Document Projection

In the information discovery process, an overview of the data collection can be very useful to recognize local and global pieces of information. However, in the context of document collections, we have seen only high-dimensional vector representations. We use *Dimensionality Reduction* techniques to tackle this issue. These techniques aim to find lower-dimensional representations of the high-dimensional data while preserving as much as possible of the relationships defined in the original data.

There are several dimensionality reduction techniques. However, since describing each technique is beyond the scope of this work, we have decided to cover only the techniques which are most relevant to our studies and to the document visualization field.

2.2.1 Principal Component Analysis (PCA)

The Principal Component Analysis (PCA) (JOLLIFFE, 2002) is a dimensionality reduction technique that finds orthogonal linear combinations called *Principal Components* (PC), which seeks to encode data variability. The first PC is the linear combination with the greatest variance. The following PCs are the linear combinations, orthogonal to the previous, which have the greatest variance, and so on. There are as many PCs as the original features' number, but generally, the first PCs can capture most of the data variance.

PCA has two main drawbacks. First, it does not scales to higher-dimensional data, since the data covariance calculation is scaled by the data dimensionality. Second, the technique seeks to maximize the global pairwise distance among the data points. However, in some cases, it might be important to prioritize the data points' neighborhood, as in the case of projecting clusters.

PCA is employed in several scenarios, from dimensionality reduction as initialization for other techniques (later described in section 2.2.4) to visual analytics systems, such as *iPCA* (JEONG *et al.*, 2009). The latter is a visual analytics system that allows the user to interact with PCA parameters, such as eigenvalues, eigenvectors, and the number of principal components. Given that the user has a thorough understanding of PCA, this set of interactions may allow gain insights into the dataset.

2.2.2 Force-Directed Placement (FDP)

The Force-Directed Placement (FDP) (FRUCHTERMAN; REINGOLD, 1991) is a technique that models the data as a linked graph, where edges represents a node neighborhood. FDP tries to reduce the number of cross-edges based on two different forces: the attractive force between two neighboring nodes, and the repulsive force between two nodes.

In the first iteration, the nodes' positions are initialized randomly. The following iterations try to minimize the overall energy of all nodes, which is defined by Equation 2.8. Each node's individual force is defined by Equation 2.7, where $\Gamma(i)$ is the set of neighbors of node i , X is the set of nodes, C and K are parameters for regulating the attractive and repulsive forces. The forces are minimized iteratively by moving nodes according to their forces' direction.

$$f(i, X, K, C) = \sum_{i \neq j} -\frac{CK^2}{\|x_i - x_j\|^2} (x_j - x_i) + \sum_{j \in \Gamma(i)} \frac{\|x_i - x_j\|}{K} (x_j - x_i) \quad (2.7)$$

$$Energy(X, K, C) = \sum_{i \in |X|} f^2(i, X, K, C) \quad (2.8)$$

To employ FDP to document projections, the nodes' forces are calculated proportionally to the dissimilarities $\gamma(x_i, x_j)$ between the data points in the original feature space and the distance $d(y_i, y_j)$ in the generated space (PAULOVICH *et al.*, 2008).

2.2.3 Least Square Projection (LSP)

The Least Square Projection (LSP) (PAULOVICH *et al.*, 2008) is a multidimensional projection technique that seeks to preserve neighborhood relationships. It relies on control points that are representative of the data distribution.

LSP's first step is to define good control points. Later, it projects the control points with a Multidimensional Scaling (MDS) technique (TORGERSON, 1952). With this step, the projection captures global pieces of information relative to the control points, which enforces the importance of selecting good ones.

Next, it interpolates the remaining data points through a sparse linear system, which tries to place each data point in the convex hull of its nearest neighbors. The number of neighbors to preserve is an adjustable parameter that has a direct effect on the projection. The higher the number of neighbors, the denser will be the clusters (see Figure 4). The steps that follow the control points projection are incremental, which means that the points can be placed on demand.

Figure 4 – The outcome of the number of neighbors in the LSP technique. In the figure, 675 documents related to the computer science field.

Source: Paulovich (2008).

Since it projects only the control points and interpolates all the remaining points, LSP shows a fast processing time and a satisfactory precision, especially in nonlinear sparse spaces, such as the case for documents represented with the BoW model.

In the context of visual analytics for document exploration, *Information Retrieval* (DIAS; MILIOS; OLIVEIRA, 2019) and exploratory topic modeling (HEIMERL *et al.*, 2016b) systems employed LSP to project documents in a 2D plane to highlight groups of documents.

2.2.4 *t*-Distributed Stochastic Neighbor Embedding (*t*-SNE)

The *t*-Distributed Stochastic Neighbor Embedding (*t*-SNE) (MAATEN; HINTON, 2008) technique is an unsupervised nonlinear dimensionality reduction technique that tries to minimize the divergence between the data points distribution in the original feature space to their corresponding data points in the projection space. Unlike linear techniques such as PCA, which seeks to maximize variance and preserves large pairwise distances, nonlinear techniques such as *t*-SNE focus on preserving small pairwise distances or local similarities.

In the experiments reported in Maaten and Hinton (2008), the authors applied PCA to truncate the number of dimensions to 30, which reduced the noise and speeded up the processing. The results have shown that *t*-SNE outperforms other dimensionality reduction techniques in terms of groups segregation.

In Figure 5, is illustrated a comparison between *t*-SNE, PCA and FDP.

Figure 5 – 675 PV-DM vectors projected with three different dimensionality reduction techniques. The dataset has four categories.

Source: Elaborated by the author.

2.3 Interactive Document Clustering

Clustering algorithms play an important role in information discovery in large text corpora. However, even the best unsupervised clustering algorithm does not reflect the user's perspective about the corpus (HU *et al.*, 2014; SHERKAT *et al.*, 2018). Additionally, the desired clustering might be different for users with different needs, which are not taken into account by an unsupervised clustering algorithm.

Semi-supervised clustering algorithm with user interaction has been adopted to address this concern. By involving the user in the loop of clustering, the chances that the clustering algorithm results reflect the user's perspective of the corpus are increased (AWASTHI; BALCAN; VOEVODSKI, 2014; BALCAN; BLUM, 2008; HU *et al.*, 2014; SHERKAT *et al.*, 2018).

Visual analytics employed to text mining bind statistical models, visual resources, and the human cognition for users to gain insight into complex and unstructured data. Such systems play a role as an intermediary between the user and the algorithms, which can help the user digest complex analytical results. And thus, the user will be interacting with an intuitive visual metaphor, instead of interacting with a complex text analysis algorithm.

The user interactions with the clustering are limited by the algorithm itself, by the document representation and by the visualization used to represent and interact with the algorithms.

The first limitation is relative since each clustering technique has its nuances and its purpose. The second one was already discussed in [section 2.1](#). And the third limitation is a design choice. The visualization must be as intuitive as possible, and it must support user-interactions with the algorithms without losing its inherent purpose, and to interact with the data without losing its latent pieces of information.

2.3.1 Interactive Topic Modeling

Topic modeling is a well-known probabilistic model, which assumes that each document within a corpus belongs to multiple topics with different degrees of membership. Whereas, each term is also related to multiple topics with different degrees of membership. A topic modeling algorithm has as its output the membership probability of each document to each topic, and of each term to each topic. Another way to look at topic modeling is by considering the most probable topic as the document cluster label, and considering the probability distribution of documents over topics as soft clustering.

2.3.1.1 Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) (BLEI; NG; JORDAN, 2003) is a popular technique for topic modeling. It models topics as distributions over words and documents as distributions over topics. The idea behind LDA's generative process is based on the co-occurrence of terms in

the documents. For instance, if the term t_i has a high probability of belonging to topic c_i , and the terms t_j and t_i co-occurs with a high frequency, the term t_j has a high probability of being related to topic c_i .

Given a document-term matrix with N documents and M terms as input, let k be the number of topics. For each document, the probability distribution over topics is a multinomial distribution drawn from a Dirichlet distribution with parameter α ; and for each topic, the probability distribution over terms is a multinomial distribution drawn from a Dirichlet distribution with parameter β . The LDA algorithm results in two matrices: a document-topic matrix with dimensionality N by k and a topic-term matrix with dimensionality k by M .

A graphical model of the LDA process is depicted in Figure 6, where each random variable is represented by a circle. The arrows indicate that the pointed variable value depends on the value of the pointing variable. The values within each plate indicates the number of times the sampling should be performed.

Figure 6 – A graphical model of the LDA algorithm. Each plate represents a loop. The outer plate represents the M documents, while the inner plate represents the repeated choice of topics and words within a document. The circles represent the random variables, except by the shaded circle, which stands for the observed variable.

Source: Blei, Ng and Jordan (2003).

The LDA algorithm has several disadvantages. It will favor general topics over specific topics. For instance, if the corpus has topics such as “classification”, “clustering”, and “data mining”, and they share the same terms, LDA will favor the generation of a general topic of “machine learning”, which contains all three topics. Another downside is that since it is a statistical approximation, it shows inconsistency after several executions, and, from a visual analytics perspective, this inconsistency can be highly unfavorable to the information discovery process.

To enhance the automatic process of text clustering, the visual analytics system *iVisClustering* (LEE *et al.*, 2012) includes the user in the LDA algorithm loop by allowing interactions with the term weights of the algorithm. The interaction and its results can enhance user’s knowl-

edge about the corpus, and thus increase the probability of the algorithm's results to improve after each user interaction. The system provides several visual tools for interaction, such as parallel coordinates (INSELBERG, 1985) for soft clustering visualization, a popular approach formerly seen in *ParallelTopics* (DOU *et al.*, 2011). A force-based graph drawing is used to place the documents spatially. Also, to overcome the ambiguous results of the LDA algorithm, the system has a hierarchical topic explorer tool, where the user can apply merge, split and remove operations. Such operations can be listed as useful in interactive clustering systems (BALCAN; BLUM, 2008). The system was validated with information visualization papers gathered in the *Jigsaw dataset*¹, and with newsgroup articles from *20NewsGroup* (LANG, 1995). Four scenarios were evaluated: 1) Noise data filtering; 2) Topics refinement; 3) Merge and splitting operations; 4) Document exploration.

Figure 7 – An overview of the iVisClustering system exploring the Jigsaw dataset. In the figure: (A) graph view of the document clusters represented by colored nodes, and document similarity represented by edges; (B) hierarchical view of the user defined topics; (C) the cluster colors and the topic terms within; (D) parallel coordinates to depict the current document distribution over topics; (E) interactive view of the term weights; (F) a heatmap view where is depicted the number of documents that switched it's cluster membership in the last interaction; and (G) the document's content viewed as plain text and with the top keyterms highlighted.

Source: Lee *et al.* (2012).

An overview of iVisClustering is depicted in Figure 7. As a downside, iVisClustering inherits LDA's inconsistency after several executions.

¹ <https://www.cc.gatech.edu/gvu/ii/jigsaw/datafiles.html>

2.3.1.2 Non-negative Matrix Factorization (NMF)

Given a matrix $V \in \mathbb{R}_+^{m \times n}$, and an integer $k \ll \min(m, n)$, Non-negative Matrix Factorization (NMF) (PAATERO; TAPPER, 1994) is a process to find a decomposition of V into two matrices $W \in \mathbb{R}_+^{m \times k}$, and $H \in \mathbb{R}_+^{k \times n}$, by optimizing the distance d between V and the dot product WH . This decomposition is illustrated in Figure 8. NMF is generally formulated in terms of the Frobenius norm, which is an extension of the Euclidean norm to matrices, as can be seen in Equation 2.9.

$$\min_{\{W, H\} \geq 0} ||V - WH||_F^2 \quad (2.9)$$

In the context of topic modeling, V is a TFIDF weighted matrix, and the factor matrices W and H represent the term-topic and topic-document relationships, respectively. Where $v_i^{m \times 1}$ is the i -th column of V , which represents the i -th document TFIDF weighting over m terms. The integer k stands for the number of topics; $w_j^{m \times 1}$ is the j -th topic represented as weighted combinations of m terms; and finally, $h_i^{k \times 1}$ is the i -th document represented as a weighted combination of k topics.

Figure 8 – A graphical model of the NMF decomposition matrices.

Source: Wikipedia (2020).

NMF has the same output matrices as LDA: (W) a distribution of topics over terms matrix, and (H) a distribution of documents over topics matrix.

From the same research group as the iVisClustering system, *UTOPIAN* (CHOO *et al.*, 2013a) is a variation of its predecessor. It uses a semi-supervised version of the NMF instead of LDA for topic modeling. The NMF technique is chosen to handle the inconsistency issue in iVisClustering, caused by the LDA algorithm. The system has most of the same visual components as its predecessor, with the addition of a tailored version of t-SNE in its graph view (see Figure 9). The modifications in the original t-SNE aim to better show clustering structure in noisy datasets by modifying the pairwise distance of documents with same cluster label.

However, both iVisClustering and *UTOPIAN* suffer from the complexity of its algorithms, and thus user interaction with the algorithms is neither straight-forward nor intuitive. Also, they both suffer from the empirical convergence problem, which can be defined as how fast the algorithm converges from a human's practical viewpoint.

In a broader context, Choo *et al.* (2013b) propose an interactive visual testbed system for dimensionality reduction and clustering of a variety of data domains, such as images, documents,

Figure 9 – An overview of UTOPIAN exploring the Jigsaw dataset. In the figure, the graph view is a tailored version of t-SNE projection. The user interactions are numbered as: 1) topic merging; 2) topic creation by document-supervision; 3) topic splitting; and 4) topic creation by keyterm supervision. The other two visual components are inherited from iVisClustering (Figure 7).

Source: Choo *et al.* (2013a).

and any other data encoded in vectors. In the testbed system, the user can switch on demand between several clustering and dimensionality techniques. This means that the user has a set of combinations between clustering and projection techniques.

The user can compare the results between clustering techniques. However, she cannot interact with the clustering algorithm to, for instance, change the topic modeling term weights, as can be seen in the iVisClustering system. To enhance the projections, the system executes the dimensionality reduction for dimensions greater than 2D and uses parallel coordinates to compare the group segregation on several dimensions, as can be seen in Figure 10E.

One drawback of the NMF approach is that it requires significant computation resources, and it has a long processing time. This impairs to some extent the user interaction due to delay in the visualization results. To tackle such an issue, the system *TopicLens* (KIM *et al.*, 2017) proposes an extension of the hierarchical NMF proposed by (KUANG; PARK, 2013), resulting in considerable improvement in the processing time. Additionally, TopicLens employs a *Focus+Context* (MUNZNER, 2014) approach to allow the user to view the corpus projection through a *Interactive Lens* metaphor, where the system applies some processing on the focused area to improve the exploratory task.

The system was validated with two usage scenarios: information visualization related papers and The New York Times articles. An overview of the system is depicted in Figure 11

Figure 10 – An overview of the testbed system, displaying several modules. (A) Pre-processing parameters. (B) Clustering techniques. (C) Dimensionality reduction techniques. (D) Alignment techniques to align clustering and projection results. (E) Parallel coordinates used to selected two dimensions for projection. (F) Scatter plot view. (G) Topic and cluster view. (H) The original data content.

Source: Choo *et al.* (2013b).

2.3.2 User-driven keyterm-based clustering

Given the constraints of the visual analytics techniques in text analysis, Nourashrafeddin *et al.* (2018) categorizes the interactive document clustering approaches as: 1) Document supervision, 2) keyterm supervision, and 3) Hybrid supervision. As can be seen in the recent literature (HU; MILIOS; BLUSTEIN, 2011; HU; MILIOS; BLUSTEIN, 2012; NOURASHRAFEDDIN *et al.*, 2018; SHERKAT *et al.*, 2018), the keyterm supervision is an intuitive and efficient approach since it requires little user effort in comparison with the other approaches.

In the keyterm-based supervision, the user interacts with the most relevant terms in the corpus for information discovery, which does not require reading any document. This is a progressive task since the user enhances her knowledge about the corpus after each interaction, and thus improves the discovered latent clusters.

To avoid inconsistency after several interactions, the empirical convergence problem, and interaction limitations due to complex mathematical models previously present in the literature, the system *Vis-Kt* (NOURASHRAFEDDIN *et al.*, 2018; SHERKAT *et al.*, 2018) is an alternative that uses document clustering algorithms instead of topic modeling algorithms. One of the proposed algorithms is *Lexical Double Clustering* (NOURASHRAFEDDIN *et al.*, 2018), which overcame state-of-the-art algorithms such as LDA and NMF in several datasets. A second

Figure 11 – An overview of TopicLens exploring information visualization related papers. In the figure, the data points are colored relative to their clusters (topics). The top terms to each topic is also projected in the center of each cluster. Highlighted in the middle is the lens approach, where the user can focus on a rectangular area. A sub-topic modeling is executed in the focused area, enhancing the exploration.

Source: Kim *et al.* (2017).

proposed algorithm is *iKMeans* (SHERKAT *et al.*, 2018). In the system, both algorithms are interoperable. The idea behind LDC and *iKMeans* is that, before finding document clusters, it is better to focus on term clusters and the keyterms that represent topics meaningful to the user (NOURASHRAFEDDIN *et al.*, 2018).

Vis-Kt has two types of document projections: a tailored version of t-SNE and a force-directed graph. In both cases, document similarity is represented as edges. Like in *iVisClustering* (LEE *et al.*, 2012) and *ParallelTopics* (DOU *et al.*, 2011), the soft clustering probability distributions are represented by *parallel coordinates* (INSELBERG, 1985). The term clusters are depicted by colored boxes with its top terms listed vertically, where the user can re-order the terms, remove, or add new terms. The top terms relevance to each cluster is represented by a bar chart. Vis-Kt also has a *WordCloud* (VIÉGAS; WATTENBERG, 2008) to depict the top terms of each cluster or each document.

However, Vis-Kt does not support dynamic datasets nor an analysis of the clustering changes after each interaction. The system's pipeline is detailed in Figure 12.

In the following sections we describe the LDC and *iKMeans* clustering algorithms.

2.3.2.1 Lexical Double Clustering (LDC)

The Lexical Double Clustering (LDC) algorithm consists of two steps. The first step is term clustering, and the second step uses the term clusters of the previous step to create distilled set of terms to guide the document clustering task. The soft clustering algorithm Fuzzy C-Means (FCM) (BEZDEK, 1981) is employed to cluster terms, with terms may belonging to more than

Figure 12 – An overview of Vis-Kt’s pipeline.

Source: Elaborated by the author.

one cluster. Fuzzy C-Means aims to optimize the objective function in Equation 2.10.

$$FCM = \sum_{i=1}^{|W|} \sum_{j=1}^k u_{ij}^z \|w_i - c_j\|^2, \quad 1 < z < \infty \quad (2.10)$$

Let $D \in \mathbb{R}^{m \times n}$ be a TFIDF weighted document-term matrix. Let W be the set of terms, $w_i^{m \times 1}$ is the i -th column of D and represents the i -th term as its TFIDF score over the m documents (see Figure 13). Let $C \in \mathbb{R}^{m \times k}$ be a matrix of k cluster centers, and $c_j^{m \times 1}$ is the j -th cluster center. The similarity between each term and term cluster centers is calculated with the *cosine distance*. Let $U^{n \times k} = \{u_{ij} \in \mathbb{R}_+ | 0 \leq u_{ij} \leq 1\}$ be a cluster membership matrix, with u_{ij} standing for the degree of membership of the i -th term to the j -th cluster, as calculated by Equation 2.11.

$$u_{ij} = \frac{1}{\sum_{l=1}^k \left(\frac{\|d_i - c_j\|}{\|d_i - c_l\|} \right)^{\frac{2}{z-1}}} \quad (2.11)$$

The matrix U and the term cluster centers C are updated during each iteration of the FCM by optimizing the objective function in Equation 2.10. In the first iteration, the U matrix is randomly initialized. In the case of user interaction, U initialization is guided by the user feedback. Finally, given the terms distilled from the term clusters, the algorithm finds the document clusters related to each term cluster.

2.3.2.2 Interactive K-means (iKMeans)

To preserve Vis-Kt’s pipeline, and to demonstrate that Vis-Kt is independent of the clustering algorithm, Sherkat *et al.* (2018) proposed the interactive clustering algorithm named *iK-Means*.

Similar to LDC, this algorithm uses FCM to cluster terms in its first iteration, to find the top 5 terms of each cluster. In the next iterations, the top 5 terms for each cluster are provided by

Figure 13 – TFIDF weighted document-term matrix. In the figure, the red column stands for the i -th column.

	Terms				
Documents			TFIDF [1, i]		1
					:
			TFIDF [j, i]		j
					:
			TFIDF [m, i]		m
	0	...	i	...	n

Source: Elaborated by the author.

the user, which can also provide more terms. Later, the vector representations (w_i) of these 5 terms are averaged to find the terms' centers and to extend the list of top terms for each term cluster. As the number of terms provided by the user increases, the document cluster will be more biased by the user perspective.

One can consider each set of top terms in each term cluster as an imaginary document which contains only these terms. Let the average TFIDF score of these terms in all documents be the TFIDF score of terms in this imaginary document. The imaginary document of each term cluster is now considered the initial center of each document cluster.

The Vis-Kt system and its visual components are depicted in [Figure 14](#).

2.3.3 Visual Clustering

Document projections are employed to encode the corpus' contents and relationships in a lower-dimensional representation. In the context of projections for local pairwise distance or clustering, the resulting data points are spatially relative to the content similarity, preserving data points neighborhoods.

Despite many efforts to insert the user into the clustering loop, one thing to notice is that the current visual analytics systems for document clustering focus on the high-dimensional representation and use the projection only for visual feedback. The user interactions with the data are generally external to the visualization. This makes the system constrained by the underlying algorithms that, in most cases, the user has none or little understanding.

One possible solution is through Interactive Visual Clustering (IVC) ([DESJARDINS;](#)

Figure 14 – An overview of Vis-Kt. (A) Cluster Tree View: each cluster viewed as a hierarchical directory structure. (B) Document View: the content of the selected document. (C) Document-Cluster View: the probabilistic distribution of each cluster membership of the selected document viewed as Parallel Coordinates. (D) Graph view: in the figure, a t-SNE projection of 682 computer science related papers; and as an alternative to t-SNE, there is a second tab with a force-directed graph view. (E) Cluster View: each cluster name and its keyterms. (F) Term Cloud View: a WordCloud of the selected cluster. (G) Cluster Keyterms View: a histogram of the importance of each keyterm of the selected cluster. (H) Term-Cluster View: the probabilistic distribution of each cluster membership of the selected term viewed as Parallel Coordinates.

Source: Elaborated by the author.

MACGLASHAN; FERRAIOLI, 2007), which allows the user to interact with approximated representations of the data (e.g., dimensionality reduction), and interprets it to the high-dimensional representation. In this approach, the process is fully user-centered, with the user's feedback is interpreted as a literal task.

Sensemaking is a process where the user expands her knowledge about complex data through manipulations of its visual representation, experimenting parameter adjustments based on her domain expertise, and investigating hypotheses about the data (PIROLI; CARD, 2005). The *sensemaking* is composed of two tasks: foraging and synthesis. The former is the process of filtering and gathering collections of interesting and relevant information. The latter is the process of constructing and testing hypotheses about the foraged data's relationships and latent aspects.

Endert, Fiaux and North (2012b) address the *sensemaking* process through a visual framework called *Semantic Interactions*. It combines the foraging and synthesis tasks in a single approach, where user interactions with the visual representation are interpreted and converted as parameters to the clustering algorithm. In other words, the user draws her evolving knowledge,

and the system incrementally learns the user's perspective about the corpus' content.

The visual analytics prototype *ForceSPIRE* (ENDERT; FIAUX; NORTH, 2012b; ENDERT; FIAUX; NORTH, 2012a) and latter *LightSPIRE* (ANDREWS *et al.*, 2011; ENDERT *et al.*, 2012), allow the user to move document representations in a 2D space, highlight text, search and annotate documents. Such interactions are interpreted and converted into term re-weighting according to the user's analytical reasoning. This approach allows the user to intuitively interact with the data representation, not with the underlying complex algorithms.

These prototypes were designed for high-resolution displays (see Figure 15), which narrows the set of target users. The requirement for such hardware can be seen as a drawback when compared with the aforementioned works.

Figure 15 – ForceSPIRE's usage with high-resolution displays.

Source: Endert, Fiaux and North (2012a).

2.4 Discussions and final considerations

Table 1 summarizes the identified main aspects of an IDC system and compares the works mentioned in this chapter.

This brief literature review on Interactive Document Clustering shows that, although the BoW model has been widely employed for visual analytics, it has several drawbacks, namely:

- The vectorial representation is high-dimensional and sparse, which scales with the corpus' vocabulary. This impairs to some extent the employed visualizations.
- Naturally, it does not capture contextual pieces of information. To address this second issue, the bag-of-n-grams is adopted, which worsens the former drawback.
- The representation is non-incremental. To add a new document to the model, the matrix must be regenerated and the TFIDF scores re-weighted.

Table 1 – A summary of the identified main aspects of IDC systems. KT stands for keyterm-based clustering algorithms.

Paper	Representation		Clustering			Projection			History View
	BoW	NLM	LDA	NMF	KT	t-SNE	FDP	LSP	
Endert, Fiaux and North (2012a)	✓	✗	✗	✗	✓	✗	✓	✗	✗
Lee <i>et al.</i> (2012)	✓	✗	✓	✗	✗	✗	✓	✗	✓
Choo <i>et al.</i> (2013a)	✓	✗	✗	✓	✗	✓	✗	✗	✓
Choo <i>et al.</i> (2013b)	✓	✗	✓	✓	✓	✓	✓	✗	✗
Kim <i>et al.</i> (2017)	✓	✗	✗	✓	✗	✓	✗	✗	✗
Nourashrafeddin <i>et al.</i> (2018), Sherkat <i>et al.</i> (2018)	✓	✗	✗	✗	✓	✓	✓	✗	✗

Source: Research data.

Alternatively, neural language models have a fixed dimensionality dense vector representation; it captures contextual pieces of information through the skip-gram model; and, since it has a fixed dimensionality vector representation, it is an incremental model.

To the best of our knowledge, no previous work on visual analytics for IDC has employed neural language models as its document representation model. Despite its drawbacks, the BoW model still is the most used one.

Although the LSP projection technique has shown satisfactory results, t-SNE still has greater results. Sherkat *et al.* (2018) present an extension of t-SNE which is enhanced with the FDP technique.

The information discovery’s incremental aspect has little contributions in the IDC visual context. Lee *et al.* (2012) and Choo *et al.* (2013a) employ a heatmap visualization to depict a comparison between the previous interaction and the current state of the clustering structure (see Figure 7F). However, this approach is not intuitive nor scalable, since it only compares the current and the previous iterations with a fixed number of clusters. In Chapter 3, we cover some evolutive visualization techniques employed for text visualization, which can be used to depict the IDC’s incremental aspect.

VISUALIZATION OF DYNAMIC DOCUMENT CLUSTERS

3.1 Initial Considerations

A small percentage of the reviewed papers in [Chapter 2](#) have explored the visualization of clustering dynamics caused by changes over time or changes caused by user interaction. To address such a literature gap, we decided to provide a deeper review in this area.

The visualizations listed in the present chapter are mainly employed in the time-varying document scenario. However, these visual metaphors can also be employed in the context of dynamic clustering, since the time-steps of each interaction can be seen as timestamps in non-temporal data.

Although its underlying clustering algorithms are not content-based, the field of co-citation analysis over the years is widely explored. The literature shows several contributions to the information visualization field. Such visualization techniques are worth mentioning in this context.

3.2 Content-based Dynamic Visualization

As stated in [section 2.3.1](#), another way to look at topic modeling is by considering the most probable topic as the document cluster label, and considering the probability distribution of documents over topics as soft clustering. Since the visualization of clustering structures of interactive document clustering systems is poorly explored in the literature, we decided to summarize the dynamic topic visualization techniques, which has the same concept of “text flows”.

Understanding how topics in a corpus evolve and how they relate to each other over time

can be useful for the user to obtain new insights about the corpus' content without the need to read each document. For instance, two or more topics can merge into a single topic that contains keyterm from both topics, which configures a more generic topic. Alternatively, one topic can split into two or more topics. This means that the resulting topics have some keyterms from their parent topic, and these topics are more specific.

Proposed by [Havre et al. \(2002\)](#), the river metaphor is the most popular visualization to illustrate topic dynamics over time. This technique represents the trend of topics at each timestamp as waves. The horizontal axis encodes the time variance, oriented from left to right. The wave's vertical thickness encodes the topic frequency in each timestamp (see [Figure 16](#)). This technique is also known as *StreamGraph* or *Stack graph*. However, the river metaphor in its original proposal does not depict critical events, such as topic merge and split operations. Therefore, most variations that followed the original river metaphor aimed to tackle this limitation. The following sections present some of these variations.

Figure 16 – Overview of the ThemeRiver technique. The visualization depicts speeches, interviews, and articles from Fidel Castro from the end of 1959 to mid-1961.

Source: [Havre et al. \(2002\)](#).

Both topic modeling techniques and variations of the river metaphor have hierarchical alternatives. Generally, hierarchical techniques aim to simplify the topic modeling computation and the cognitive load requested to understand complex visualizations.

3.2.1 Flat visualizations

Text Insight via Automated Responsive Analytics (TIARA) ([WEI et al., 2010](#)) is a pioneer work in visual text analysis that supports the user in exploratory tasks in large text

corpora by combining topic modeling techniques with interactive visualizations. The system also extracts time-sensitive key words to summarize how each topic evolved. To evaluate this proposal, the authors employed TIARA in two usage scenarios: email summarization and medical report analysis. TIARA depicts the temporal topic trends in the traditional river metaphor fashion. The system counts both with topic modeling and clustering techniques.

Figure 17 – An overview of the TIARA system exploring approximately 8,000 emails. In the figure, each wave represents a topic, described as a set of keywords. (a) an input field where the user can specify a query to retrieve emails from the collection. (b) the content of the retrieved email listed. (c) StreamGraph of the retrieved emails. (d) more tools to interact with the topics besides the StreamGraph.

Source: [Wei et al. \(2010\)](#).

TIARA has inspired most of the following works on the temporal analysis of topics. However, TIARA neither allows the user to interact with the underlying topic modeling techniques nor permits new documents to be added. The system has only static corpus and analysis capabilities.

To address the soft-clustering visualization in topic modeling, *ParallelTopics* (DOU *et al.*, 2011) uses *Parallel Coordinates* (INSELBERG, 1985) to depict the probabilistic distribution of documents over topics. The system also projects the documents in a *Scatterplot* view (CHAMBERS *et al.*, 1983), and upon selection, a pie glyph describes the topical distribution to the selected document. A traditional river metaphor depicts the topical changes over time. An overview of the *ParallelTopics* system is illustrated in Figure 18.

Dou *et al.* (2011) report two scenarios: awarded scientific proposals of the *National Science Foundation* (NSF) from 2000 to 2010 (NSF, 2020), and proceedings of the IEEE VAST conference from 2006 to 2010. The experiments validated the system’s efficiency in four tasks:

Figure 18 – ParallelTopics system exploring information visualization related documents. (A) Document distribution over topics displayed in parallel coordinates. (B) Traditional river metaphor depicting topical dynamics from 2006 to 2010. (C) The extracted topics listed and described by their most relevant terms. (D) Scatterplot describing the documents in 2D.

Source: Dou *et al.* (2011).

(1) Revealing main topics, (2) Describing the documents' features, (3) Document analysis based on its number of topics, and (4) Describing the topic dynamics over the years.

However, Dou *et al.* (2011) also states that a general issue with topic modeling techniques is the decision on the number of topics that best describe the corpora. A badly chosen number of topics might lead to a bad representation of the corpus.

To identify critical events in evolving topical structures, such as how new topics emerge (topic birth), how topics disappear (topic death), how one general topic becomes two or more specific topics (topic splitting), and how two or more topics condense into one generic topic (topic merging), Cui *et al.* (2011) proposes *TextFlow*, a visual analytics system that depicts topic and keywords relations in a tailored river-like visualization. The relationships are represented by glyphs. The visual framework proposed to show such pieces of information is illustrated in Figure 19. The system also counts with a *WordCloud* (VIÉGAS; WATTENBERG, 2008) to summarize the most relevant words to the selected topic.

TextFlow employs an incremental hierarchical clustering algorithm based on the Hierarchical Dirichlet Process (HDP) (TEH *et al.*, 2006), which is used as a topic modeling technique. The algorithm clusters documents in a batch-mode manner, batching documents by its timestamps. Although the underlying algorithm is hierarchical, the visualization is a flat representation of the topics over time.

To validate the proposed system, Cui *et al.* (2011) report on two scenarios: (1) 933 papers from the proceedings of the conferences IEEE (Vis) and IEEE Information Visualization

Figure 19 – An illustration of the TextFlow visual framework. The flow areas represent topics, whereas the blue threads represents keywords within the topics. The glyphs represent: (1) topic birth, (2) topic splitting, (3) topic death, and (4) topic merging.

Source: Cui *et al.* (2011).

(InfoVis); and (2) 2,980 news articles of the *Bing News* ¹ from January to February of 2011.

Although TextFlow deals with incremental corpora, the underlying techniques have a high computational cost. The HDP algorithm is a non-parametric Bayesian technique to cluster groups of data hierarchically. Moreover, the river metaphor is based on a three-level Directed Acyclic Graph (DAG), which needs to be re-calculated after each increment. This limits the system's scalability.

The latest contribution to Vis-Kt (SHERKAT; MILIOS; MINGHIM, 2019) adds the temporal aspect to the IDC system. A traditional river metaphor depicts the temporal changes, allowing the user to explore changes of the clustering structure over time (see Figure 20).

Despite the new visual contribution, the underlying clustering algorithms do not deal with the data's temporal similarity, only the content similarity. If the clustering algorithm takes into account the temporal similarity and the content similarity, the clustering structure from one time step to the next is smoother.

3.2.2 Hierarchical visualization

Topic trees build a multi-branch tree from a corpus, where the root nodes are topics, and the leaf nodes are documents. This approach has shown to be effective in managing and visualiz-

¹ <https://www.bing.com/>

Figure 20 – The river metaphor in the Vis-Kt system. Since the system overview was described in [Figure 14](#), we blurred the already explained components and highlighted the new component's area. In the highlighted area, is possible to see a temporal visualization of the document clustering structure from April of 2016 to December of 2018.

Source: [Sherkat, Milios and Minghim \(2019\)](#).

ing large text corpora. These topic modeling techniques do not require a pre-defined number of topics, which is a drawback of the traditional topic modeling techniques (see [section 2.3.1](#)).

Since real-world textual data naturally has a hierarchical structure, which generally changes over time, the use of hierarchical models to represent and interpret it is justified ([WANG et al., 2013a](#); [WANG et al., 2013b](#)). Also, hierarchical approaches generally mitigate the scalability issue both in processing time and in visual cluttering when visualizing a large number of data points.

HierarchicalTopics ([DOU et al., 2013](#)) employs a tailored version of the Bayesian Rose Tree (BRT) ([BLUNDELL; TEH; HELLER, 2012](#)), which builds hierarchical structures through hierarchical clustering algorithms. The proposed technique called Topic Rose Tree (TRT) was designed upon three fundamental topic modeling operations: absorb, join, and collapse. TRT allows the user to split and merge topics. With this set of topic modeling operations, the user can draw her insights about the corpus.

The TRT technique is complemented by a visual analytics system (see [Figure 21](#)) that represents the hierarchical structure as a directional graph. The user can execute any of the three operations by dragging the nodes that represent each topic. With our literature review, we notice that *HierarchicalTopics* was the first system to address the dynamic analysis concern. *HierarchicalTopics* also counts with multiple river metaphor visualizations, which depict the temporal evolution of the topics at each tree level.

[Dou et al. \(2013\)](#) start with the hypothesis that when compared with flat topics, hierarchical topic structures yield faster identification of similar topics. A study with 18 users was

Figure 21 – An overview of the HierarchicalTopics system exploring a corpus of news articles extracted from CNN News. In the left, the graph view depicting the hierarchical topic structure. In the middle and right views contain river metaphors to depict the temporal evolution of topics in a hierarchical fashion.

Source: Dou *et al.* (2013).

conducted to support this hypothesis.

Although the system has the hierarchical functionality, its visual design does not scale to several hierarchical levels. The metaphor adopted to depict both the topic and temporal hierarchies can cause visual confusion and lead to cognitive overload.

To explore and track opinion diffusion on social media, Wu *et al.* (2014) introduce *OpinionFlow*, a visual analytics system that relies on an advanced diffusion model to capture the opinion diffusion among users over time. The system has five main goals, well-specified by communication and opinion mining experts: 1) summarize opinion diffusion on social media, 2) organize tweets in several levels of topic hierarchy, 3) analyze user influence in opinion diffusion, 4) flexible time granularity navigation, and 5) simulate opinion diffusion scenarios.

A visual stacked tree represents the hierarchical topics extracted with the BRT technique. One can select topics of the same level of hierarchy and analyze the flow of users in a tailored version of the Sankey diagram, where the system depicts the users' dynamics across multiple topics. Sankey diagrams are widely-accepted visual metaphors to illustrate the magnitude of information flow (RIEHMANN; HANFLER; FROEHLICH, 2005). To depict opinion diffusion, the authors adopt a density map within the Sankey diagram. Green and red visually encode positive and negative opinions, respectively (see Figure 22).

The user can also select one area and tune the opinions to predict the opinion diffusion in the following time steps. For instance, in a scenario of negative opinion about a product, the company can simulate how an influential user spreading a positive opinion about the product could revert the negative scenario.

Figure 22 – An overview of the OpinionFlow system exploring tweets about the PRISM scandal. In the yellow area, a stacked tree depicts the hierarchical topic structure with five leaf topics selected. In the red area, (E) the Sankey diagram depicts how users of the fourth topic transitioned to the third topic. The selections A, B, C, and D of the red area are detailed in the green area, depicting the user influence within the density areas. Influential users are represented as bigger nodes in the graph.

Source: Wu *et al.* (2014).

Although the system has advanced processing algorithms and a large set of interaction tools, these are also drawbacks, since its underlying algorithms have a high computational cost and the complex interactions narrow the target users.

The aforementioned TextFlow system assumes that the topics are flat structures, and thus it cannot be used to depict evolving hierarchical topic structures. However, topics have a natural hierarchical structure, which means that hierarchical representations are more suitable for this task. OpinionFlow also does not have the functionality of analyzing the hierarchical topic structure in different levels of the hierarchy.

Also based on the BRT technique, Cui *et al.* (2014) present the visual analytics system *RoseRiver* to aid the user in understanding how hierarchical topics evolve (see Figure 23). An issue in the analysis of hierarchical topics evolution is that related topics can occur in different hierarchical levels. To address this issue and to preserve the user's mental map about the corpus, *RoseRiver* employs an incremental and user-driven dynamic tree cutting algorithm.

The proposed tree cut algorithm approximates each tree, allowing the user to model the tree cuts to reflect her knowledge about the corpus. The algorithm also balances the fitness of each tree cut, smoothing the structure from one tree to the next, which also reflects in the overall structure, meaning that the following trees preserve the base structure of the former trees. This algorithm has a significant processing time, which reduces its scalability.

The system has several limitations concerning the tree cut algorithm. First, the quality of tree cut results depends on the quality of the hierarchical topic structure, meaning that if the topic modeling algorithm does not generate a well-structured tree, the tree cut algorithm might result in some meaningless trees. Second, even with well-structured trees, not all tree cuts are

Figure 23 – The visual analytics system RoseRiver exploring tweets and news articles about the PRISM scandal. (a) A tailored version of the river metaphor depicts the four main topics: Obama (orange), Prism (purple), Asylum (green), and EU Allies (blue). (b) Most relevant terms in the selected topic. The font size encodes the term frequency. (c) A tree cut executed by the user, resulting in a splitting operation in the following topics.

Source: Cui *et al.* (2014).

meaningful, depending on the hierarchical topic structure defined by the user tree cuts.

Cui *et al.* (2014) report three scenarios, each one with a different types of document: 1) 3,860 conference papers related to data mining and information visualization; 2) 69,867 news articles, and 568,225 *tweets* related to the PRISM scandal; and 3) 1,815,715 news articles from 51 news portals.

3.3 Citation-based Dynamics Visualizations

The visual analytics system *CiteRivers* proposed by Heimerl *et al.* (2016a) introduces a novel approach to scientific literature analysis, focusing on understanding the dynamics of research fields and scientific communities. Previous works only focused on either the citation pattern analysis or the content analysis, separately. *CiteRivers* extends the state-of-the-art by proposing a novel visual analytics system that combines both content and citation aspects. This approach allows the user to alternate between interactions with citation and content patterns seamlessly.

The system counts with clustering of two different aspects of scientific publications: 1) document content structure, including their dynamics over time; and 2) metadata, such as publication venue, and citations. In both aspects, *CiteRivers* employs the hierarchical spectral clustering technique (LUXBURG, 2007), which is a top-down approach that generates a binary tree that allows the adjustment of the number of clusters without modifying the tree's boundaries.

CiteRivers has five main interconnected visual components (see Figure 24). A river metaphor-like streamgraph view, depicting the topic trends over the years. The user can tune the

Figure 24 – An overview of the CiteRivers system. (a) A river metaphor segmented in timesteps. Within each segment is the most frequent words. (b) A slider to adjust the number of clusters. (c) A line chart depicting the citation aggregation over time. (d) The author panel, displaying the top 10 authors at each time-step. (e) The flow panel, depicting the hierarchical structure of venue communities. (f) A slider that controls the hierarchy levels of (e), adjusting the hierarchy granularity. (g) A document scatterplot with dimensions: citation count and trendiness score.

Source: Heimerl *et al.* (2016a).

number of clusters in the spectral clustering through a slider button. By selecting a specific topic in a year, a flow component emerges from the topic block, displaying a hierarchical structure of community venues, which can also be adjusted through a slider button. Below the streamgraph, a line graph shows two features over the years: the average citation age of each block and the citation entropy. The former is the average age of documents' references in the current block, which can help the user to understand how far back the document are sourcing their references. The latter measures the diversity of the citations of the publications along the stream based on the cited publication venues. Below the aggregation panel, the author panel displays the top 10 authors at each time step as circles. If an author occurs more the once in the view, a link connects the adjacent occurrences. And finally, a document scatterplot displays all documents along two axes: citation count and trendiness score.

The system has several functionalities. However, it also employ computationally expensive algorithms and its document representations are based on the BoW model, what limits its scalability in an incremental approach. Nevertheless, the combination of citation patterns with evolutionary topic structures presents a great contribution to the literature.

Chen (2006) presents *CiteSpace II*, a visual analytics system that builds visual represen-

tations of a document co-citation network to depict topic dynamics, and relevant documents to each topic over time.

In CiteSpace II, the authors assume that there are two citation patterns: classic and transient articles. The former are articles with persistent high citations over the year, and the latter are documents with citations peaked within a short period. The system identifies research *front terms* when a term has a significant frequency raise within a period. In the system, an *intellectual base* is defined by the group of cited articles that contain the research front terms. With this approach, CiteSpace II finds emerging terms to define temporal topics and relates the articles with these terms to find documents clusters over time.

The system encodes time slices in a color spectrum. Documents are depicted as nodes with colored rings for each time-slice color, and links between nodes represent document co-citation. Each ring's width encodes the citation count in the respective time slice. The research front terms are displayed with its time-slice color and scaled proportionally to its burst weight (KLEINBERG, 2002). The graph-based visualization is illustrated in Figure 25a. Later, Chen, Ibekwe-SanJuan and Hou (2010) introduced a timeline view (see Figure 25b) which enhances the previous graph-based view.

3.4 Discussion and Final Considerations

Table 2 lists a summary of the main aspects identified in the literature review presented in this chapter.

With this brief literature review, we can identify that the river metaphor is the most used visualization technique to depict evolutive patterns on textual data. However, the systems with more robust clustering and topic modeling techniques require a more complex visual representation. Hierarchical patterns are better illustrated with tailored versions of the river metaphor, or with Sankey diagrams.

Since document clusters' membership changes can be described as flows between clusters in different time steps, it is also noticed that the Sankey diagram is better to illustrate clusters' structure changes.

Although graph-based temporal visualizations are also reported, such techniques are neither intuitive nor scalable, since graph processing algorithms have a high computational cost.

An aspect identified as important is that the clusters' granularity must be interactive, allowing the user to explore the changes caused by the adjustment of this parameter.

Figure 25 – An overview of the temporal visualization of the CiteSpace II system.

- (a) The graph-based visualization of the CiteSpace II system. The color spectrum illustrates time slices. Documents are represented as nodes and citations are encoded in link connections between the nodes. Research front terms are colored respectively to its time slice color and scaled proportionally to its burst weight.

Source: [Chen \(2006\)](#).

- (b) A timeline view of CiteSpace II. In the figure, the research fronts are vertically displayed and along its horizontal axis is depicted the co-citation pattern over the years. The burst rings are also present in this visualization. The co-citation network graph is displayed in a matrix fashion.

Source: [Chen, Ibekwe-SanJuan and Hou \(2010\)](#).

Table 2 – A summary of the identified main aspects of visual analytics systems for dynamic document clustering exploration.

Paper	Summarization		Visualization		Corpus		Analysis	
	TM	Clustering	Hierarchical	Flat	Dynamic	Static	Dynamic	Static
Chen (2006), Chen, Ibekwe-SanJuan and Hou (2010)	✗	✓	✗	✓	✗	✓	✗	✓
Wei <i>et al.</i> (2010)	✓	✓	✗	✓	✗	✓	✗	✓
Dou <i>et al.</i> (2011)	✓	✗	✗	✓	✗	✓	✗	✓
Cui <i>et al.</i> (2011)	✓	✓	✗	✓	✗	✓	✗	✓
Dou <i>et al.</i> (2013)	✓	✓	✓	✗	✗	✓	✓	✓
Wu <i>et al.</i> (2014)	✓	✓	✓	✓	✗	✓	✓	✓
Cui <i>et al.</i> (2014)	✗	✓	✓	✓	✗	✓	✗	✓
Heimerl <i>et al.</i> (2016a)	✓	✓	✗	✓	✗	✓	✗	✓
Sherkat, Milios and Minghim (2019)	✗	✓	✓	✓	✗	✓	✓	✓

Source: Research data.

THE PROPOSED FRAMEWORK

4.1 Initial Considerations

As [Nourashrafeddin *et al.* \(2018\)](#) and [Sherkat *et al.* \(2018\)](#) report, the keyterm-based approach for document clustering presents greater clustering quality measures when compared to state-of-the-art algorithms, such as LDA and NMF. With that in mind, this project seeks to extend the formerly proposed keyterm-based IDC system, Vis-Kt, to enhance its document representation to a more efficient representation based on neural language models.

We propose a novel document clustering algorithm which inherits the Vis-Kt's original clustering pipeline with its approach of clustering terms first. Additionally, to support an incremental clustering pipeline, we propose a visual component based on the traditional Sankey diagram to depict clustering structure changes caused by user interactions or by automatic clustering of time-labeled documents.

Our goal is to preserve the well-succeed keyterm-based document clustering approach, updating Vis-Kt to more current techniques. In the following sections, we present Vis-Kt's detailed pipeline and the proposed contributions. Later in [section 4.4](#), we discuss some known limitations.

4.2 The Vis-Kt System

Vis-Kt is a web-based visual analytics system for interactive document clustering that integrates interactive visual components with keyterm-based clustering algorithms. The user can upload her documents, which will be pre-processed and encoded in a fixed size vector representation for clustering. Once this step is done, the user can visualize several pieces of information extracted from her corpus and obtain insight from it. After the corpus exploration with the visual components, the user might have some feedback that she wants to insert into the

automatic analysis, aiming to enhance the overall clustering structure.

The main idea behind the system’s framework proposed in (NOURASHRAFEDDIN *et al.*, 2018; SHERKAT *et al.*, 2018) is that, if the document clustering starts with a term clustering task, then the term clusters will generate good seeds for the later document clustering task. In an interactive clustering system, this also gives the user a simpler and more intuitive set of interaction tools, since interacting with terms is simpler than working with full-length text. This pipeline is shown in Algorithm 1.

Algorithm 1 – Vis-Kt’s general clustering framework (Adapted from Sherkat (2018))

```

1: if firstIteration then
2:   termClusters  $\leftarrow$  Generate term Clusters
3:   Get top terms of each term cluster
4: else
5:   termClusters  $\leftarrow$  User defined top terms
6: end if
7: Calculate the centroids of the term clusters
8: Find top keyterm from each clusters
9: Find document seeds based on the term clusters
10: Doc clusters  $\leftarrow$  Use seed documents to guide the clustering algorithm and then cluster
    documents

```

The limitations of the BoW model and the mean-TFIDF (see Equation 2.3) feature selection technique impairs Vis-Kt to work with a dynamic corpus without full re-processing of documents. The BoW document representation does not support dynamic datasets. Therefore, the previously proposed algorithms (LDC and iKMeans) do not work incrementally, since they are designed for the BoW model. In other words, document representation and the clustering algorithm are intrinsically static.

The user can explore and interact with visual components to obtain insights into the corpus content, adjust clustering parameters to draw her insights, and finally commit her changes once she finds a representative clustering structure. Later, the user can save her whole workspace in a structure that we call a “session”. A session includes all clusterings and visual components’ parameters. The user can also save as many sessions as she wants, and switch from one session to another.

4.3 Proposed Framework

Given the aforementioned limitations, this project modifies the document representation and proposes a novel document clustering algorithm, both based on neural language models (BENGIO *et al.*, 2003). More specifically, we choose WV (MIKOLOV *et al.*, 2013b) for word representations and PV-DM (LE; MIKOLOV, 2014) for document representations because in the context of NLM, these techniques are simpler and consistently produce satisfying results.

In [section 5.3.1](#), we compare the document representation quality between the PV and the BoW models.

The proposed system has the same pipeline and the same goal as its predecessor, which is to bind keyterm-based visual interactions with interactive document clustering algorithms. Because of the change of the document representation model, the previous algorithms are not interoperable with the proposed one and some additions to the pipeline have to account for the new document and term representations. An overview of the proposed framework is illustrated in [Figure 26](#).

Figure 26 – Pipeline for the new framework. The darker lines represents the main tasks, whereas the lighter ones stand for the sub-tasks.

Source: Elaborated by the author.

4.3.1 Document processing

In document processing, some pre-processing tasks are needed. The pre-processing algorithm produced for this project can be found in [Appendix A](#). We start by filtering, which consists of removing punctuation, numbers, and non-discriminative terms such as stop words, which are the most common in a language lexicon. After the filtering task, we need to tokenize the words of the documents. In this step it is also common to lemmatize words to reduce the vocabulary size. Both tokenization and lemmatization concepts are described in [Appendix A](#). The vocabulary is the set of all words that appear in the corpus.

Once the corpus is tokenized, we can train our WV and PV models. Generally, the models are trained with large datasets in a way that these models can generalize their predictions to most domains. However, our goal is to specify the predictions, not to generalize them. At first, we use only the corpus to train the models. To add new documents to the analysis, the user can update the models with the new documents by predicting their PV representation. However, since the predictions may be biased by the first training step, the user can also re-train the model with

the whole corpus. This makes the training time relative to the corpus size and its increments. The models are trained with the recommended hyper-parameters (MIKOLOV *et al.*, 2013b; LE; MIKOLOV, 2014).

We used *Gensim*'s implementation of WV and PV-DM algorithms, made publicly available by Rehurek and Sojka (2010)¹. To evaluate the pre-trained PV predictions and the training predictions, we have used a public available PV-DBOW model pre-trained with Wikipedia articles (LAU; BALDWIN, 2016). Since training a PV-DM with large datasets for generalization has a high computational cost, we could not either find a pre-trained model nor execute such a task.

Our experiments have shown (see Table 3) that using a model trained with the corpus itself results in better Silhouette Score (ROUSSEEUW, 1987) and less processing time. The Silhouette Score is an unsupervised metric to measure the consistency within clusters. The best value is 1 and the worst value is -1. Values near 0 indicate overlapping clusters. Negative values generally indicate that the cluster assignment is erroneous since that sample is closer to another cluster. Since our goal is to represent documents and later find representative groups, the Silhouette Score is an effective metric.

Table 3 – A comparison between the results obtained after 100 executions of predicting the PV representation with a pre-trained embedding and an embedding trained with the corpus. The Datasets D1, D2, and D3 are detailed in section 5.2. In gray background we highlight the cells with greater Silhouette Score and lower processing time.

	Metric	Datasets		
		D1	D2	D3
Pre-trained	Time (mins.)	3.2469 ± 0.0003	3.1356 ± 4.8672	4.0791 ± 0.0037
	Silhouette	0.0622 ± 0.0003	0.2758 ± 0.0010	0.0177 ± 0.0001
Trained	Time (mins.)	0.2600 ± 0.4344	0.0307 ± 0.0358	3.7102 ± 1.2048
	Silhouette	0.4593 ± 0.0073	0.5019 ± 0.0036	0.0772 ± 0.0005

Source: Research data.

4.3.2 Clustering Algorithm

With our models trained, first we normalize the WV and PV representations with the l^2 -norm (see Equation 4.1). Let x_i be the i -th vector representation. We normalize the representation matrix by the division of each vector by its Euclidean distance. Following that, if we are running the algorithm for the first time, we apply the *K-means++* (ARTHUR; VASSILVITSKII, 2007) to randomly initialize the clustering algorithm on the WV representations to find term clusters. If we are running an iteration with user feedback, first we sum the WM representations of the terms

¹ <https://radimrehurek.com/gensim/>

that define each cluster to find its 50 nearest neighbors. Once we have the 50 WV representations to each term cluster, we average them to define the seed centroids to the K-means algorithm.

$$|x_i|_2 = \frac{x_i}{\sqrt{\sum_{j=1}^n |x_{ij}|^2}} \quad (4.1)$$

The K-means algorithm will update the cluster centroids to minimize the inertia, which measures how internally coherent clusters are with the *within-cluster sum-of-squares* criterion, defined by Equation 4.2. Let X be the set of samples and C be the set of cluster centroids, where each centroid is described by the mean μ_j (PEDREGOSA *et al.*, 2012).

$$\sum_{i=0}^{|X|} \min_{\mu_j \in C} (||x_i - \mu_j||^2) \quad (4.2)$$

After the execution of the K-means algorithm in the WM representations, we find the 50 words nearest to each centroid μ_j . We construct a paragraph with the 50 words from each term cluster. These paragraphs are treated as seeds to the document clustering step. One may consider each seed paragraph as an imaginary document.

Next, we predict the PV representation of each seed paragraphs with our trained PV model. The inferred seed vectors are used to initialize the K-means algorithm to find document clusters on the PV representations.

Finally the user interacts with term clusters discovered in the WV representations. The interaction adjusts the terms that represent the term clusters (illustrated in Figure 27), a step that reflects in the document clustering seeding. The pseudo-code for the clustering process is described in Algorithm 2.

Algorithm 2 – Proposed Framework based on NLM.

```

1: if firstIteration then
2:   termSeeds  $\leftarrow$  K-means++ random seeds
3: else
4:   top50WV  $\leftarrow$  Find the 50 nearest WV to each user-defined term cluster
5:   initialCentroids  $\leftarrow$  Average top50WV to each term cluster
6:   termClusters  $\leftarrow$  Find the term clusters initialized with initialCentroids
7: end if
8: termCentroids  $\leftarrow$  Calculate the centroids of termClusters
9: for centroid in termCentroids do
10:  seedParagraphs[i]  $\leftarrow$  Get 50 nearest neighbors terms of centroid
11: end for
12: docSeeds  $\leftarrow$  Predict PV representations of seedParagraphs
13: Doc clusters  $\leftarrow$  Find document clusters with docSeeds initialization on the PV representations

```

The algorithm works as an incremental and continuous process, updating the clustering structure given each user feedback. If the user wishes, she can also re-cluster from scratch instead of updating the existing clustering structure. This is also treated as a user interaction in the system pipeline.

The K-means clustering algorithm can be replaced with any other clustering algorithm without any interference to the overall clustering pipeline. The most important part of the algorithm, which is the term clusters to find document seeds, can be achieved with any general-purpose clustering algorithm, such as FCM, DBSCAN (ESTER *et al.*, 1996), and BIRCH (ZHANG; RAMAKRISHNAN; LIVNY, 1996).

Figure 27 – Illustration of user interactions with term clusters, adapted from Figure 14E. In the figure, the Cluster2 is selected. It can have terms added or removed, and it can be deleted or renamed.

Source: Elaborated by the author.

In section 5.3.2, we compare the proposed clustering algorithm with the algorithms used in Vis-Kt's previous version. The algorithm has been implemented using the programming language *Python*².

4.3.3 Interaction History Visualization

Developed with *D3.js*³ (BOSTOCK; OGIEVETSKY; HEER, 2011), a JavaScript library generally used for creating dynamic visualizations, the proposed system inherits the visual components of the previous version of Vis-Kt (see Figure 14) with the addition of a new visual component called Document Clustering Versions, which represents the clusterings after each user interaction as flows on a Sankey diagram. The goal is to identify changes in clusters and movement of documents between clusters after each session is saved.

The system represents the Sankey diagram as a DAG. Within the DAG structure, M layers represent the M saved sessions. Each session's cluster is represented as a node in the graph. The links between two nodes are defined by the document cluster membership changing from one session to the next.

² <http://www.python.org>

³ <https://d3js.org/>

The system calculates the diagram's layout by representing the nodes as rectangles displayed vertically within M columns. To place the rectangles in the 2D plane, its areas are calculated with Equation 4.3 and Equation 4.4, where $columnWidth$ is a fixed rectangle width for all nodes, and $hGap$ and $vGap$ are horizontal and vertical spacings between nodes, respectively. $nodeHeight(i, j)$ stands for the height of the j -th node of the i -th column.

$$X_{min} = (i - 1) * (columnWidth + hGap), X_{max} = X_{min} + columnWidth \quad (4.3)$$

$$Y_{min} = \sum_{k=0}^{j-1} (nodeHeight(i, k) + vGap), Y_{max} = Y_{min} + nodeHeight(i, j) \quad (4.4)$$

Next, the flows are calculated with a cubic Bézier curve function, starting from the source node to the target node. A cubic Bézier curve is a parametric function that applies linear interpolations to draw a curve, given two reference points and two relative control points. This process is illustrated in Figure 28.

Figure 28 – Cubic Bézier curves applied to display the Sankey diagram flows. The dashed line represents the calculated curve, whereas the continuous line represents the minimal distance from the source to the target nodes.

Source: Elaborated by the author.

With the Sankey diagram, the user can track how her feedback affected the clustering result. She can highlight any cluster at any time, explore the documents that compose the selected cluster and analyze its clustering membership in previous iterations. The top five keyterms of each cluster are shown by hovering over the cluster representation.

This visual component takes advantage of the incremental aspect of the keyterm-based clustering algorithm. The user can track where each term cluster went after each interaction.

The split and merge operations are intuitively represented as flows on the Sankey diagram. In the split operation, a cluster represented by a rectangle has two or more transitions to other clusters in the next iteration. In the merge operation, a cluster inputs two or more transitions from the previous iteration. The visual component and the operations are depicted in Figure 29.

Figure 29 – The overview of the proposed visual component. In the left (A), the Sankey diagram depicting five clustering steps (V1, V2, V3, V4, and V5) on a dataset of 381 news articles. The clustering at each step is represented by rectangles vertically oriented. Document transition from one cluster at time step t to time step $t + 1$ is represented as horizontal stripes. The selected transition is highlighted, and the cluster hovered over shows its keyterms within a white box. In the right (B), the list of documents in the current selection. The colored dots represents the document's cluster membership in each iteration. The red circle (1) stands for the beginning of a split operation of Cluster9 to Cluster1 and Cluster6. The green circle (2) stands for the beginning of a merge operation of Cluster1 and Cluster6 into Cluster4.

Source: Elaborated by the author.

4.4 Discussions and limitations

The proposed enhancements to Vis-Kt's document representation model are a proof of concept that NLM can be applied to keyterm-based document clustering algorithms. This is a novel approach, since, to the best of our knowledge, no IDC system has employed NLM as its document representation model.

However, the models employed in this project can be considered out-dated, since, there are novel approaches that overcame certain limitations. The WV and PV-DM models are pioneer contributions to the field of NLM, but models such as the Bidirectional Encoder Representations from Transformers (BERT) (DEVLIN *et al.*, 2018) present higher scores in prediction tasks. The adoption of such novel techniques requires further evaluations.

Another limitation of this project is that Vis-Kt was designed for the BoW model. Its visual components and its software engineering design rely on BoW model data structures. To overcome these limitations, some of Vis-Kt's functionalities were adjusted to work properly with the new model. For instance, the document similarity represented as links in the graph view (see Figure 14D) had its threshold decreased because the document similarity in the NLM dense vectors is significantly higher in comparison with the sparse vectors in the BoW model.

The next limitation concerns the system's visual design. The NLM models have several advantages over the BoW model, such as the ability to depict the relationship between words,

positive and negative feedback to describe the terms that define a term cluster. However, the system in its current version does not have the visual components to depict such features, which can be useful for the user to obtain insights into the corpus' content.

Given these limitations, the system does not fully support our proposed algorithm. The original system was used to validate our proof of concept of applying keyterm-based clustering algorithms integrated to NLM models. However, there are still major changes to perform on the Vis-Kt system for it to take advantage of all features provided by the NLM model.

In addition, the Sankey diagram is not located at the main view ([Figure 14](#)). It is placed in an external page. In Vis-Kt future versions, it might be useful to redesign the visual components, excluding unnecessary components, and adding the Sankey diagram in the main view for applications where evolution of clustering are central.

EXPERIMENTS AND DISCUSSIONS

5.1 Initial Considerations

In this chapter, the proposed clustering algorithm is compared with iKMeans and LDC, the original techniques used in the Vis-Kt system. LDC has been demonstrated to produce better results than LDA and NMF ([NOURASHRAFEDDIN *et al.*, 2018](#)). The iKMeans algorithm produces better results and requires less processing time than LDC in some cases ([SHERKAT *et al.*, 2018](#)).

In regards to the new algorithm, a quantitative evaluation is reported to compare the processing time, the representation effectiveness to embed the document in a vector representation and the clustering quality with several clustering quality measures.

All experiments were run on a dedicated machine with the following specifications:

- OS: Ubuntu 18.04 Bionic
- Kernel: x86_64 Linux 5.0.0-32-generic
- CPU: Intel Core i7-6850K, 12 cores (4GHz)
- Memory: 4x 16GB DDR4 (2400MHz)

In order to evaluate the proposed visual component, we also engaged one visual analytics expert to provide quantitative feedback on the utility of the Sankey Diagram for depicting the interactive clustering steps. The expert was a Masters student with experience in visual analytics on textual data. The expert used a news article dataset of his preference.

5.2 The Data

We applied the algorithm on three publicly available datasets. In the following sections, we are going to refer to these datasets as D1, D2 and D3, respectively as described below:

1. **CBR-ILP-IR-SON (D1)**, 675 scientific papers of four computer science subjects namely: Case-Based Reasoning (CBR), Inductive Logic Programming (ILP), Information Retrieval (IR) e Sonification (SON). Its instances contain the title, authorship, affiliation, abstract, and references collected in 2006 with a search using corresponding queries to each subject. This dataset also has a variation with a extra Intruder (INT) category, which we used in the usage scenario described in [section 5.4.1](#). This dataset was produced and proposed in [Paulovich *et al.* \(2008\)](#).
2. **NewsSeparate (D2)**, 381 news RSS feed digests manually labeled with 13 labels. This dataset was also proposed in [Paulovich *et al.* \(2008\)](#) and collected in 2006, from 4 different news outlets web pages (CNN, BBC, Reuters and Associated Press).
3. **20NewsGroup Training Set (D3)**, 11314 news group articles partitioned evenly across 20 different labels. This dataset was proposed in [Lang \(1995\)](#). We fetched the dataset with the *Scikit-learn* library ([PEDREGOSA *et al.*, 2012](#)).

5.3 Quantitative Evaluation

To quantitatively evaluate the proposed clustering algorithm and compare it with the state-of-the-art in keyterm-based clustering algorithms, LDC and iKMeans, we have conducted an experiment with the three datasets presented in [section 5.2](#). We follow the previous evaluation process performed by [Nourashrafeddin *et al.* \(2018\)](#) and [Sherkat *et al.* \(2018\)](#), consisting of collecting clustering quality measures of 100 executions and presenting their average and standard deviation.

Besides the Average Silhouette Score previously introduced, we also employed the Adjusted Rand Score (ARS), Adjusted Mutual Information (AMI), Fowlkes-Mallows Index (FMI), and V-Measure to evaluate the clustering algorithm in multiple aspects. The clustering experiments were fully automated and since only the first step with random seeds is being evaluated, no user interactions were simulated.

ARS is a similarity measure between the ground truth classes and the predicted clustering ([HUBERT; ARABIE, 1985](#)). This measure ensures that a value near 0 stands for random labeling and values near 1 stands for identical clusterings. ARS is penalized by the number of false-positive and false-negative predictions.

Mutual Information (MI) measures the agreement between the ground truth and the clustering prediction assignments, ignoring permutations. AMI is an adjustment of the MI score

to reduce the effect of the agreement by chance (VINH; EPPS; BAILEY, 2010). Values near 0 stand for random clustering.

The FMI measures the similarity between two clusterings and it is defined as the geometric mean of the pairwise precision and recall (FOWLKES; MALLOWS, 1983). Precision measures the proportion of correct cluster membership assignment, whereas recall measures the proportion of all cluster members that were correctly assigned.

A clustering algorithm has perfect *Homogeneity* (H) if each cluster contains only data points of a single class; a clustering has perfect *Completeness* (C) if all data points of a given class are assigned to the same cluster. Rosenberg and Hirschberg (2007) define the *V-Measure* as the harmonic mean of H and C. The scores are in the range of 0 to 1, where 1 stands for perfect clustering.

Table 4 lists the aspects identified as most relevant to measure in a clustering algorithm.

Table 4 – Clustering aspects of each metric in the project. Random clustering: if the metric is random clustering aware. Cluster shape: if the metric can compare clusterings with different shapes. Precision + Recall: if the metric measures precision and recall.

Metric	Clustering Aspect		
	Random Clustering	Cluster Shape	Precision + Recall
Silhouette	✓	✗	✗
ARS	✓	✓	✗
AMI	✓	✗	✗
FMI	✓	✓	✓
V-Measure	✗	✓	✗

Source: Research data.

5.3.1 Document Representation

The first evaluation concerned the document representation processing time and embedding quality, measured by the Silhouette score. As can be seen in Table 5, the PV model outperforms the BoW model with mean-TFIDF in processing time in most cases, except for the D2 case. And the PV model Silhouette score is significantly greater in all three cases.

In Figure 30, we report the results of 100 executions of the t-SNE dimensionality reduction technique on the best embeddings from the experiment reported in Table 5. We have chosen the Silhouette score metric because it can measure how well-segregated a group is, which is what we are aiming at with the dimensionality reduction. The l^2 -norm was applied in both representation models. We used a public available¹ incremental implementation of the t-SNE technique that has shown faster and greater results than the most popular implementations (POLIČAR;

¹ <https://github.com/pavlin-polcar/openTSNE>

Table 5 – Document processing time (in minutes) and representation silhouette score. Average and standard deviation of 100 executions. The gray tinting highlights the cells with greater Silhouette Score and lower processing time.

Model	Metric	Datasets		
		D1	D2	D3
Paragraph Vector	Time (mins.)	0.26 ± 0.434	0.03 ± 0.035	3.71 ± 1.2048
	Silhouette	0.45 ± 0.007	0.50 ± 0.003	0.07 ± 0.0005
BoW (mean-TFIDF)	Time (mins.)	0.45 ± 0.049	0.02 ± 0.008	3.82 ± 0.558
	Silhouette	0.11 ± 0.0	0.27 ± 0.0	0.01 ± 0.0

Source: Research data.

STRAŽAR; ZUPAN, 2019), such as the one available at *Scikit-learn*² (PEDREGOSA *et al.*, 2012).

In Figure 30a, we can notice that the PV model shows better group segregation, which can be confirmed with its silhouette score. Several groups are overlapping in the PV model in Figure 30b, which penalizes its silhouette score. This occurs because the dataset D2 is the smallest one and it consists of small RSS feed digests, which means that the groups are not far apart in the NLM representation. Although the original space has a higher Silhouette score, when the t-SNE reduction is executed, the small distance between the groups causes the t-SNE algorithm to not be able to distinguish closer groups well (WATTENBERG; VIÉGAS; JOHNSON, 2016). In this case of projection, the l^2 normalized BoW's sparse vector is more effective.

Finally, in Figure 30c, several groups are overlapping in both models, explaining the negative Silhouette score. However, we can also observe that the PV model shows more compact groups.

5.3.2 Document Clustering

The speed of the proposed algorithm compared with iKMeans and LDC is shown in Table 6. The proposed algorithm is faster in all experimented scenarios. Focusing on the dataset D3, we see that our algorithm has a significant processing time difference. D3 is the larger dataset (11,000 documents) with a large vocabulary, which results in longer processing time on BoW based models.

The Silhouette scores of the clusters obtained with the proposed algorithm are greater than the baseline algorithms, as shown in Figure 31a. This outcome is a consequence of the dense vector that PV representations have. The PV dense vector generally has fewer features and all those features carry information about the document. On the other hand, BoW vectors are sparse, i.e. most features have zero value that do not carry any information. Also, the BoW based

² <https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html>

Figure 30 – The average of the silhouette score in the t-SNE projection technique executed 100 times in the datasets. This metric shows values closer to 1 on projections with good segregation, values closer to 0 when the groups overlap each other, and -1 to random clustering.

(a) CBR-ILP-IR-SON dataset with 675 documents projected with t-SNE.

(b) NewsSeparate dataset with 381 documents projected with t-SNE.

(c) 20NewsGroup dataset with 11,314 documents projected with t-SNE.

Source: Elaborated by the author.

Table 6 – Document clustering time (in minutes) of the proposed algorithm, iKMeans, and LDC. Average and standard deviation of 100 executions. The gray tinting highlights the cells lower processing time.

Technique	Datasets		
	D1	D2	D3
Proposed Algorithm	0.3210 $\pm$ 0.1447	0.2137 $\pm$ 0.0944	0.0470 $\pm$ 0.0050
iKMeans	0.8596 $\pm$ 0.0999	0.3452 $\pm$ 0.0190	1.6441 $\pm$ 0.0326
LDC	0.5337 $\pm$ 0.1310	0.7975 $\pm$ 0.7189	4.1326 $\pm$ 3.1289

Source: Research data.

models tend to have more features as the dataset size grows, making the vector representation even more sparse.

Figure 31 – Document clustering algorithms average metrics with standard deviation after 100 executions.

Source: Research data.

The ARS (Figure 31b), AMI (Figure 31c), and V-Measure (Figure 31d) scores demonstrate that the proposed algorithm outperforms iKMeans and LDC in the third case. In the first and second cases, the proposed algorithm shows a quality loss over the baseline algorithms. We

can notice in [Figure 31e](#) that similar to the clustering metrics, the FMI score shows a quality loss over the baseline algorithms in the first and second cases. In the third case, the new algorithm shows a significant lead. Moreover, the proposed algorithm shows stable results in all three cases, as we can see in the error bars that depict the standard deviation.

The quality loss is directly related to the sizes of the datasets. D1 and D2 are the smallest datasets, which impairs the learning process of NML algorithms. However, the algorithm still has results as good as or greater than the baseline algorithms, and it is also faster in most cases, as depicted in [Table 6](#).

The reported results demonstrate the effectiveness of the proposed algorithm, which in most cases outperforms state-of-the-art keyterm-based clustering algorithms in clustering quality and processing time. The results also show how the algorithm does not decrease its quality as the dataset size increases, which makes the algorithm very scalable. Moreover, the model increases its quality as the dataset grows, given that the model will have more information to learn from.

5.4 Usage Scenarios

In the following sections, we report two usage scenarios that were explored through this project in regards to the visualization of time varying document clusters. The first scenario is related to how the Sankey diagram employed in this project can be used to depict clustering flows across several clustering iterations and how the visual component addresses the scenario of data streams. Following that, we report an expert study executed to qualitatively evaluate the proposed clustering algorithm and the Sankey diagram for clustering evolution analysis.

5.4.1 Incremental Clustering Analysis

The new Sankey Diagram visual component has the purpose of depicting clustering iterations with user feedback to the clustering algorithm. Moreover, the visual component also can be used to depict clustering of text data streams.

In both scenarios, the visualization is useful for the user to recognize change in clustering patterns. If a cluster is constant between several executions or iterations, this means that this is a strong and well-defined cluster. In our context it means that the keyterms that compose this cluster are good seeds.

This clustering behavior is depicted in [Figure 32](#). In this usage scenario, the dataset D1 is partitioned into six equally proportional partitions. The clusterings at V1, V2, V4, and V6, are examples of incremental clustering, whereas the clusterings at V3 and V5 are clusterings with user feedback. We can notice that both scenarios are seamlessly represented in the Sankey diagram, and the clustering structure shows consistency across five iterations with different types of interference (data streams and user interaction). This is confirmed with the WordClouds of the

selected clusters.

To evaluate the proposed visual component, we engaged one visual analytics expert to provide quantitative feedback on the utility of the Sankey Diagram to depict the interactive clustering steps. The expert is a Masters student with experience in visual analytics on textual data. The expert used a news article dataset of his preference.

One unexpected and surprising feedback was that the expert found the clustering visualization of the Sankey Diagram more intuitive than the whole set of visual components. He reported that after six clustering iterations, the resulting flow visualization on the Sankey Diagram brought him more insight into the content of the corpus than the previous interaction with the main visual components. He suggested that the Sankey Diagram should be used not

Figure 33 – The expert’s feedback on the visual component utility. The colors represent a qualitative measure, where *Green* stands for “Very useful”, *Orange* stands for “Useful” and *Red* stands for “Rarely used”.

Source: Elaborated by the author.

only to depict the results of the interactions with the main components but it should also be considered as a clustering interaction tool. For instance, the user may visually choose which clusters she wants to merge or split by dragging and dropping the cluster representations in the Sankey Diagram. Also, removal or addition of terms to each cluster, similar to what is offered on the other visual components, could be done via the streaming graph.

He reported that the merge and split operations were well-represented in the visual component and that the visualization of these operations in the Sankey Diagram brought very useful insight on which keyterms he should use as seeds in subsequent iterations. The expert’s feedback on the visual component utility is depicted in [Figure 34](#).

5.5 Final Considerations

With our quantitative evaluation, we can observe that the proposed algorithm shows significantly greater results on larger datasets. This is the expected behavior since it relies on a machine learning algorithm that depends on a good dataset to effectively learn its content. A valuable experiment to execute in the future is to compare the variety of available NLMs and measure if employing the most recent models in our algorithm outperform the PV-DM proposed

Figure 34 – One insight reported by the expert. In the figure, a split operation (Step 5) and a merge operation (Step 7) are identified. The keyterm of the clusters marked as 1, 2 and 3 were reported to be related, which explains the sequential split and merge operations.

Source: Elaborated by the author.

in [Le and Mikolov \(2014\)](#).

Regarding the visual tools, our expert study emphasizes the need to execute a user study to evaluate Vis-Kt's usability, considering a full redesign of the system. Moreover, this first feedback brings great perspectives to embed the Sankey diagram as a major visual component to support the keyterm-based clustering algorithm, because of its intuitiveness and flexibility, still being scalable to growing datasets and incremental clustering scenarios.

In conclusion, although the proposed algorithm has shown satisfying results when compared with the baseline algorithms, it still needs a tailored visual analytics system to support its flexible set of features that are not addressed by the Vis-Kt's original design. The Sankey diagram was evaluated in limited scenarios, and it needs further validation.

CONCLUSION AND FUTURE RESEARCH

In order to overcome the limitations caused by the BoW document representation employed in keyterm based clustering and in the Vis-Kt system, this project updates the document representation of the approach to NLM, which tackles these limitations. The novel document representation shows higher clustering quality measures and faster processing time in all of the explored cases. Since Vis-Kt's baseline algorithms are designed for the BoW model, we also propose a novel interactive clustering algorithm tailored for NLM. We reported that the proposed clustering algorithm has results as good as the baseline algorithms in a shorter time. In the case of larger datasets, the proposed algorithm shows a significant lead on clustering measure and processing time when compared with the baseline algorithms. This indicates that the algorithm has larger potential for scalability. Resource-wise, the proposed algorithm does not require more computational resources than the baseline algorithms.

The proposed algorithm advantages are impaired by Vis-Kt's original design, which does not have the capability of exploring NLM features. Although we have adapted the system's original design to work with the proposed model, the system was designed for BoW based data structures. Since it can not be done without disassembling the whole system, this hampers deep changes to its core. We intend to engage a formal user study to evaluate and redesign Vis-Kt's visual components, tailoring it for NLM, to explore the whole set of features that the proposed model provides. A valuable future work for this research is to execute an exploration of more recent NLM, such as GloVe (PENNINGTON; SOCHER; MANNING, 2014), FastText (BOJANOWSKI *et al.*, 2017), and BERT (DEVLIN *et al.*, 2018), which have shown improvement of document representation in comparison with the PV-DM proposed by Le and Mikolov (2014) and employed here.

Additionally, to support the iterative process of keyterm-based document clustering, we have proposed a visual component based on the traditional Sankey diagram to depict cluster flows across several document clustering iterations. The iterations can be resulting from a growing dataset or caused by consecutive user interactions with the document clustering algorithm. The

proposed visual component was qualitatively evaluated with two usage scenarios, where the first was an experiment with a growing dataset. In the second scenario, we engaged one researcher with expertise in visual analytics applied to documents. The feedback obtained from these usage scenarios indicates that the proposed visual component plays a key role in the user-driven keyterm-based document clustering system. However, it still needs a formal user study to measure and validate its usability. We intend to extend the proposed Sankey diagram to not only depict cluster flows but to also integrate it in the document clustering task, by providing interactions for the user to design clusters from the Sankey diagram's interface.

The results reported in this document are also reported in a paper accepted for publication in the proceedings of the International Conference on Advanced Visual Interfaces (AVI 2020).

BIBLIOGRAPHY

ALENCAR, A. B.; OLIVEIRA, M. C. F. de; PAULOVICH, F. V. Seeing beyond reading: a survey on visual text analytics. **Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery**, v. 2, n. 6, p. 476–492, 11 2012. Available: <http://doi.wiley.com/10.1002/widm.1071>>. Citation on page 25.

ANDREWS, C.; ENDERT, A.; YOST, B.; NORTH, C. Information visualization on large, high-resolution displays: Issues, challenges, and opportunities. **Information Visualization**, v. 10, n. 4, p. 341–355, 10 2011. ISSN 1473-8716. Available: <http://journals.sagepub.com/doi/10.1177/1473871611415997>>. Citation on page 47.

ARTHUR, D.; VASSILVITSKII, S. k-means++: The advantages of careful seeding. In: **Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics**. [S.l.: s.n.], 2007. Citation on page 66.

AVONS, S. E.; WRIGHT, K. L.; PAMMER, K. The Word-Length Effect in Probed and Serial Recall. **The Quarterly Journal of Experimental Psychology Section A**, v. 47, n. 1, p. 207–231, 2 1994. ISSN 0272-4987. Available: <http://journals.sagepub.com/doi/10.1080/14640749408401151>>. Citation on page 25.

AWASTHI, P.; BALCAN, M. F.; VOEVODSKI, K. Local algorithms for interactive clustering. **Journal of Machine Learning Research**, v. 32, n. 2, p. 550–558, 2014. Available: <http://proceedings.mlr.press/v32/awasthi14.html>>. Citation on page 37.

BALCAN, M.-F.; BLUM, A. Clustering with Interactive Feedback. In: **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**. [s.n.], 2008. v. 5254 LNAI, p. 316–328. ISBN 9783540879879. Available: http://link.springer.com/10.1007/978-3-540-87987-9_27>. Citations on pages 37 and 39.

BENGIO, Y.; DUCHARME, R.; VINCENT, P.; JAUVIN, C. A Neural Probabilistic Language Model. **Journal of Machine Learning Research**, v. 3, p. 1137–1155, 2003. Available: <http://www.jmlr.org/papers/volume3/bengio03a/bengio03a.pdf>>. Citations on pages 26 and 64.

BEZDEK, J. C. **Pattern Recognition with Fuzzy Objective Function Algorithms**. Boston, MA: Springer US, 1981. 267 p. ISBN 978-1-4757-0452-5. Available: <http://link.springer.com/10.1007/978-1-4757-0450-1>>. Citation on page 43.

BIRD, S.; LOPER, E.; KLEIN, E. **Natural Language Processing with Python**. First. [S.l.]: O'Reilly Media Inc., 2009. ISBN 978-0-596-51649-9. Citation on page 95.

BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent Dirichlet Allocation. **Journal of Machine Learning Research**, v. 3, p. 993–1022, 2003. Available: <https://dl.acm.org/doi/10.5555/944919.944937>>. Citations on pages 37 and 38.

BLUNDELL, C.; TEH, Y. W.; HELLER, K. A. Bayesian Rose Trees. **XXVI Conference on Uncertainty in Artificial Intelligence**, p. 65–72, 3 2012. Available: <http://arxiv.org/abs/1203.3468>. Citation on page 54.

BOJANOWSKI, P.; GRAVE, E.; JOULIN, A.; MIKOLOV, T. Enriching Word Vectors with Subword Information. **Transactions of the Association for Computational Linguistics**, v. 5, p. 135–146, 12 2017. ISSN 2307-387X. Available: https://www.mitpressjournals.org/doi/abs/10.1162/tacl_a_00051. Citation on page 83.

BOSTOCK, M.; OGIEVETSKY, V.; HEER, J. D Data-Driven Documents. **IEEE Transactions on Visualization and Computer Graphics**, v. 17, n. 12, p. 2301–2309, 12 2011. ISSN 1077-2626. Available: <http://ieeexplore.ieee.org/document/6064996/>. Citation on page 68.

CHAMBERS, J. M.; CLEVELAND, W. S.; KLEINER, B.; TUKEY, P. A. **Graphical Methods for Data Analysis**. [S.l.]: Wadsworth, 1983. Citation on page 51.

CHEN, C. CiteSpace II: Detecting and visualizing emerging trends and transient patterns in scientific literature. **Journal of the American Society for Information Science and Technology**, v. 57, n. 3, p. 359–377, 2006. ISSN 15322882. Citations on pages 58, 60, and 61.

CHEN, C.; IBEKWE-SANJUAN, F.; HOU, J. The structure and dynamics of cocitation clusters: A multiple-perspective cocitation analysis. **Journal of the American Society for Information Science and Technology**, v. 61, n. 7, p. 1386–1409, 3 2010. ISSN 15322882. Available: <https://onlinelibrary.wiley.com/doi/full/10.1002/asi.21309><http://doi.wiley.com/10.1002/asi.21309>. Citations on pages 59, 60, and 61.

CHOO, J.; LEE, C.; REDDY, C. K.; PARK, H. UTOPIAN: User-Driven Topic Modeling Based on Interactive Nonnegative Matrix Factorization. **IEEE Transactions on Visualization and Computer Graphics**, v. 19, n. 12, p. 1992–2001, 12 2013. ISSN 1077-2626. Available: <http://ieeexplore.ieee.org/document/6634167/>. Citations on pages 40, 41, and 48.

CHOO, J.; LEE, H.; LIU, Z.; STASKO, J.; PARK, H. An interactive visual testbed system for dimension reduction and clustering of large-scale high-dimensional data. In: WONG, P. C.; KAO, D. L.; HAO, M. C.; CHEN, C.; HEALEY, C. G. (Ed.). **Proc. SPIE 8654, Visualization and Data Analysis 2013**. [s.n.], 2013. Available: <http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.2007316>. Citations on pages 40, 42, and 48.

CUI, W.; LIU, S.; TAN, L.; SHI, C.; SONG, Y.; GAO, Z.; QU, H.; TONG, X. TextFlow: Towards Better Understanding of Evolving Topics in Text. **IEEE Transactions on Visualization and Computer Graphics**, ACM Press, New York, New York, USA, v. 17, n. 12, p. 2412–2421, 12 2011. ISSN 1077-2626. Available: <http://dl.acm.org/citation.cfm?doid=1835804.1835827><http://ieeexplore.ieee.org/document/6065008/>. Citations on pages 52, 53, and 61.

CUI, W.; LIU, S.; WU, Z.; WEI, H. How Hierarchical Topics Evolve in Large Text Corpora. **IEEE Transactions on Visualization and Computer Graphics**, IEEE, v. 20, n. 12, p. 2281–2290, 12 2014. ISSN 1077-2626. Available: <http://ieeexplore.ieee.org/document/6875938/>. Citations on pages 56, 57, and 61.

DESJARDINS, M.; MACGLASHAN, J.; FERRAIOLI, J. Interactive visual clustering. **Proceedings of the 12th international conference on Intelligent user interfaces - IUI '07**, p. 361, 2007. ISSN 0016-5085. Available: <http://dl.acm.org/citation.cfm?id=1216295.1216367>. Citation on page 46.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. n. Mlm, 10 2018. Available: <http://arxiv.org/abs/1810.04805>>. Citations on pages 70 and 83.

DIAS, A. G.; MILIOS, E. E.; OLIVEIRA, M. C. F. de. TRIVIR. In: **Proceedings of the ACM Symposium on Document Engineering 2019 - DocEng '19**. New York, New York, USA: ACM Press, 2019. p. 1–10. ISBN 9781450368872. Available: <http://dl.acm.org/citation.cfm?doid=3342558.3345401>>. Citation on page 35.

DOU, W.; WANG, X.; CHANG, R.; RIBARSKY, W. ParallelTopics: A probabilistic approach to exploring document collections. In: **2011 IEEE Conference on Visual Analytics Science and Technology (VAST)**. IEEE, 2011. p. 231–240. ISBN 978-1-4673-0014-8. Available: <http://ieeexplore.ieee.org/document/6102461/>>. Citations on pages 39, 43, 51, 52, and 61.

DOU, W.; YU, L.; WANG, X.; MA, Z.; RIBARSKY, W. HierarchicalTopics: Visually Exploring Large Text Collections Using Topic Hierarchies. **IEEE Transactions on Visualization and Computer Graphics**, v. 19, n. 12, p. 2002–2011, 12 2013. ISSN 1077-2626. Available: <http://ieeexplore.ieee.org/document/6634160/>>. Citations on pages 54, 55, and 61.

ENDERT, A.; FIAUX, P.; NORTH, C. Semantic Interaction for Sensemaking: Inferring Analytical Reasoning for Model Steering. **IEEE Transactions on Visualization and Computer Graphics**, IEEE, v. 18, n. 12, p. 2879–2888, 12 2012. ISSN 1077-2626. Available: <http://ieeexplore.ieee.org/document/6327294/>>. Citations on pages 47 and 48.

_____. Semantic interaction for visual text analytics. **Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12**, p. 473, 2012. ISSN 1473-6551. Available: <http://dl.acm.org/citation.cfm?doid=2207676.2207741>>. Citations on pages 46 and 47.

ENDERT, A.; FOX, S.; MAITI, D.; LEMAN, S.; NORTH, C. The semantics of clustering. In: **Proceedings of the International Working Conference on Advanced Visual Interfaces - AVI '12**. New York, New York, USA: ACM Press, 2012. p. 555. ISBN 9781450312875. Available: <http://dl.acm.org/citation.cfm?doid=2254556.2254660>>. Citation on page 47.

ESTER, M.; KRIEGEL, H.-P.; SANDER, J.; XU, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: **2nd International Conference on Knowledge Discovery and Data Mining**. AAAI Press, 1996. v. 2, p. 226–231. Available: <https://www.aaai.org/Papers/KDD/1996/KDD96-037.pdf>>. Citation on page 68.

FEDERICO, P.; HEIMERL, F.; KOCH, S.; MIKSCH, S. A Survey on Visual Approaches for Analyzing Scientific Literature and Patents. **IEEE Transactions on Visualization and Computer Graphics**, v. 23, n. 9, p. 2179–2198, 2017. Citation on page 25.

FOWLKES, E. B.; MALLOWS, C. L. A Method for Comparing Two Hierarchical Clusterings. **Journal of the American Statistical Association**, v. 78, n. 383, p. 553–569, 9 1983. ISSN 0162-1459. Available: <http://www.tandfonline.com/doi/abs/10.1080/01621459.1983.10478008>>. Citation on page 75.

FRUCHTERMAN, T. M. J.; REINGOLD, E. M. Graph drawing by force-directed placement. **Software: Practice and Experience**, v. 21, n. 11, p. 1129–1164, 11 1991. ISSN 00380644. Available: <http://doi.wiley.com/10.1002/spe.4380211102>>. Citation on page 34.

HARRIS, Z. S. Distributional Structure. **WORD**, v. 10, n. 2-3, p. 146–162, 8 1954. ISSN 0043-7956. Available: <<http://www.tandfonline.com/doi/full/10.1080/00437956.1954.11659520>>. Citation on page 30.

HAVRE, S.; HETZLER, E.; WHITNEY, P.; NOWELL, L. ThemeRiver: visualizing thematic changes in large document collections. **IEEE Transactions on Visualization and Computer Graphics**, IEEE, v. 8, n. 1, p. 9–20, 2002. ISSN 10772626. Available: <<http://ieeexplore.ieee.org/document/981848/>>. Citation on page 50.

HEIMERL, F.; HAN, Q.; KOCH, S.; ERTL, T. CiteRivers: Visual Analytics of Citation Patterns. **IEEE Transactions on Visualization and Computer Graphics**, v. 22, n. 1, p. 190–199, 2016. ISSN 10772626. Citations on pages 57, 58, and 61.

HEIMERL, F.; JOHN, M.; Qi Han; KOCH, S.; ERTL, T. DocuCompass: Effective exploration of document landscapes. In: **2016 IEEE Conference on Visual Analytics Science and Technology (VAST)**. IEEE, 2016. p. 11–20. ISBN 978-1-5090-5661-3. ISSN 2325-9442. Available: <<http://ieeexplore.ieee.org/document/7883507/http://ip-science.thomsonreuters.com/m/pdfs/aureka>>. Citation on page 35.

HU, Y.; BOYD-GRABER, J.; SATINOFF, B.; SMITH, A. Interactive topic modeling. **Machine Learning**, v. 95, n. 3, p. 423–469, 6 2014. ISSN 0885-6125. Available: <<https://dl.acm.org/citation.cfm?id=2002505http://link.springer.com/10.1007/s10994-013-5413-0>>. Citations on pages 25 and 37.

HU, Y.; MILIOS, E. E.; BLUSTEIN, J. Interactive feature selection for document clustering. In: **Proceedings of the 2011 ACM Symposium on Applied Computing - SAC '11**. New York, New York, USA: ACM Press, 2011. p. 1143. ISBN 9781450301138. Available: <<http://portal.acm.org/citation.cfm?doid=1982185.1982436>>. Citation on page 42.

_____. Enhancing semi-supervised document clustering with feature supervision. In: **Proceedings of the 27th Annual ACM Symposium on Applied Computing - SAC '12**. New York, New York, USA: ACM Press, 2012. p. 929. ISBN 9781450308571. Available: <<http://doi.acm.org/10.1145/2245276.2245457http://dl.acm.org/citation.cfm?doid=2245276.2245457>>. Citation on page 42.

HUBERT, L.; ARABIE, P. Comparing partitions. **Journal of Classification**, v. 2, n. 1, p. 193–218, 1985. ISSN 01764268. Citation on page 74.

INSELBERG, A. The plane with parallel coordinates. **The Visual Computer**, v. 1, n. 2, p. 69–91, 8 1985. ISSN 0178-2789. Available: <<http://link.springer.com/10.1007/BF01898350>>. Citations on pages 39, 43, and 51.

JEONG, D. H.; ZIEMKIEWICZ, C.; FISHER, B.; RIBARSKY, W.; CHANG, R. iPCA: An Interactive System for PCA-based Visual Analytics. **Computer Graphics Forum**, v. 28, n. 3, p. 767–774, 6 2009. ISSN 01677055. Available: <<http://doi.wiley.com/10.1111/j.1467-8659.2009.01475.x>>. Citation on page 34.

JOLLIFFE, I. T. **Principal Component Analysis**. 2nd. ed. New York: Springer-Verlag, 2002. 513 p. (Springer Series in Statistics). ISSN 0172-7397. ISBN 0-387-95442-2. Available: <<http://link.springer.com/10.1007/b98835>>. Citation on page 34.

JONES, K. S. A STATISTICAL INTERPRETATION OF TERM SPECIFICITY AND ITS APPLICATION IN RETRIEVAL. **Journal of Documentation**, v. 28, n. 1, p. 11–21, 1 1972. ISSN 0022-0418. Available: <<https://www.emerald.com/insight/content/doi/10.1108/eb026526/full/html>>. Citations on pages 26 and 30.

KIM, M.; KANG, K.; PARK, D.; CHOO, J.; ELMQVIST, N. TopicLens: Efficient Multi-Level Visual Topic Exploration of Large-Scale Document Collections. **IEEE Transactions on Visualization and Computer Graphics**, IEEE, v. 23, n. 1, p. 151–160, 1 2017. ISSN 1077-2626. Available: <<http://ieeexplore.ieee.org/document/7539597/>>. Citations on pages 41, 43, and 48.

KLEINBERG, J. Bursty and hierarchical structure in streams. In: **Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '02**. New York, New York, USA: ACM Press, 2002. p. 91. ISBN 158113567X. Available: <<http://portal.acm.org/citation.cfm?doid=775047.775061>>. Citation on page 59.

KUANG, D.; PARK, H. Fast rank-2 nonnegative matrix factorization for hierarchical document clustering. In: **Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '13**. New York, New York, USA: ACM Press, 2013. Part F1288, p. 739. ISBN 9781450321747. Available: <<http://dl.acm.org/citation.cfm?doid=2487575.2487606>>. Citation on page 41.

LANG, K. NewsWeeder: Learning to Filter Netnews. In: **Machine Learning Proceedings 1995**. Elsevier, 1995. p. 331–339. Available: <<https://linkinghub.elsevier.com/retrieve/pii/B9781558603776500487>>. Citations on pages 39 and 74.

LAU, J. H.; BALDWIN, T. An Empirical Evaluation of doc2vec with Practical Insights into Document Embedding Generation. In: **Proceedings of the 1st Workshop on Representation Learning for NLP**. Stroudsburg, PA, USA: Association for Computational Linguistics, 2016. p. 78–86. Available: <<http://aclweb.org/anthology/W16-1609>>. Citation on page 66.

LE, Q. V.; MIKOLOV, T. Distributed Representations of Sentences and Documents. **Proceedings of the 31 st International Conference on Machine Learning**, v. 32, n. 2, p. 1188–1196, 5 2014. Available: <<http://arxiv.org/abs/1405.4053http://proceedings.mlr.press/v32/le14.pdf>>. Citations on pages 30, 32, 33, 64, 66, 82, and 83.

LEE, H.; KIHM, J.; CHOO, J.; STASKO, J.; PARK, H. iVisClustering: An Interactive Visual Document Clustering via Topic Modeling. **Computer Graphics Forum**, v. 31, n. 3pt3, p. 1155–1164, 6 2012. Available: <<http://doi.wiley.com/10.1111/j.1467-8659.2012.03108.x>>. Citations on pages 38, 39, 43, and 48.

LIU, T.; LIU, S.; CHEN, Z.; MA, W. Y. An Evaluation on Feature Selection for Text Clustering. **Proceedings, Twentieth International Conference on Machine Learning**, v. 2, p. 488–495, 2003. Citation on page 31.

LUXBURG, U. von. A tutorial on spectral clustering. **Statistics and Computing**, v. 17, n. 4, p. 395–416, 12 2007. ISSN 0960-3174. Available: <<http://link.springer.com/10.1007/s11222-007-9033-z>>. Citation on page 57.

MAATEN, L. van der; HINTON, G. Visualizing High-Dimensional Data Using t-SNE. **Journal of Machine Learning Research**, n. 9, p. 2579–2605, 8 2008. Available: <<http://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf>>. Citation on page 36.

MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. Efficient Estimation of Word Representations in Vector Space. p. 1–12, 1 2013. Available: <http://arxiv.org/abs/1301.3781>. Citations on pages 26 and 30.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G.; DEAN, J. Distributed Representations of Words and Phrases and their Compositionality. **Advances in Neural Information Processing Systems**, p. 1–9, 10 2013. Available: <http://arxiv.org/abs/1310.4546>. Citations on pages 30, 31, 64, and 66.

MUNZNER, T. **Visualization Analysis & Design**. [S.l.]: AK Peters, 2014. 428 p. ISBN 9781466508910. Citations on pages 25 and 41.

NOURASHRAFEDDIN, S.; SHERKAT, E.; MINGHIM, R.; MILIOS, E. E. A Visual Approach for Interactive Keyterm-Based Clustering. **ACM Transactions on Interactive Intelligent Systems**, v. 8, n. 1, p. 1–35, 2 2018. ISSN 21606455. Available: <http://dl.acm.org/citation.cfm?doid=3185338.3181669>. Citations on pages 26, 42, 43, 48, 63, 64, 73, and 74.

NSF, N. S. F. **NSF Award Search: Download**. 2020. Available: <https://www.nsf.gov/awardsearch/download.jsp>. Accessed: 2020-01-05. Citation on page 51.

PAATERO, P.; TAPPER, U. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. **Environmetrics**, v. 5, n. 2, p. 111–126, 6 1994. ISSN 11804009. Available: <http://doi.wiley.com/10.1002/env.3170050203>. Citation on page 40.

PAULOVICH, F.; NONATO, L.; MINGHIM, R.; LEVKOWITZ, H. Least Square Projection: A Fast High-Precision Multidimensional Projection Technique and Its Application to Document Mapping. **IEEE Transactions on Visualization and Computer Graphics**, v. 14, n. 3, p. 564–575, 5 2008. ISSN 1077-2626. Available: <http://ieeexplore.ieee.org/document/4378370/>. Citations on pages 35, 74, and 95.

PAULOVICH, F. V. **Mapeamento de dados multi-dimensionais - integrando mineração e visualização**. 1–144 p. Phd Thesis (PhD Thesis) — Universidade de São Paulo, São Carlos, 10 2008. Available: <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-04032009-145018/>. Citation on page 35.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; MÜLLER, A.; NOTHMAN, J.; LOUPPE, G.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 1 2012. ISSN 15324435. Available: <http://scikit-learn.sourceforge.net.http://arxiv.org/abs/1201.0490>. Citations on pages 67, 74, and 76.

PENNINGTON, J.; SOCHER, R.; MANNING, C. Glove: Global Vectors for Word Representation. In: **Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Stroudsburg, PA, USA: Association for Computational Linguistics, 2014. v. 19, n. 5, p. 1532–1543. ISSN 00047554. Available: <http://aclweb.org/anthology/D14-1162>. Citation on page 83.

PIROLI, P.; CARD, S. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In: **Proceedings of International Conference on Intelligence Analysis**. [s.n.], 2005. Available:

https://www.e-education.psu.edu/geog885/sites/www.e-education.psu.edu/geog885/files/geog885q/file/Lesson_02/Sense_Making_206_Camera_Ready_Paper.pdf. Citation on page 46.

POLIČAR, P. G.; STRAŽAR, M.; ZUPAN, B. openTSNE: a modular Python library for t-SNE dimensionality reduction and embedding. **bioRxiv**, p. 731877, 2019. Citation on page 76.

PORTER, M. An algorithm for suffix stripping. **Program**, v. 40, n. 3, p. 211–218, 7 2006. ISSN 0033-0337. Available: <https://www.emerald.com/insight/content/doi/10.1108/00330330610681286/full/html>. Citation on page 30.

REHUREK, R.; SOJKA, P. Software Framework for Topic Modelling with Large Corpora. In: **Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks**. Valletta, Malta: ELRA, 2010. p. 45–50. Available: <http://is.muni.cz/publication/884893/en>. Citation on page 66.

RIEHMANN, P.; HANFLER, M.; FROEHLICH, B. Interactive Sankey diagrams. In: **IEEE Symposium on Information Visualization, 2005. INFOVIS 2005**. IEEE, 2005. p. 233–240. ISBN 0-7803-9464-X. Available: <http://ieeexplore.ieee.org/document/1532152/>. Citation on page 55.

ROSENBERG, A.; HIRSCHBERG, J. V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure. **Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning**, n. June, p. 410–420, 2007. ISSN 1524-4040. Available: <http://acl.ldc.upenn.edu/D/D07/D07-1043.pdf>. Citation on page 75.

ROUSSEEUW, P. J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. **Journal of Computational and Applied Mathematics**, v. 20, n. C, p. 53–65, 11 1987. ISSN 03770427. Available: <https://linkinghub.elsevier.com/retrieve/pii/0377042787901257>. Citation on page 66.

RUOTSALO, T.; JACUCCI, G.; MYLLYMÄKI, P.; KASKI, S. Interactive intent modeling. **Communications of the ACM**, v. 58, n. 1, p. 86–92, 12 2014. ISSN 00010782. Available: <http://dl.acm.org/citation.cfm?doid=2688498.2656334>. Citation on page 25.

SHERKAT, E. **Interactive Text Analytics for Document**. 137 p. Phd Thesis (PhD Thesis) — Dalhousie University, 2018. Available: <https://dalspace.library.dal.ca/handle/10222/75022>. Citations on pages 31 and 64.

SHERKAT, E.; MILIOS, E. E.; MINGHIM, R. A Visual Analytics Approach for Interactive Document Clustering. **ACM Transactions on Interactive Intelligent Systems**, v. 10, n. 1, p. 1–33, 8 2019. ISSN 21606455. Available: <http://dl.acm.org/citation.cfm?doid=3352585.3241380>. Citations on pages 26, 31, 53, 54, and 61.

SHERKAT, E.; NOURASHRAFEDDIN, S.; MILIOS, E. E.; MINGHIM, R. Interactive Document Clustering Revisited. In: **Proceedings of the 2018 Conference on Human Information Interaction & Retrieval - IUI '18**. New York, New York, USA: ACM Press, 2018. p. 281–292. ISBN 9781450349451. ISSN 00207578 (ISSN). Available: <http://dl.acm.org/citation.cfm?doid=3172944.3172964>. Citations on pages 37, 42, 43, 44, 48, 63, 64, 73, and 74.

TEH, Y. W.; JORDAN, M. I.; BEAL, M. J.; BLEI, D. M. Hierarchical Dirichlet Processes. **Journal of the American Statistical Association**, v. 101, n. 476, p. 1566–1581, 12 2006. ISSN 0162-1459. Available: <http://www.tandfonline.com/doi/abs/10.1198/016214506000000302>. Citation on page 52.

TORGERSON, W. S. Multidimensional scaling: I. Theory and method. **Psychometrika**, v. 17, n. 4, p. 401–419, 12 1952. ISSN 0033-3123. Available: <http://link.springer.com/10.1007/BF02288916>. Citation on page 35.

TURNERY, P. D.; PANTEL, P. From frequency to meaning: vector space models of semantics. **Journal of Artificial Intelligence Research**, v. 37, p. 141–188, 2010. Citation on page 31.

VIÉGAS, F. B.; WATTENBERG, M. TIMELINE Tag clouds and the case for vernacular visualization. **interactions**, v. 15, n. 4, p. 49, 7 2008. ISSN 10725520. Available: <http://portal.acm.org/citation.cfm?doid=1374489.1374501>. Citations on pages 43 and 52.

VINH, N. X.; EPPS, J.; BAILEY, J. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. **Journal of Machine Learning Research**, v. 11, p. 2837–2854, 2010. ISSN 15324435. Citation on page 75.

WANG, C.; DANILEVSKY, M.; DESAI, N.; ZHANG, Y.; NGUYEN, P.; TAULA, T.; HAN, J. A phrase mining framework for recursive construction of a topical hierarchy. In: **Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '13**. New York, New York, USA: ACM Press, 2013. p. 437. ISBN 9781450321747. Available: <http://dl.acm.org/citation.cfm?doid=2487575.2487631>. Citation on page 54.

WANG, X.; LIU, S.; SONG, Y.; GUO, B. Mining evolutionary multi-branch trees from text streams. In: **Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '13**. New York, New York, USA: ACM Press, 2013. p. 722. ISBN 9781450321747. ISSN 9781450321747. Available: <http://dl.acm.org/citation.cfm?doid=2487575.2487631> <http://dl.acm.org/citation.cfm?doid=2487575.2487603>. Citation on page 54.

WATTENBERG, M.; VIÉGAS, F.; JOHNSON, I. How to use t-sne effectively. **Distill**, 2016. Available: <http://distill.pub/2016/misread-tsne>. Citation on page 76.

WEI, F.; LIU, S.; SONG, Y.; PAN, S.; ZHOU, M. X.; QIAN, W.; SHI, L.; TAN, L.; ZHANG, Q. TIARA. In: **Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '10**. New York, New York, USA: ACM Press, 2010. v. 3, n. 2, p. 153. ISBN 9781450300551. Available: <http://dl.acm.org/citation.cfm?doid=1835804.1835827>. Citations on pages 50, 51, and 61.

WIKIPEDIA. **Non-negative matrix factorization**. 2020. Available: https://en.wikipedia.org/wiki/Non-negative_matrix_factorization. Accessed: 2020-01-05. Citation on page 40.

WU, Y.; LIU, S.; YAN, K.; LIU, M.; WU, F. OpinionFlow: Visual Analysis of Opinion Diffusion on Social Media. **IEEE Transactions on Visualization and Computer Graphics**, v. 20, n. 12, p. 1763–1772, 12 2014. ISSN 1077-2626. Available: <http://lucene.apache.org/https://www.scopus.com/inward/record.uri?eid=2-s2.0-84910061860&partnerID=40&md5=ff6f64c687a8a4d42d854a6029096ac6http://ieeexplore.ieee.org/document/6876032>. Citations on pages 55, 56, and 61.

YANG, Y.; PEDERSEN, J. O. A Comparative Study on Feature Selection in Text Categorization. **International Conference on Machine Learning**, p. 412–420, 1997. Available: <http://www.aicit.org/aiss/global/paper_detail.html?jname=AISS&q=870>. Citation on page 31.

ZHANG, T.; RAMAKRISHNAN, R.; LIVNY, M. BIRCH. **ACM SIGMOD Record**, v. 25, n. 2, p. 103–114, 6 1996. ISSN 01635808. Available: <<http://portal.acm.org/citation.cfm?doid=235968.233324>>. Citation on page 68.

DOCUMENT PRE-PROCESSING ALGORITHM

In the proposed system, was developed a pre-processing algorithm that applies several filtering tasks and a complex lemmatizing procedure. The sequential filtering tasks are executed as follows:

1. Transform all words to lowercase.
2. Removes text marcations, such as HyperText Markup Language (HTML) or Extensible Markup Language (XML) tags.
3. Removes non-ASCII (American Standard Code for Information Interchange characteres to prevent encoding issues.
4. Removes punctuations.
5. Removes numbers.

Next, we combine the stop words publicly available¹ by the Natural Language Toolkit (NLTK) (BIRD; LOPER; KLEIN, 2009), and the stop words collected by Paulovich *et al.* (2008), specific to scientific articles related to the computer science field. With the resulting set, we remove stop words from the pre-processed text.

After the text is filtered, the words are tokenized and lemmatized. The tokenization step consists of the segmentation of sentences into its component words and its contractions, called tokens. For instance, the tokens for the sentence “I don’t love you” are: [“I”, “do”, “n’t”, “love”, “you”]. For lemmatization, we used the *WordNet Lemmatizer*, also publicly available by the NLTK library. This step reduces the words to a “lemma”, their minimal meaningful form. Since

¹ <https://www.nltk.org/>

words can be polysemic and may have multiple lemmas, the algorithm must identify which is the best-fitting lemma for each word. The WordNet Lemmatizer counts with a Part-of-Speech (POS) tag functionality that given the word's position in a sentence, the Lemmatizer extracts the correct lemma to the specified context.

The algorithm is implemented in the Python programming language in version 3.6, and the full code is displayed in [Source code 1](#).

Source code 1 – Pre-processing algorithm produced by the author. Python version: 3.6.

```

1: import string, re
2: from nltk import word_tokenize, pos_tag
3: from nltk.stem import WordNetLemmatizer
4: from nltk.corpus import stopwords, wordnet
5:
6: def strip_tags(data: str) -> str:
7:     p = re.compile(r'<.*?>')
8:     return p.sub('', data)
9:
10: def get_wordnet_pos(word):
11:     tag = pos_tag([word])[0][1][0].upper()
12:     tag_dict = {
13:         "J": wordnet.ADJ,
14:         "N": wordnet.NOUN,
15:         "V": wordnet.VERB,
16:         "R": wordnet.ADV
17:     }
18:
19:     return tag_dict.get(tag, wordnet.NOUN)
20:
21: def get_stopwords(langs: list) -> set:
22:     short = {
23:         "'s", "'ll", "n't", "'d", "'ve", "'m", "'re", ""
24:     }
25:
26:     stopword_list = []
27:     for lang in langs:
28:         stopword_list += stopwords.words(lang)
29:
30:     return set(stopword_list).union(short)
31:
32: def clean_text(text: str) -> str:
33:     punctuation = r"[{}]" .format(re.sub(r"[-']", "", string.

```

```
punctuation))
34:
35:     return " ".join(list(filter(lambda x: x != "", [
36:         re.sub(r"^\d+\$", r"", i) for i in re.findall(r"\S+",
37:         re.sub(punctuation, " ",
38:         re.sub(r'[\x00-\xb7f\xc0-\xff]', r' ',
39:         strip_tags(
40:         text.lower()
41:         )
42:         )
43:         )
44:         )
45:     ]))).rstrip()
46:
47:
48: def process_text(text: str, stopwords_langs:list=[]) -> str:
49:     text = clean_text(text)
50:
51:     stopwords = get_stopwords(["english"] + stopwords_langs)
52:
53:     lemmatizer = WordNetLemmatizer()
54:     return [
55:         lemmatizer.lemmatize(    ## LEMMATIZE
56:             token, get_wordnet_pos(token)
57:         ) for token in word_tokenize(text)    # TOKENIZE
58:         if token not in stopwords    # REMOVE STOPWORDS
59:     ]
```