

INTERACTIVE TEXT ANALYTICS FOR DOCUMENT  
CLUSTERING

by

Ehsan Sherkat

Submitted in partial fulfillment of the requirements  
for the degree of Doctor of Philosophy

at

Dalhousie University  
Halifax, Nova Scotia  
November 2018

© Copyright by Ehsan Sherkat, 2018

# Table of Contents

<b>List of Tables</b> . . . . .	<b>v</b>
<b>List of Figures</b> . . . . .	<b>vii</b>
<b>Abstract</b> . . . . .	<b>viii</b>
<b>Chapter 1 Introduction</b> . . . . .	<b>1</b>
1.1 Contributions . . . . .	3
1.2 Outline . . . . .	5
<b>Chapter 2 Interactive Document Clustering</b> . . . . .	<b>7</b>
2.1 Introduction . . . . .	7
2.2 Related Work . . . . .	9
2.2.1 Constrained Clustering Algorithms . . . . .	9
2.2.2 Interactive Clustering Algorithms . . . . .	10
2.2.3 Constrained Topic Modeling . . . . .	10
2.2.4 Interactive Topic Modeling Systems . . . . .	11
2.2.5 Interactive Document clustering Systems . . . . .	12
2.3 Overview of the Proposed System . . . . .	16
2.3.1 Document Preprocessing . . . . .	17
2.3.2 Document Clustering . . . . .	19
2.3.3 Document Projection . . . . .	22
2.3.4 Visual Components . . . . .	27
2.4 Experiments . . . . .	30
2.4.1 Quantitative Evaluation . . . . .	31
2.4.2 Usage Scenario . . . . .	34
2.4.3 Use Case . . . . .	38
2.4.4 User Study . . . . .	39
2.5 Conclusion . . . . .	44
<b>Chapter 3 Deterministic Seeding of KMeans</b> . . . . .	<b>45</b>
3.1 Introduction . . . . .	45
3.2 Proposed Method . . . . .	48
3.3 Baseline Methods . . . . .	52

3.4	Datasets and Evaluation Metrics . . . . .	54
3.5	Experimental Results . . . . .	56
3.6	Conclusion . . . . .	59
<b>Chapter 4</b>	<b>Interactive Temporal Document Clustering . . . . .</b>	<b>60</b>
4.1	Incorporating Temporal Aspect in the Clustering Algorithm . . . . .	60
4.1.1	Temporal Similarity . . . . .	60
4.1.2	Interactive Temporal Document Clustering . . . . .	63
4.2	Incorporating Temporal Aspect in the Visualization . . . . .	69
4.2.1	Visualization Modules . . . . .	69
4.2.2	Case Study . . . . .	70
4.3	Conclusion . . . . .	76
<b>Chapter 5</b>	<b>Wikipedia Based Semantic Similarity . . . . .</b>	<b>77</b>
5.1	Vector Embedding of Wikipedia Concepts . . . . .	78
5.1.1	Introduction . . . . .	78
5.1.2	Related Work . . . . .	79
5.1.3	Distributed Representation of Concepts . . . . .	81
5.1.4	Evaluation . . . . .	83
5.2	Conclusion . . . . .	87
<b>Chapter 6</b>	<b>Conclusion and Future Research . . . . .</b>	<b>89</b>
6.1	Conclusion . . . . .	89
6.2	Future Research . . . . .	91
	<b>Bibliography . . . . .</b>	<b>93</b>
	<b>Appendix A Evaluation Metrics . . . . .</b>	<b>103</b>
	<b>Appendix B User Study Design . . . . .</b>	<b>106</b>
B.1	Screening Questionnaire . . . . .	106
B.2	Demographic Questionnaire . . . . .	108
B.3	Interface Features Questionnaire . . . . .	109
B.4	Interface Rating Questionnaire . . . . .	112

B.5 Dalhousie Ethic Board's Letter of Approval . . . . .	113
<b>Appendix C Copyright Permissions . . . . .</b>	<b>117</b>



## List of Tables

2.1	A summary of different types of interactions between the user and clustering algorithm. . . . .	14
2.2	A summary of common visualization modules used for document cluster and topic visualization. . . . .	15
2.3	The impact of <i>bigram</i> and <i>unigram</i> on the clustering result . . .	33
2.4	The comparison result of clustering algorithms with random initialization . . . . .	33
2.5	The role of seeds (documents) quality in the result of document clustering. The <i>NG5</i> dataset is used in this experiment. . . . .	34
2.6	The Silhouette indices of clusterings generated on <i>NewsSeparate</i> in each iteration . . . . .	35
2.7	The improvement of the Avg. Silhouette after each interaction.	39
2.8	Statistics about the user study . . . . .	41
2.9	Comparing the result of the system with (Vis.) and without (Base) visualization . . . . .	42
2.10	The Post-task questionnaire . . . . .	43
3.1	Description of Datasets . . . . .	53
3.2	Comparing precision of seeds . . . . .	54
3.3	Comparing clustering accuracy . . . . .	55
3.4	Comparing clustering NMI score . . . . .	56
3.5	Running time (seconds) of seeding methods . . . . .	57
4.1	List of notations . . . . .	61
4.2	Statistics of the Datasets used in the Temporal Similarity experiments. . . . .	62
4.3	The comparison between different similarity methods and their impact on clustering performance . . . . .	63

4.4	The comparison between different similarity methods and their impact on the clustering performance . . . . .	64
4.5	Comparing the Accuracy of the proposed key-term based interactive clustering algorithm . . . . .	68
4.6	Top terms of final clustering result of the email list dataset . .	73
4.7	Most frequent operations that the expert conducted during the case study . . . . .	74
4.8	Clustering Silhouette after each re-clustering request from the expert. . . . .	76
5.1	Top similar terms to <i>amazon</i> . . . . .	78
5.2	Comparing the results of three different versions of ConVec . .	82
5.3	Comparing the results in Phrase Similarity dataset . . . . .	85
5.4	Comparing the results in the Phrase Similarity datasets for the common entries between all approaches. . . . .	87

## List of Figures

2.1	The proposed system overview. . . . .	17
2.2	The comparative result of t-SNE and the t-SNE combined with Force-directed . . . . .	24
2.3	The user has an access to both augmented t-SNE and only Force-directed placement . . . . .	25
2.4	The visual interface of the proposed system . . . . .	28
2.5	The Document's glyph in <i>Graph view</i> . . . . .	30
2.6	The impact of the confidence level on the clustering result . . . . .	34
2.7	Use case 1: clustering <i>NewsSeparate</i> - Iteration 1 . . . . .	35
2.8	Use case 1: clustering <i>NewsSeparate</i> - Iteration 2 . . . . .	37
2.9	Use case 1: clustering <i>NewsSeparate</i> - Iteration 3 . . . . .	37
2.10	Use case 1: clustering <i>NewsSeparate</i> - Iteration 4 . . . . .	38
2.11	The screen-shots of different interaction rounds conducted by the user . . . . .	40
2.12	The comparison between the initial clustering and the final clustering after user interactions . . . . .	41
3.1	The comparative result of pairwise cosine and <i>dsim</i> similarity of Newsgroup5 dataset . . . . .	49
3.2	The impact of number of initialization on the Accuracy performance and running time . . . . .	58
4.1	The two modes of the Temporal view . . . . .	69
4.2	The heat map of the expert clicks and mouse movements . . . . .	75

## Abstract

Clustering has been widely used to efficiently get insight into text collections containing more documents than a human can effectively read. Although there exist several different document clustering algorithms, most of them do not consider user preferences. Due to the personalized nature of document clustering, even best algorithms cannot create clusters that accurately reflect the user’s perspectives. On the other hand, it is necessary to visualize the results of clustering to be easily interpretable by the human. In this thesis, we revisit the problem of document clustering to incorporate the user’s perspective in the clustering process and effectively visualize data in the process of being clustered to enhance user’s sense-making of the data. First, we design clustering algorithms that are interactive and can adapt to the user’s feedback. Second, a collection of coordinated visualization modules and document projection is designed to guide the user towards a better insight into the document collection and the clustering algorithm results. It has been demonstrated that exploiting external knowledge sources such as Wikipedia can help the clustering algorithm to consider the semantic similarity between documents. The process of linking terms and phrases of a document to the related Wikipedia page is called Wikification of a document. To help the process of Wikification, we introduce a model to extract high-quality distributed vector representations for each Wikipedia page. Finally, we considered the temporal similarity between documents and introduced a couple of visualization modules to depict the temporal aspect of clusters. This has enabled us to study the dynamics of document clusters over time. A set of quantitative experiments, use cases, and a user study has been conducted on real-world datasets to show the advantages of interactive visual analytics clustering.

# Chapter 1

## Introduction

A huge amount of text documents are produced each day, resulting in the generation of many documents per subject. One of the most effective solutions to manage and get insight into this huge amount of data is by automatically clustering them into meaningful clusters. There exist several clustering algorithms for documents, which try to group similar documents together with the help of different similarity measures. However, it is not always possible to automatically create the clustering that best matches the user's point of view or the application's goals, even with state-of-the-art clustering algorithms. The reason is that the clustering problem is a NP complete problem and we may find a local optimum instead of the global solution. A sensible solution to this problem is involving the human in the loop of clustering. In this approach, it is possible to generate clusters close to the user's perspectives, while in the process supporting the human in making sense of the document collection [57, 47, 8, 6]. An interactive clustering system needs to answer questions such as: What are the interaction types between the user and the clustering algorithm? Which visualization modules should be used? How to visualize the clustering result? How fast should be the clustering algorithm? These are questions that we tried to answer in Chapter 2.

There are several semi-supervised clustering algorithms that use a few labeled instances to improve the quality of their results [11, 101]. Some other algorithms exploit the user's feedback in terms of the split or merge signals [6]. In that case, the user only asks to merge two clusters or split a cluster without specifying how to split the cluster. Most of these approaches have been tested in theory, but not used in everyday activities. In practice, an interactive clustering algorithm that gives the user an intuitive way of interaction and is feasible in terms of user effort would be more appropriate. On the other hand, the constrained clustering algorithms assume that constraints or user feedback are known before clustering starts, and they do

not support, in their set up, interaction during clustering or progressive clustering improvement by the user, preventing knowledge acquisition during the process.

Topic modeling, sometimes viewed as soft clustering of documents, is a different way of categorizing similar content by extracting meaningful topics from the document collection. In reality, it is sometimes hard to know how distinct the topics generated by topic modeling really are, which makes it difficult for user interactions. A few research papers have proposed interactive topic modeling systems for the end user by providing different visualization modules [57, 29]. In spite of these achievements, there is still a need to propose more effective visualization and interaction components to bring better insight into a document collection for the user.

We propose a practical solution for interactive document clustering by combining an interactive clustering algorithm with a visual interface, that, besides offering a meaningful way to understand and tailor the clustering process, is equipped with a set of visual functionalities that facilitate the reasoning and knowledge acquisition of the data set. For that we have chosen key-term based interaction because it has been demonstrated that not only is it effective in improving the results but it is more intuitive for the user to interact with the clustering algorithm [13, 73, 60, 75]. In our key-term based interaction method, the user only assigns a short list of key-terms (less than five) to the desired cluster to guide the clustering algorithm. Ability to interact with the clustering algorithm with a very short list of key-terms distinguishes the proposed method from other key-term based interactive systems. Additionally, interactions are convertible to each other. For example, the document-based interaction can be achieved easily by assigning top key terms of the selected document(s) to a cluster as opposed to just directly linking the document(s) to a cluster.

We have designed a visual interface to integrate the key-term based user interactions with the clustering algorithms as well as to allow the users to explore the data set and the relationships between documents and clusters. Several visual components are developed to guide the user for more effective interaction with the clustering algorithm and give the user better insight into the document collection. We combined the t-distributed stochastic neighbor embedding (t-SNE) [65] with the Force-directed placement [37] in a novel way to better improve cluttering typical of such embeddings

and provide a better distinction among clusters. We have conducted various experiments using real-world datasets including a use case and a user study to demonstrate the effectiveness of the proposed system.

KMeans is usually initialized by random seeds that can drastically impact the final algorithm performance. There exist many random or order-sensitive methods that try to properly initialize KMeans but their problem is that their result is non-deterministic and unrepeatable. Thus KMeans needs to be initialized several times to get a better result which is a time-consuming operation. We introduce a novel deterministic seeding method for KMeans that is specifically designed for text document clustering in chapter 3. Due to its simplicity, it is fast and can be scaled to large datasets.

Each document has a creation time. If one considers the document temporal aspect and not only cluster documents similar in content but also temporally similar, it is possible to extract the trends and evolution of clusters over time. In Chapter 4, we try to consider the temporal dimension of documents while clustering document collection. The user can select a single time span for the dataset when the temporal aspect is not important for her. To achieve this goal, first the clustering algorithm needs to be adapted to the temporal dimension of data, second the visualization modules need to visualize the evolution of clusters over time. We evaluate this system on a dataset that belongs to the Canadian Society of Respiratory Therapists which includes email messages between respirator therapists in a shared mailing list.

In addition to involving human in the process of clustering, considering semantic similarity can also improve the result of document clustering. This idea is mostly studied in Chapter 5 where we introduce a method for representing Wikipedia concepts with low dimensional vectors. These Wikipedia concept vectors can be used for linking concepts of documents to the related Wikipedia page. In this case, we can disambiguate the meaning of concepts in document collection which is somehow using the semantic of text for document clustering.

## 1.1 Contributions

- We introduce a new approach for representing Wikipedia concepts with a vector. We evaluate the performance of the proposed method based on Concept Analogy

and Concept Similarity tasks.

- We explain the design of a novel visual interface to integrate the key-term based user interactions with the clustering algorithms as well as to allow the users to explore the data set and the relationships between documents and clusters. Several visual components such as the Document Cloud and Temporal View will be developed to guide the user for more effective interaction with the clustering algorithm and give the user better insight into the document collection.
- We explain the process of combining the t-distributed stochastic neighbor embedding (t-SNE) [65] with the Force-directed placement [37] in a novel way to better improve distinction among clusters.
- Novel preprocessing steps for preparing email conversations for interactive document clustering is introduced.
- A case study on a real world listserv dataset is conducted to further evaluate the proposed system in a practical applications. We asked a Registered Respiratory Therapist (RRT) to conduct the case study on the dataset of email messages belonging to the Canadian Society of Respiratory Therapists email list. The goal of this case study was to apply the proposed method to help Respiratory Therapists in the process of evidence-based decision making in improving patient care. This process consists of three main components: clinical expertise, patient values, and the best scientific research evidence [95]. The proposed method will be helpful for combining sources of clinical expertise in the process of evidence-based decision making. The importance of electronic mailing lists for sharing information and experiences has been reported in several research papers [63, 97, 58].
- We explain how to incorporate the temporal similarity in the process of document clustering.
- Conducting various experiments using real-world datasets and a user study to demonstrate the effectiveness of the proposed interactive document clustering system.



- Introducing a new approach for initializing the clustering algorithms. Previously, the clustering algorithms were initialized randomly in the first iterations. Then, in the next iterations, they were initialized by the user feedback in terms of key-terms. The non-deterministic clustering resulting from a random initialization can confuse the user and makes the clustering not reproducible. The new initialization method is not only deterministic but it is faster and guides the clustering algorithm to converge to better clustering result.
- Most of the code developed in this thesis are open source and can be publicly accessed on the author's Github page<sup>1</sup>.

## 1.2 Outline

The overall structure of the thesis report is in the following.

**Chapter 2** In this chapter we introduce a novel Interactive Document Clustering System. The proposed system has two main parts: The clustering algorithm and visualization modules. Different interactions between the user and the clustering algorithm can be divided to three categories: document interaction, key-term interaction and hybrid key-term and document interactions. key-term based interactions are easier for the user and enables her to group a bunch of documents by a few key-terms [73]. In our system, we select clustering algorithms that can interact by key-terms. In the visualization section, we design different modules that can help the user to get better insight from the document collection and help her to find the desired key-terms effectively. We evaluate the proposed method by quantitative experiments, usage scenarios and a user study. In the design of the user study, we tried to both evaluate the effectiveness of the visualization for efficient visualizing of document collection and the quality of final clustering before and after user interactions.

**Chapter 3** In this chapter, we focus on the initialization part of the clustering algorithm. Because the KMeans is non-convex in nature it is sensitive to the initial

---

<sup>1</sup><https://github.com/ehsansherkat/IDC>

seeds. Most of the existing initialization methods for KMeans are random or order-sensitive. In case of interactive document clustering, if the result of initial clustering changes due to its initialization method, it can confuse the user. To overcome this problem, we design a fast and simple initialization method which is deterministic. Because of deterministic feature of this method, it only needs to be initialized once which makes the clustering process faster.

**Chapter 4** This chapter is the extension of Chapter 2 and the goal is to consider the temporal aspect of documents in the process of document clustering. In this case, it is possible to extract the trends and evolution of clusters over time. Similar to Chapter 2, we will focus on the KMeans like clustering algorithms and try to incorporate the temporal feature of the data both in the similarity measure of the clustering algorithm. In this chapter, we involved the user feedback (key-terms) in the objective function of the KMeans algorithm. In case of violating the user preferred feedback a penalty term will be assigned to the objective function.

**Chapter 5** The main goal of this chapter is improving the quality of document clustering by incorporating the semantic similarity. Linking concepts in document to external resources such as Wikipedia can be used for semantic similarity. One of the key steps in linkage process is disambiguating the concepts with several surface forms. The higher quality of document clustering is reported by considering the bag of concepts [44]. In the first step, we use word embedding methods for representing Wikipedia concepts with a vector. These vectors can be used in the process of disambiguation and consequently help to better quality bag of concepts.

## Chapter 2

### Interactive Document Clustering

Document clustering is an efficient way to get insight into large text collections. Due to the personalized nature of document clustering, even the best fully automatic algorithms cannot create clusters that accurately reflect the user’s perspectives. To incorporate the user’s perspective in the clustering process and, at the same time, effectively visualize document collections to enhance user’s sense-making of data, we propose a novel visual analytics system for interactive document clustering. We built our system on top of clustering algorithms that can adapt to user’s feedback. First, the initial clustering is created based on the user-defined number of clusters and the selected clustering algorithm. Second, the clustering result is visualized to the user. A collection of coordinated visualization modules and document projection is designed to guide the user towards a better insight into the document collection and clusters. The user changes clusters and key-terms iteratively as a feedback to the clustering algorithm until the result is satisfactory. The user is satisfied with the result if she does not want to apply any new changes to the clusters. In key-term based interaction, the user assigns a set of key-terms to each target cluster to guide the clustering algorithm. A set of quantitative experiments, a usage scenario, a use case, and a user study have been conducted to show the advantages of the approach for document analytics based on clustering<sup>1</sup>.

#### 2.1 Introduction

Document clustering is one of the key modules in Visual Text Analytics. The goal of document clustering is to group similar documents without having any prior knowledge about their labels. Document clustering has many applications in different fields from news trends to health analytics. It is also a very important first step for other visualizations and mining strategies. Despite numerous text clustering algorithms

---

<sup>1</sup>Part of this chapter is published in [92] (best student paper award recipient), [93], and [75].

and also statistical techniques to measure quality of clusters, user-supervised clustering is preferred to totally unsupervised clustering in analytic applications. However, presenting documents and collecting feedback from the user require an interactive clustering algorithm with easy to learn parameters as well as a well-designed visualization platform.

We propose a practical solution for interactive document clustering by combining an interactive clustering algorithm with a visual interface. We have chosen key-term based interaction because it has been demonstrated that not only is it effective in improving the results but it is more intuitive for the user to interact with the clustering algorithm [13, 73, 60, 75]. In our key-term based interaction method, the user only assigns a short list of key-terms (less than five) to the desired cluster to guide the clustering algorithm. Ability to interact with the clustering algorithm with a very short list of key-terms distinguishes the proposed method from other key-term based interactive systems. Additionally, interactions are convertible to each other. For example, document based interaction can be achieved easily by assigning top key terms of the selected document(s) to a cluster as opposed to just directly linking the document(s) to a cluster.

Lexical Double Clustering (LDC) [74] and a novel KMeans style interactive clustering algorithm called iKMeans are incorporated in our proposed system. We have selected document clustering algorithms instead of topic modeling ones to avoid problems of inconsistency after several runs, empirical convergence criteria and difficulty to interact with their complicated formula and algorithm. LDC has shown better performance compared to state-of-the-art topic modeling algorithms such as Latent Dirichlet Allocation (LDA) [18] and Non-negative Matrix Factorization (NMF) [76] on several standard document clustering datasets. Both LDC and iKMeans are deterministic in each interaction and generate similar results for the same key-term based user feedback. Because of the simple nature of KMeans-like algorithms, it is easy to incorporate user feedback and scale the algorithm to very large datasets with a short running time, which is hard to achieve in topic modeling algorithms.

In the proposed system, it is possible to switch between the LDC and iKMeans algorithms on demand. Interactively switching algorithms during clustering is not used in any other interactive clustering system. For example, the user can start with

the LDC algorithm which is robust to outliers and when the user gets sufficiently familiar with the dataset, switch to iKMeans to generate highly customized clusters. The ability to alternate between different clustering algorithms demonstrates the independence of our system of the specific clustering algorithm, which means it can incorporate any other key-term based interactive clustering algorithm.

## 2.2 Related Work

Each interactive clustering system consists of two integrated parts: an interactive clustering algorithm and a visual interface. The main focus of some research papers is in the clustering algorithm, some in the visualization part and others focus on the integration of both. In the following we targeted these three types of related work.

### 2.2.1 Constrained Clustering Algorithms

Semi-supervised clustering algorithms use labeled data to guide the clustering process. Two semi-supervised clustering algorithms, Seeded-KMeans and Constrained-KMeans, inspired by the KMeans algorithm, were introduced in [11]. In Seeded-KMeans, instead of randomly initializing the KMeans, the labeled data points are used to initiate the clustering. Constrained-KMeans is similar to the seeded one but it keeps the label of seeds unchanged in every iteration of the algorithm. These two algorithms are modeled based on the Expectation-Maximization (EM) algorithm on a mixture of  $k$  (number of clusters) Gaussians. Each data point has  $k$  possible conditional distributions and the initial supervision is to determine these conditional distributions for seed points. These two algorithms showed better performance in comparison to COP-KMeans [101]. The supervision in COP-KMeans is in terms of *must-link* (two data points must be in the same cluster) and *cannot-link* (two data points cannot be in the same cluster) constraint. In each step of KMeans, partitions are generated in a way that satisfies all the given constraint. The problem with these algorithms is that they assume that all the constraints are known in advance, hence they are not designed for interactive use.

## 2.2.2 Interactive Clustering Algorithms

A class of interactive clustering algorithms captures all the necessary interactions between the user and the clustering algorithm as cluster *split* and *merge* queries [8]. The user sends a Merge request if two clusters are a subset of a target cluster and a split request if a cluster contains data points belonging to distinct targeted clusters. The user does not enter how to split or point out possible mistakes in the have led to the request. Several different merge and split operations are introduced by Awasthi et al. [6] to reduce the number of merge and split requests. We believe that for textual data the user should not only be able to send the split request but also to specify how to split the cluster by providing some key-terms, aiming to reduce the number of split and merge requests. These algorithms have not been equipped with visualization or tested by end users.

## 2.2.3 Constrained Topic Modeling

*Latent Dirichlet Allocation* (LDA) [18] has been widely used for topic modeling. LDA models topics as distributions over words and documents as distributions over topics. One can consider the most probable topic of a document as a document cluster label. LDA has been extended to incorporate domain knowledge into the topic modeling through must-link and cannot-link constraints over terms [4]. The *must-link* constraint between two terms indicates that they tend to be generated by the same topic, while the cannot-link constraint says that two terms tend to be generated by different topics. The *Dirichlet forest prior* is used to encode must-link and cannot-link constraints in which the prior is a mixture of Dirichlet tree distributions. The must-link constraints only were used as a supervision to the LDA in [104]. They used a collection of tree-structured (based on *must-link* terms) multinomial distributions instead of multinomial distributions over words. Must-link and cannot-link constraints over documents were incorporated in the LDA model by changing the Gibbs sampling of the original LDA [104]. Contrary to the original LDA, the document topic distribution prior is updated in every iteration based on user feedback. They applied this system to improve topic model stability after adding new documents to the model.

A supervised version of LDA that incorporates labeled documents in the model has been proposed by [81]. In this model, the user can improve the quality of LDA

by providing some topics for documents. These algorithms are not designed to obtain the user feedback in an interactive way. One of the major problems of topic modeling algorithms is inconsistency of results after each iteration which makes them difficult to use interactively. Algorithms described in this subsection have not been tested by the end user with the help of a visualization.

#### 2.2.4 Interactive Topic Modeling Systems

*iVisClustering* is an interactive clustering system that introduced a visual interface on top of the LDA algorithm to involve the user in the loop [57]. The interaction with the LDA is performed only by changing terms' weights. This system has several different visualization modules called views. The *Graph View* using force-directed layout shows the general view of the document collection. The summary of clusters with their top terms are depicted in rectangle shaped boxes called *Cluster Summary View* beside the hierarchical style visualization of clusters (*Cluster View*). The user hands over a cluster in *Cluster Summary View* and the system shows document grids with the color spectrum which depicts their relatedness to the cluster. By clicking on a grid, the relatedness of that document to other clusters will be shown in a *Parallel Coordinates View* plus the content of documents in a *Document View*. The list of top terms of each cluster is in the *Term-Weight View* in which the user can change the weight of each term and impact the result of clustering. Because of the complicated nature of LDA formulas the interaction with this algorithm is not straightforward so changing term weights may confuse the user. In our system, the user only needs to define a set of key-terms without any need to assign a weight to them. Relative importance of terms is given by their order, without having to assign values. In addition to their lack of intuitiveness, topic modeling algorithms suffer from inconsistency in the results after each interaction. In contrast to *iVisClustering*, in our proposed system, all the visualization modules are coordinated with each other.

*UTOPIAN* [29] is an interactive topic modeling system, which is based on Non-negative Matrix Factorization (NMF) [76] instead of LDA. The interaction with NMF is based on changing term weights. Authors used Graph View, Document View, and Term-Weight View as supporting visualizations. In the Graph View, the location of

nodes is assigned by t-SNE [65]. The t-SNE algorithm is a method for dimensionality reduction mostly used for 2D and 3D projection of data points. UTOPIAN is developed by the same group of developers of iVisClustering system. In this system, they used NMF instead of LDA to handle the inconsistency problem of LDA algorithms. UTOPIAN still suffers from the empirical convergence problem and difficulty to meaningfully interact because of their complicated formula and algorithm. Unlike our proposed system, neither UTOPIAN nor iVisClustering were evaluated by end users. The overall differences between the key-term based document clustering systems and the UTOPIAN and the iVisClustering is described in more details in [75].

### 2.2.5 Interactive Document clustering Systems

An active learning scheme for selecting seed documents to be labeled by the user is presented in [45]. These seed documents will be used as an input to a semi-supervised KMeans algorithm. To facilitate the process of finding seed documents, a visualization is designed, which contains a *Term Cloud View* of each document and cluster, *Document View*, and *Pie Visualization* of clusters. In the *Pie Visualization*, each slice is a cluster and in the middle of the pie, there is a circle which contains all unlabeled documents. The user can assign these unlabeled documents to a desired cluster. *TopicPanorama* has also used a pie visualization for topics. All these topics are sub-topics of a few major topics determined by the user. Topics are linked together based on graph matching techniques in this system. This sub-topic graph structure is inside a hollow circle named *Radial Icicle Plot* in which for each major topic there is a color-coded arc. The user can zoom in on a topic by changing the length of the arc. Subtopics more similar to the major topics are near its arc and the common subtopics are in the middle of the circle. In our proposed system, we used key-term based interactions, which is easier and more effective than document based interactions [75].

In our proposed system, we focus on the key-term based interaction which is more effective and intuitive for the user than the document based interaction or a hybrid one [75]. In this paper, we present a flexible framework for key-term based document clustering where users can interact based on key-terms with more than one word (e.g. bigrams); a new key-term based clustering algorithm is introduced, which shows how to combine a key-term interaction with document supervision; documents projection



is improved by combining (t-SNE) with the Force-directed placement in order to distinguish groups of documents better; and we have conducted a user study that evaluates both the effectiveness of this type of interface for supervised clustering and the overall interactive document clustering approach.

The summary of different types of interactions between the user and the clustering algorithm is in Table 2.1. These interactions are the most commonly used interactions. Most of the algorithms incorporated KMeans, LDA, and NMF as a base algorithm and tried to incorporate user interactions in the loop. In terms of visualization, different common types of views are summarized in Table 2.2. Some of these visualization modules are designed for data visualization systems and some are appropriate for interactive systems.

The summary of different types of interactions between the user and the clustering algorithm is described in Table 2.1. These interactions are the most commonly used interactions and can mainly be categorized into three types of 1) seeding 2) must-link and cannot-link constraints 3) split and merge signals. In the seeding method, the algorithm gets some labeled data such as labeled documents, labeled topics, or labeled terms from the user to better cluster the dataset. Must-link and cannot-link constraint methods are sometimes called a Constrained clustering algorithm. A must-link constraint indicates that two data instances must be in the same group or cluster while cannot-link constraint emphasizes that two data point should not be associated in the same cluster. Optimization methods such as Integer Linear and Quadratic Programming is applied to satisfy the constraints [54]. In split and merge signal method, the user asks to merge two clusters or split a cluster to half without explaining how to do the split or merge. In our proposed method, the user can merge two clusters by joining the top key-terms of them in the same cluster or can split a cluster by separating the top key-terms of a cluster into two clusters. In this way, the user not only is able to split or merge clusters but she can also indicate how to do the merge and split. In Table 2.1, most of the algorithms incorporated KMeans, LDA, and NMF as a base algorithm and tried to incorporate user interactions in the loop.

In terms of visualization, different common types of visualization modules are summarized in Table 2.2. Some of these visualization modules are designed for data

Table 2.1: A summary of different types of interactions between the user and clustering algorithm.

#	Type of Interaction	Description	Base Algorithm
1	Seed documents	Providing labeled document	KMeans [11, 45]
2	Seed terms	Providing seed terms or changing the weight of terms	CMRF [13], LDA [57], NMF [29, 51]
3	Seed topics (labels)	Providing a list of seed topics	LDA [81]
4	Seed feature selection (Aspect based)	The user selects which set of features best describes his/her point of view.	CMRF [13]
5	Must-link documents	Documents pairs that should be in the same cluster	KMeans [101], LDA [104]
6	Cannot-link documents	Documents pairs that should not be in the same cluster	KMeans [101], LDA [104]
7	Must-link terms	A set of terms should be in the same cluster or topic	LDA [4, 47]
8	Cannot-link terms	A set of terms should not be in the same cluster or topic	LDA [4]
9	Split cluster	The user asks to split a cluster but not says how to split it	No base [8, 6]
10	Merge cluster	The user ask to join two clusters	No base [8, 6]

Table 2.2: A summary of common visualization modules used for document cluster and topic visualization.

#	Visualization module name	Description	Reference
1	Graph view	General view of Document collection, which nodes are documents color-coded by their assigned cluster. Similar documents are linked by edges.	[57, 29, 51, 61]
2	Cluster (summary) view	A summary of clusters with their important terms.	[57]
3	Cluster Tree view	Depicting clusters structure in a tree style	[57]
4	Parallel Coordinates view	Demonstrating the relatedness of documents or terms to other clusters.	[57]
5	Document View	A view for showing the content of documents.	[57, 29, 30]
6	Term list view	Providing list of terms plus their weight using bar charts or just their value.	[57, 29, 30]
7	Document Tracer view	Following documents cluster change with a heat map style visualization.	[57]
8	Term cloud view	Depicting important terms based on font size and color.	[14, 61]
9	Topic Rose Tree view	Documents' topics are depicted in a hierarchical fashion in which user can interactively explore topics with three operations of <i>Join</i> , <i>Absorb</i> and <i>Collapse</i>	[34]
10	Hierarchical ThemeRiver view	This view is used to demonstrate the temporal evolution of topics.	[34]
11	Pie visualization	In the Pie Visualization, each slice is color-coded cluster.	[45, 61]
12	Scatter plot	Using scatter plot to visualize documents with their membership color-coded. In this view, there are no links between each document.	[14, 51]
13	Matrix	Visualizing document-topics or word-topic matrix	[30, 3]

visualization systems and some are appropriate for interactive systems. The Graph view is one of the major visualization components used in several systems to help the user bird’s eye view of all documents and their relation. Document projection methods such as t-SNE, Principle Component Analysis (PCA) and Force-directed placement are the most common methods applied in the graph view. Sometimes Scatter plot is used to give the user a general view of documents. In this case, the link between documents is not indicated. In our proposed method, we combined the t-SNE and Forced-directed placement in a novel way to better visualize all document and their relations in a single graph. Users usually like to access the content of each document and this is why several systems have a document view. The summary of the content of a cluster can be presented by a list of terms and because of that, it is common to have a Term-list view. In our proposed method, we elaborated several views such as Graph view, Document view, Term-list view, and Parallel coordinates view with the customized features in an integrated and novel way to help the user to better sense-making of the dataset and at the same time to select better key-terms to interact with the clustering algorithm.

### **2.3 Overview of the Proposed System**

The clustering is a personalized task in nature and even the best fully-automatic clustering algorithm may not reflect the user perspectives. Because of that, the clustering algorithm needs guidance from the user to fully adapt to the user needs. On the other hand, to effectively interact with the clustering algorithm the user needs a good visualization of the result of the clustering. To tackle these problems, we proposed a novel visual analytics system which has a highly interactive visual interface to facilitate the user interaction with the clustering algorithm.

The proposed system is a web-based user-centered document clustering system that integrates key-term based clustering algorithms with an interactive visualization (Fig. 2.1). First, the user uploads documents, which the system preprocess and provides an initial fully automatic clustering. Second, the user obtains insight into the document collection by inspecting the visual components including the document projection. Third, the user provides key-term based feedback to the clustering algorithm to guide the result of clustering. In the following, each component of the

system is described.

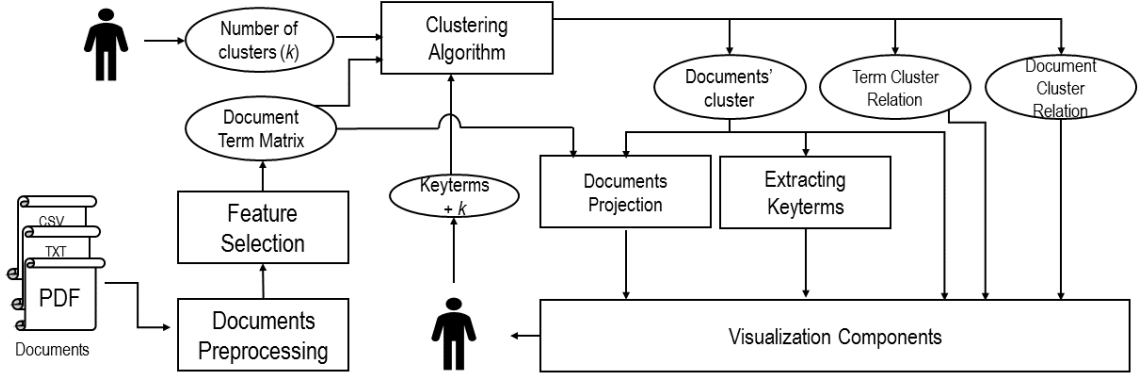


Figure 2.1: The proposed system overview.

### 2.3.1 Document Preprocessing

In the first step, we extract the plain text of each documents after removing punctuations and numbers. In the next step, we remove useless terms from the document collection dictionary. Even a few documents can lead to several thousand unique terms. This will cause a negative impact on the quality of document clustering [59] and increases the clustering time which is very crucial for interactive systems. In order to tackle this problem, we used an unsupervised feature selection method based on terms *tf-idf* score. This score shows the discriminative power of a term over each document [67]. Let  $D$  be the set of documents and  $d$  a document in  $D$ , the *tf-idf* score of the term  $w$  is as Equation 5.1.

$$tf\_idf(w, d, D) = f(w, d) \times \log \frac{|D|}{|d \in D : w \in d|} + 1. \quad (2.1)$$

The  $f(w, d)$  is the frequency of term  $w$  in document  $d$ . Now each document has a vector of terms' *tf-idf* score. We normalize each document vector based on the *Euclidean (L2) norm*. For each term, the *mean-tf-idf* score is calculated based on Equation 5.2.

$$mean\_tf\_idf(w, D) = \frac{1}{|D|} \times \sum_{d \in D} tf\_idf(w, d, D). \quad (2.2)$$

All terms with the *mean-tf-idf* score above the average *mean-tf-idf* of all terms are selected to shape the final document-term matrix. Approaches based on term frequency are reported [62, 105] to be as effective as more complicated methods while

having linear time complexity for unsupervised feature selection in textual datasets.

The clustering algorithm gets the *document-term matrix* plus the number of clusters determined by the user as an input. There are different types of interactions between the user and the clustering algorithm (see Table 2.1). We decided to use *seed term* based interaction because it has been reported in several research papers that term labeling is an effective and intuitive way to involve the user in the process of document clustering [13, 73, 60]. It has been reported by Nourashrafeddin et al. that a clustering algorithm could achieve the same performance or even higher by the *seed term* compared to the *seed document* interaction [73]. Unlike the *Split cluster* interaction, the user can split clusters by determining seed terms for each new clusters and help the clustering algorithm decide how to split the clusters. This will reduce the number of interactions between the user and the clustering algorithm. Lexical Double Clustering (LDC) [74] which is designed for *seed term* interaction is selected as a base algorithm in our system. In addition to this, we introduced a framework that applies the user selected *seed term* to extract seed documents for each class of documents (see Section 2.3.2). This framework enables the use of document clustering algorithms with *seed documents* interaction as well.

Outputs of the clustering algorithm are *documents' cluster*, *term cluster relation* (relation of each term to every cluster), and *document cluster relation* (relation of each document to every cluster). The *document-term matrix* and the assigned label of the clusters are used to project the document collection in a 2D space (see Section 2.3.3). Document projection gives a general view of the clustering result and helps the user to interact with the clustering algorithm. The document cluster label assigned by the clustering algorithm is used as a supervision to extract the key-terms for each document cluster. The output of clustering algorithm, document projection, and extracted key-terms are fed to a novel visualization as an interface between the user and the clustering algorithm (see Section 2.3.4). The user can interact with the clustering algorithm in terms of assigning his/her desired terms to each cluster. The user can also remove or create a new cluster based on his/her point of view. The clustering algorithm re-creates new document clustering based on the feedback. This process can be done iteratively until the user is satisfied with the result.

### 2.3.2 Document Clustering

key-term (seed term) based interaction is intuitive for the user and it is been reported that the user can achieve his/her desired clusters faster than by document labeling [73]. We chose Lexical Double Clustering (LDC) [74] which is appropriate for *seed term* based interactions. In order to demonstrate the independence of the proposed system from the clustering algorithm, we introduced an interactive version of KMeans (called *iKMeans*) which is inspired by Basu et al. [11]. The proposed framework for the *iKMeans* can be reused for other clustering algorithms such as *Labeled LDA* [81] to employ them in the proposed system.

#### LDC Algorithm

LDC contains two steps, the first step is term clustering and the second step is using term clusters to create a distilled set of terms to guide the assignment of documents to each term cluster. The *Fuzzy C-means* [16] algorithm is used for term clustering. This algorithm allows a term to belong to more than one cluster (soft clustering). The goal of the *Fuzzy C-means* (FCM) is optimizing the objective function in Equation 2.3.

$$FCM = \sum_{i=1}^{|W|} \sum_{j=1}^k u_{ij}^m \|w_i - c_j\|^2, \quad 1 < m < \infty \quad (2.3)$$

The  $w_i$  ( $w \in W$ :  $W$ =set of terms) is the  $i$ th column of the *document-term matrix* which is a  $|D|$  dimensional vector of *tf-idf* scores. The  $c_j$  is a  $|D|$  dimensional vector of the  $j$ th ( $k$ =number of clusters) cluster center. The *cosine distance* is used to calculate the similarity between each term and term clusters center. The  $u_{ij}$  is degree of membership of term  $w_i$  in the cluster  $j$ . which is defined as Equation 2.4.

$$u_{ij} = \frac{1}{\sum_{k=1}^K \left( \frac{\|d_i - c_j\|}{\|d_i - c_k\|} \right)^{\frac{2}{m-1}}}. \quad (2.4)$$

The membership matrix of  $u$  is recalculated in every iteration of FCM to optimize the objective function shown in Equation 2.3. In the first iteration, all the values in matrix  $u$  are assigned randomly. In case there exist user feedback, the matrix  $u$  will be initialized based on user's feedback instead of random initialization. LDC distills the term clusters and then assigns documents to the closest term cluster after several steps.

---

**Algorithm 1:** *iKMeans* algorithm
 

---

```

input : K=Number of clusters; document-term matrix $^{D \times W}$ ; F=User defined
         seed term; Confidence(%)
output: Doc_clusters

1 if firstIteration then
2   | termClusters  $\leftarrow$  FCM(document-term matrix,K,m=1.1,maxIter=50);
3   | foreach termCluster  $\in$  termClusters do
4   |   | termCluster  $\leftarrow$  getTopTerms(Default=5)
5 else
6   | termClusters  $\leftarrow$  F
7 end
8 termClustersCenter  $\leftarrow$  CC(termClusters);
9 for  $i \leftarrow 1$  to  $K$  do                                     // Expand key-terms
10  | while  $counter_1 < \alpha \times |W| \times 2^{(2 - \frac{Confidence}{25})}$  do
11  |   | termClustersi +=                                       // Cosine Sim.
12  |   |   | nextSimilar(term $\in$ W,termClustersCenteri);
13  |   |   | counter1  $\leftarrow$  counter1 + 1
14 end
15 termToDocCenter  $\leftarrow$  CC(termClustersT);
16 for  $i \leftarrow 1$  to  $K$  do                                     // Find seed docs
17  | while  $counter_2 < \beta \times |D| \times 2^{(2 - \frac{Confidence}{25})}$  do
18  |   | SeedDocsi +=                                       // Cosine Sim.
19  |   |   | nextSimilar(doc $\in$ D,termToDocCenteri);
20  |   |   | counter2  $\leftarrow$  counter2 + 1
21 end
22 SeedDocsCenter  $\leftarrow$  CC(SeedDocs);
23 Doc_clusters  $\leftarrow$  KMeans(document-term matrix,K,SeedDocsCenter);
24 Function CC( $M^{I \times J}$ ):                                       // Calculate Center
25  | for  $i \leftarrow 1$  to  $I$  do
26  |   |  $Center_{M_i} \leftarrow \frac{\sum_{j=1}^{|M_i|} (M_{ij}=[m_1, m_2, \dots, m_n])}{|M_i|}$ 
27  | end

```

---



### ***iKMeans* Algorithm**

In order to demonstrate that our proposed system is independent of the clustering algorithm, we propose an interactive version of KMeans algorithm based on *seed term* interaction. The first step of the framework is *term clustering* (see Algorithm 1). The *Fuzzy C-means* is used to find top terms (the top 5) for each cluster. In term clustering, we cluster the columns of the *document-term matrix*. This step is only for the first iteration of the algorithm; in the next iterations the top terms are determined by the user (line 1-7 of Algorithm 1). Result of our User Study (see Section 2.4.4) demonstrates that the user intends to assign a few terms for each cluster; this is why we only get the top 5 terms for each term cluster. The center of these term clusters is calculated to extend the list of top terms for each cluster (line 8 of Algorithm 1). The center is the average *entrywise* sum of each term vector (column of *document-term matrix*). A term vector is the representation of a term in the vector space defined by the documents. Terms that co-occur often in the same documents will have similar term vectors. As the number of top terms increases, the result of document clustering will be more biased to the top terms indicated by the user. The user can determine his/her confidence percentile in every interaction. The user confidence level regulates the number of top terms to be extended. Based on user confidence, the number of extension of top terms is calculated from the equation in line 10 of Algorithm 1 (Default  $\alpha = 0.2$ ). A term is assigned to the list of top terms of a term cluster if it is more similar to its center according to *Cosine similarity*.

Each term cluster contains a list of terms in which one could imagine that all these terms may belong to a single imaginary document. The average *tf-idf* score of these terms in all documents is considered as the *tf-idf* score of terms in this imaginary document (line 11 of Algorithm 1). This imaginary document is now the center of document clusters. Several documents are assigned to each document cluster center based on Cosine similarity. The number of assigned documents for each document center is related to the user confidence level (line 17 of Algorithm 1). The KMeans algorithm uses these seed documents to initialize the document clusters and then assigns all the remaining documents to each cluster. Based on our experiments in Section 2.4.1, KMeans can produce promising results by having good seed documents.

The proposed framework for *iKMeans* can be reused for other clustering algorithms. To do so, one could only replace lines 2 and 23 of Algorithm 1 with other algorithms. For the first iteration (line 2 of Algorithm 1) which is for finding the top 5 terms for each term cluster, one can use LDA to extract keywords. Instead of KMeans (line 23 of Algorithm 1), the Labeled LDA [81] or any semi-supervised (based on document labels) algorithm can be used as the clustering algorithm. Algorithm 2 is the pseudo code for the general framework. This algorithm is similar to Algorithm 1 with a higher abstraction level. Line 2 and 12 in Algorithm 1 are the ones that user can apply different algorithms.

The independence of the proposed system from the clustering algorithm enables the user to switch between different clustering algorithms on demand.

---

**Algorithm 2:** General Framework

---

```

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
7 calculate the center of the term clusters;
8 expand key-terms for each term cluster based on user confidence;
9 calculate the center of the term clusters;
10 find top (due to user confidence) seed document for each term cluster;
11 calculate the center of the seed documents;
12 Doc_clusters  $\leftarrow$  Use seed documents to guide the clustering algorithm and then
   cluster documents;

```

---

### 2.3.3 Document Projection

Projecting all documents in a 2D space gives the user a global view of the document clusters with their internal and external relations. Principal Component Analysis (PCA) [102] and t-Distributed Stochastic Neighborhood Embedding (t-SNE) [65] are

among the most popular approaches for projection of documents. The t-SNE demonstrate better performance in visualizing clusters of data point than the PCA [65]. The t-SNE and PCA use the bag of words representation of documents to calculate the pairwise similarity between documents. In t-SNE, there is a chance that data points are loosely grouped and consequently, there may be many overlaps between data points from different clusters which makes it difficult for the user to fully understand the structure of clusters. The UTOPIAN [29] and TopicLens [51] systems tried to tackle this problem by multiplying the pairwise distance of data points belonging to the same cluster by a particular factor. In this way, data points belonging to the same cluster are grouped together which results in a clearer view of clusters. The problem with this approach is that by changing the pairwise distance of data points the position of nodes are no longer that match meaningful and the user cannot visually find similar documents belonging to different clusters. In our system, we used t-SNE to project document clusters and combined it with Force-directed placement [37] to illustrate the clusters more distinctly with fewer data points overlap. In the following we briefly describe the structure of Force-directed placement and t-SNE algorithms and then we demonstrate how we combined these two algorithms (extended t-SNE) for better projection of documents.

*Force-directed placement:* The Force-directed placement does not consider the pairwise similarity of data points, instead, it projects them in a way that reduces the number of cross-links (edges) based on different forces. Let  $x_i$  denotes the coordinate of the data point (node)  $i$ , the  $\Gamma(i)$  be the set of neighbors of  $i$ , and  $\|x_i - x_j\|$  the Euclidean distance between two data points. These two nodes are neighbors if there is an edge between them. The *repulsive force*  $f_r$  between two nodes is according to Equation 2.5 and the *attractive force*  $f_a$  between two neighboring nodes is defined as Equation 2.6. The *attractive force* exists only for neighboring nodes. These forces somehow resemble Hookes Law.

$$f_r(i, j) = \frac{-CK^2}{\|x_i - x_j\|}, \quad i \neq j, \quad K = \sqrt{\frac{area}{\#nodes}} \quad (2.5)$$

$$f_a(i, j) = \frac{\|x_i - x_j\|^2}{K}, \quad i, j = neighbor \quad (2.6)$$

In Equations 2.5 and 2.6,  $K$  is a parameter known as *optimal distance* and is calculated based on projection area and the number of nodes [37]. The  $C$  is a constant for

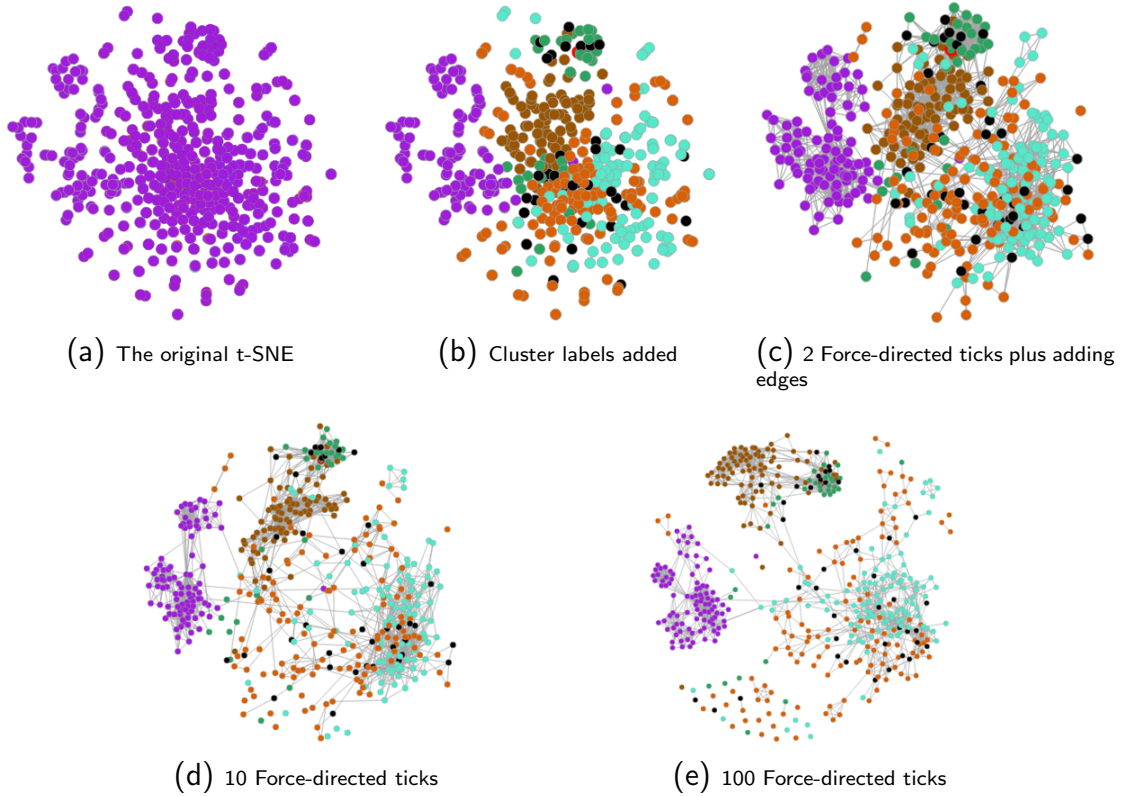


Figure 2.2: The comparative result of t-SNE (a, b) and the t-SNE combined with Force-directed (c, d, e). Cluster labels are based on clustering algorithm. Even with applying a few Force-directed iterations a clearer projection of documents is producing (d). If the number of Force-directed iterations increases, the result of t-SNE will be similar to the Force-directed layout (e). A subset of 490 randomly chosen documents of *BBC sport* data set [39] is used in these figures.

regulating repulsive and attractive forces between nodes. Equation 2.7 shows the combined repulsive and attractive forces of the node  $i$  [46].

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

The overall energy of all nodes is defined as Equation 2.8. The Force-directed placement tries to minimize the total energy of nodes. The forces can be minimized iteratively by moving nodes according to their forces' direction (each iteration is called a *tick*). In the first iteration, all nodes are placed randomly in the area. The *BarnesHut*

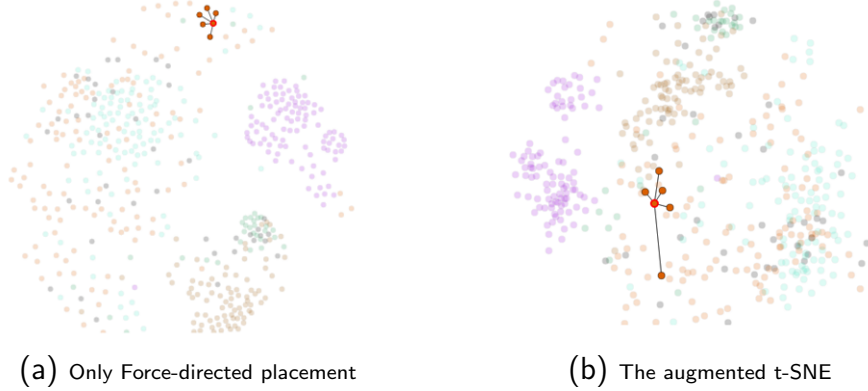


Figure 2.3: The user has an access to both augmented t-SNE (b) and only Force-directed placement (a) layouts. If the user clicks on a node in one of these layouts, that node will be highlighted in another layout as well.

approximation [10] is usually applied to make the optimization of forces faster.

$$Energy(x, K, C) = \sum_{i \in Vertices} f^2(i, x, K, C). \quad (2.8)$$

*t-SNE*: The t-SNE minimizes the divergence between the distribution of input nodes and the distribution of corresponding nodes in the low-dimensional (typically 2 or 3) space according to their pairwise similarity. Let  $I = \{x_1, x_2, \dots, x_{|D|}\}$  be the set of input nodes and  $|D|$  the number of documents in the *document-term matrix*. Each node in  $I$  has the dimension equal to the vocabulary size of the document collection. The node  $x_j$  is similar to node  $x_i$  based on the conditional probability described in Equation 2.9 [99].

$$p(j|i) = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)}, \quad p(i|i) = 0 \quad (2.9)$$

In Equation 2.9, the  $\sigma_i$  is the bandwidth of the Gaussian kernels and is set based on the user predefined perplexity. Let  $O = \{y_1, y_2, \dots, y_{|D|}\}$  be the set of counterparts of nodes in  $I$  in the corresponding low denominational space. Thus, the similarity between node  $y_j$  and  $y_i$  is computed as Equation 2.10.

$$q(j|i) = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k \neq i} \exp(-\|y_i - y_k\|^2)}, \quad q(i|i) = 0 \quad (2.10)$$

The location of data points (nodes) is determined as a result of minimization of the joint distribution of  $P$  and  $Q$ . This task is iteratively done by optimizing the Kullback-Leibler divergence (Equation 2.11). Similar to the Force-directed placement, nodes

are attractive or repulsive to each other based on their  $p$  and  $q$  probabilities in each iteration.

$$KL(P \parallel Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}. \quad (2.11)$$

*Extended t-SNE*: Projecting documents with only t-SNE forces may lead to an overlapped view of documents (data points). In order to improve the result of the t-SNE based projection, we combined Force-directed placement forces with t-SNE ones. First, only t-SNE forces are applied to nodes until they get balanced, then based on the current location of nodes, Force-directed placement forces will be applied to nodes. Let  $iter$  be the number of iterations that Equation 2.11 gets optimized, and  $[u \geq iter]$  be a function to be 0 when  $u < iter$  and 1 otherwise then data points (documents) project based on Equation 2.12. The  $\lambda$  is a regulatory constant to control the impact of Force-directed placement on data points.

$$Pr(P, Q, x, K, C) = KL(P \parallel Q) + \lambda [u \geq iter] Energy(x, K, C) \quad (2.12)$$

The  $Energy(x, K, C)$  function of Equation 2.12 will be applied iteratively to nodes. A few iteration of it can significantly improve the projection of data points located by t-SNE. Fig. 2.2a demonstrates the projection of 489 documents belonging to *BBC sport* dataset [39] based on original t-SNE. After assigning the nodes label by the clustering algorithm it is clearer that some nodes are overlapped (Fig. 2.2b). After certain iterations of t-SNE, the forces between nodes will be changed to the Forced-directed ones (Figs. 2.2c, 2.2e, 2.2d). As the goal of Force-directed placement is reducing the number of cross-links, before applying the Force-directed forces, we add links to nodes. The links are according to the cosine similarity between the bag of words vector representation of documents (nodes). The two nodes will have a link if they are similar due to a certain threshold. Even applying a few Force-directed forces the overlaps between nodes decreases significantly while preserving the meaningful (based on t-SNE) locations of nodes (Fig. 2.2d). If the number of Force-directed iterations increases, the result of t-SNE will be similar to the original Force-directed layout (Fig. 2.2e).

Projecting documents based on Forced-directed approach will place documents with similar cluster labels together while placing the isolated nodes far from the

center (Fig. 2.2e). The problem with Forced-directed placement is that the location of nodes is not according to their pairwise similarity (contrary to t-SNE). This means that two nodes placed near each other are not necessarily similar to each other. In addition to the augmented t-SNE with the Force-directed projection (Fig. 2.3b) of documents, we also provide the user the projection of documents only with the Force-directed layout (Fig. 2.3b). If the user clicks on a node in one of these layouts, that node will be highlighted in another layout as well (Fig. 2.3).

### 2.3.4 Visual Components

The overall picture of the proposed web-based user-centered system is depicted in Fig. 2.4. The system contains several components called *view*. The name of each component is indicated in its header part. The size and the location of each *view* are designed as a result of our user study due to their importance and usage frequency (see Section 2.4.4). The *Graph view* is the most frequently used component, and *views* on its right part are more frequently used than the ones on the left. The user has the freedom of resizing and relocating every component of the system. The components are interconnected to each other and changing a parameter in one could impact the other components as well. In Fig. 2.4 the components with the similar colorful header are sharing information about the selected cluster (the one with red color margin) in *Cluster view*. As depicted in Fig. 2.1, the user first determines the number of clusters then the system clusters the document collection and visualizes the result in different *views* (Fig. 2.3). The user gains benefit from *views* to get insights into the document collection to provide feedback to the clustering algorithm. The feedback is in terms of adding, removing, and reordering of clusters' top terms or adding a new cluster or removing a cluster (which change the number of clusters) in *Cluster view*. The user can iteratively interact with the clustering algorithm several times and in each iteration can ask a different clustering algorithm to cluster documents. The clustering result can be saved in each iteration so the user can rollback the previous results of clustering.

The document's glyph in *Graph view* (Fig. 2.5) contains information such as document cluster(s) color, document's name, and a list of top terms based on its terms *tf-idf* score. The user can find the location of the selected document in the

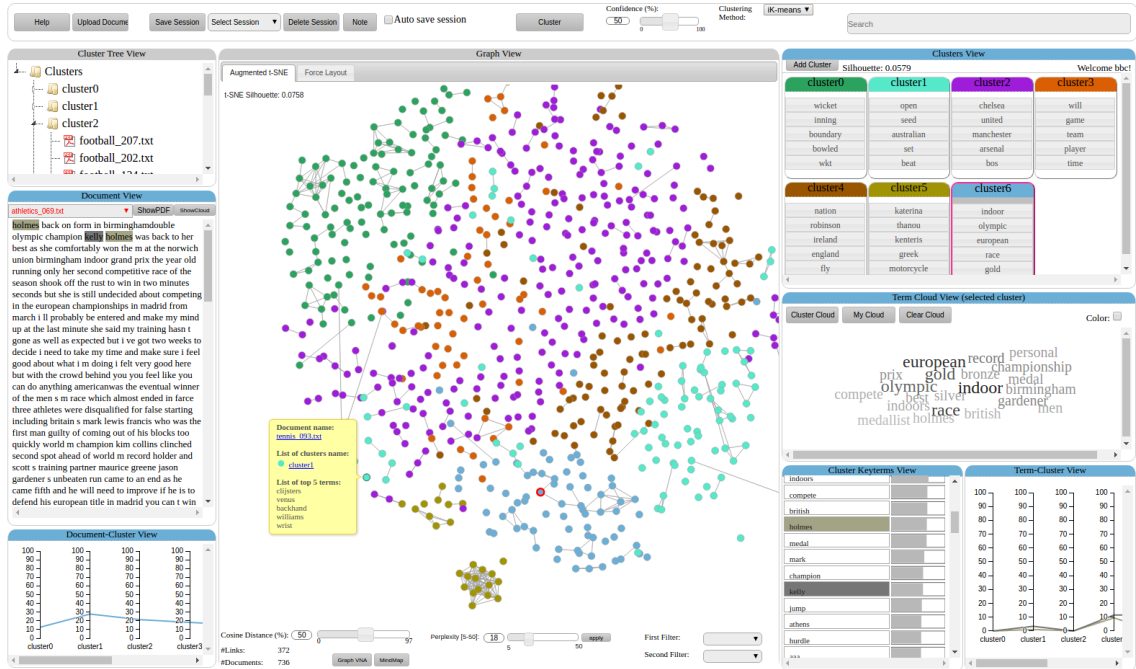


Figure 2.4: The visual interface of the proposed system. In the middle, the projection of 737 documents of the *BBC Sport* dataset is depicted (*Graph view*). On the left, we see the *Cluster tree view* for a hierarchical display of clusters and documents, the *Document view* for showing the plain text of documents, and the *Document-cluster view* to depict the relatedness of the selected document to each cluster. The name of each visual component is given in its header. On the right, we see the *Clusters view*, which demonstrates top terms of clusters, the *Term cloud view* for highlighting top terms of a selected cluster or set of documents, the *Cluster key-terms view* for listing top terms of a selected cluster with their level of importance (bar charts), and beside it the *Term-cluster view* to depict relatedness of a selected term(s) in *Cluster key-terms view* to each cluster. The views with colored header are all related to the selected cluster in the *Clusters view*. The selected cluster has a red margin and the same header color. The user can add, remove, rename or recolor a cluster or merge two clusters in *Clusters view*. The feedback to the clustering process by changing the number of clusters or adding/removing terms in *Clusters view*, as well as adding or removing cluster. The user can send changes made by pressing the *cluster* button on the top of the *Graph view*.

*Document view* by the red color stroke of a glyph, or load the textual content of a node in *Document view* by clicking on the name of a cluster in node's tool-tip. If a document belongs to more than one cluster, its inner color turns to black. This is more convenient for the user to spot multi-clustered documents than using *Pie Visualization* when there are many documents. The user can zoom in/out the *Graph view* to have a better view of documents. The glyphs are connected according to



Cosine similarity. If the user change the threshold, links will be added/removed from graph view in real time. By setting the Cosine distance threshold to zero, all duplicated (repetitive) documents will still have a link. Two documents are repetitive if they share exactly the same textual content with different file names. If the user clicks on a glyph (see Fig. 2.3), that glyph and its neighbors will be highlighted in both augmented t-SNE and Force-directed layouts (the second tab in *Graph view*). It is possible to add more nodes to the selected nodes (*keep function*) or remove a node from the selected nodes (*un-keep function*). The user can get the summary of selected nodes in *Term cloud view*. Let  $G$  be a set of selected glyphs (documents), and  $M^{|G| \times |W|}$  a subset of *document-term matrix* containing selected documents. The score (its level of importance) of a term  $t$  belonging to the selected documents is defined as Equation 2.13.

$$Score(t_i) = \frac{1}{|G|} \sum_{j \in G} M_{ji} \quad (2.13)$$

In *Term cloud view*, a term with the higher score has a larger font size with a darker color. The single color term representation helps the user to spot the important terms faster than the colorful version of it. It is possible to switch between single color to the colorful version in case of better spotting of *bigram* terms.

Selecting terms in *Cluster key-term view*, highlights all documents (nodes) containing those terms in *Graph view*, and *Document view*. This will help the user to find the discriminative power of selected terms which is somehow similar to visualizing the *tf-idf* score of selected terms. We chose the grayscale color for highlighting the selected terms in *Cluster key-term view*, and *Document view* to differentiate them from the clusters' color. The user can search a term in the document collection and directly add it to the list of top terms of a cluster. The documents relatedness in *Document-cluster view* and term relatedness in *Term-cluster view* are calculated by the Chi-squared statistic ( $\chi^2$ ) using assigned clusters labeled by the clustering algorithm. Chi-squared statistic is a supervised feature selection algorithm.

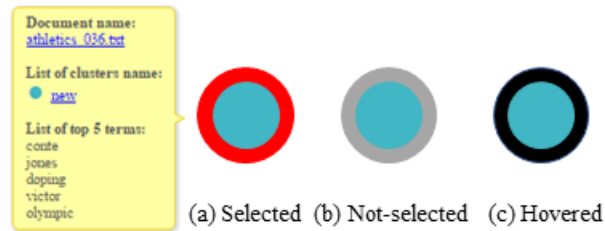


Figure 2.5: The Document’s glyph in *Graph view*. The inner circle indicates document cluster color and the outer one demonstrates its selection status. The left glyph with red color stroke is the selected document in *Document view*. Selecting a node in *Graph view*, turns the node stroke color from gray (the middle glyph) to black (the right glyph). The user can see the node’s document name, cluster name, and a list of top terms of each node by hovering the mouse on a glyph.

## 2.4 Experiments

In this section, a quantitative evaluation, a usage scenario, a case study and a user study are reported to examine the quality and effectiveness of the proposed system. The proposed web-based system is implemented in Python in the back-end and Javascript, jQuery, HTML, and D3 [19] in the front-end. The result of document clustering can be saved as a Zip file of clustered documents, MindMap, or VNA graph format. The following datasets are used in our experiments.

- *BBC Sport*: This dataset was introduced by Greene and Cunningham [39] which contains 737 news articles about 5 sport categories extracted from BBC website from 2004-2005.
- *Yahoo Answers*: A collection of 189,467 questions and answers extracted from Yahoo! answers website with 20 top-level and 280 sub-categories [28]. In the user study (Section 2.4.4), we used 6 sub-categories containing general questions about *Computer*, *Education*, *Music*, *Food Receipts*, *General Health*, and *Society*. For each category we randomly selected 100 question and answer pairs.
- *R8*: A subset of *Reuters-21578* dataset containing 8 categories with 2,189 test and 5,485 training documents (7,674 documents together) [25]. *Reuters-21578* is a popular dataset for text classification with 21,578 documents extracted from Reuters newswire in 1987 then assembled by David Lewis [86].

- *NewsSeparate*: This dataset contains 381 news feeds manually categorized into 13 categories [77]. In the user study (Section 2.4.4), we randomly selected 100 documents belonging to 4 categories from this dataset.
- *WebKB*: This dataset is 4199 faculty, student, project, and course web sites crawled from the computer science faculty website of four universities in January 1997 [31].
- *NG5*: This dataset is a subset of 20 *Newsgroups*<sup>2</sup> dataset including 5 categories with 80 randomly chosen documents for each category. *Newsgroups* dataset consists of nearly 20,000 messages of Internet news articles with 20 categories collected by Ken Lang [55].

#### 2.4.1 Quantitative Evaluation

The impact of using *unigram* and *bigram*, as the bag of words representation of documents, on the clustering algorithm are presented in Table 2.3. Based on different evaluation metrics, the combination of *unigram* and *bigram* can have some negative impacts on the quality of the clustering algorithm but users may find more meaningful phrases by having *bigrams*. The default configuration is only *unigrams* unless the user decides to have *bigrams* as well. In this experiment, we selected *Fuzzy C-means* algorithm as clustering algorithm and reported the average result of 20 runs. The feature selection is based on *mean-tf-idf* method. The experiment is evaluated by *Adjusted Random Score* (ARS), *Adjusted Mutual Information* (AMI), *Homogeneity* (H), and *Average Silhouette* (S) (the average of all documents' *Silhouette* score). ARS is a measure of similarity between the ground truth clustering and the clustering algorithm [49]. The ARS value near 0 indicates the random labeling. ARS is penalized by the number of false positive and false negative predictions. *Mutual Information* is a measure for calculating the amount of mutual information between predicted labels and the actual labels. AMI is an adjustment of the *Mutual Information* to reduce the effect of the agreement by chance and it is 0 in random labeling [100]. A clustering result has a higher *Homogeneity* score (between 0 and 1) if its clusters contain more documents which are members of a single class [87]. *Silhouette* is an

---

<sup>2</sup>[www.qwone.com/~jason/20Newsgroups/](http://www.qwone.com/~jason/20Newsgroups/)

unsupervised metric for evaluation of the document clustering algorithm and helping the user to find the optimum number of clusters [88]. The *Silhouette* is between  $-1$  and  $1$  in which the higher positive value for a document shows it is more similar to the assigned cluster than the other clusters.

We compared different clustering algorithms with our proposed framework without the user interactions in Table 2.4. The experiments are as a result of using *unigrams* with *mean-tf-idf* as feature selection, and algorithms are initialized randomly. First, *iKMeans* (Algorithm 1) receives the top 5 terms for each cluster from the output of the *Fuzzy C-means* algorithm, then it provides seed documents for the *KMeans* algorithm. In all of the datasets *iKMeans* performed better than *KMeans* which demonstrates the effectiveness of our proposed framework. Comparing *iKMeans* with *LDC* and *Fuzzy C-means* algorithms shows that there is not any clear winner and some approaches performed better in some datasets but not in all.

In the next experiment, the interactivity feature of the *KMeans* algorithm which is called *Seeded KMeans* in Table 2.5 is evaluated. In this experiment, the documents' label are provided as supervision to the algorithm so instead of random initialization it has been initialized based on these documents. This is somehow simulating lines 1-21 of Algorithm 1 by directly providing the seed documents to the clustering algorithm (*KMeans*). We provided four different sets of seed documents.

- *Bad Seeds*: In the case of *Bad Seeds*, 3 seed documents are assigned to each cluster where all of these seed documents have a similar label.
- *Semi Bad Seeds*: If the number of seed documents per cluster is reduced to 1, *Semi Bad Seeds* situation will occur.
- *Good Seeds*: For clustering with *Good Seeds*, 5 seed documents are assigned to each cluster, where each cluster's seeds are labeled to a single correct label different from other clusters' seeds.
- *Semi Good Seeds*: Reducing the number of seed documents to 1 results in a *Semi Good Seeds* setting.

The results in Table 2.5 demonstrates that by providing a few good labeled documents *KMeans* performance increases significantly. On the other hand, the *Fuzzy*

*C-means* performance did not change with this number of documents and more seeds are needed to impact the result of it. From the user’s point of view, the algorithm which could be improved with a fewer number of interactions is more favorable.

The impact of user confidence (lines 10 and 17 of Algorithm 1) on the result of clustering is depicted in Fig. 2.6. The higher the confidence the lesser is the expansion of user’s terms which result in a cluster containing documents having only the user’s terms (Fig. 2.6b). With lower confidence level the user’s terms will be expanded and consequently, the resulting cluster will have more documents.

Table 2.3: The impact of *bigram* and *unigram* on the clustering result (The average 20 runs of *Fuzzy C-means* algorithm)

Dataset Name	Evaluation metric	Unigram	Unigram & Bigram
R8 (7674 doc. 8 classes)	Adjusted Random Score	0.290	0.233
	Adjusted Mutual Info	0.397	0.371
	Homogeneity	0.585	0.540
	Average Silhouette	0.089	0.058
WebKB (4199 doc. 4 classes)	Adjusted Random Score	0.302	0.296
	Adjusted Mutual Info	0.354	0.341
	Homogeneity	0.368	0.356
	Average Silhouette	0.025	0.014

Table 2.4: The comparison result of clustering algorithms with random initialization (The average 200 runs for each algorithm).

Dataset Name	Metric	Fuzzy C-means	LDC	KMeans	iKMeans
NG5 (400 doc. 4 classes)	ARS	0.521	0.501	0.201	0.628
	AMI	0.614	0.577	0.297	0.710
	H	0.619	0.582	0.306	0.714
	S	0.090	0.074	0.034	0.087
R8 (2189 doc. 8 classes)	ARS	0.353	0.440	0.195	0.305
	AMI	0.438	0.481	0.387	0.447
	H	0.637	0.659	0.473	0.592
	S	0.092	0.083	0.077	0.092
WebKB (4199 doc. 4 classes)	ARS	0.310	0.324	0.265	0.320
	AMI	0.337	0.326	0.281	0.311
	H	0.354	0.337	0.283	0.314
	S	0.027	0.025	0.024	0.022

Table 2.5: The role of seeds (documents) quality in the result of document clustering. The *NG5* dataset is used in this experiment.

Seed Type	Evaluation metric	Fuzzy C-means	Seeded KMeans
Bad seeds	Adjusted Random Score	0.558	0.006
	Adjusted Mutual Info	0.647	0.006
	Homogeneity	0.651	0.051
	Average Silhouette	0.104	0.010
Semi-bad seeds	Adjusted Random Score	0.486	0.067
	Adjusted Mutual Info	0.588	0.198
	Homogeneity	0.593	0.208
Semi-good seeds	Average Silhouette	0.102	0.052
	Adjusted Random Score	0.548	0.305
	Adjusted Mutual Info	0.641	0.414
Good seeds	Homogeneity	0.646	0.422
	Average Silhouette	0.103	0.048
	Adjusted Random Score	0.545	0.637
Good seeds	Adjusted Mutual Info	0.639	0.711
	Homogeneity	0.644	0.715
	Average Silhouette	0.103	0.088

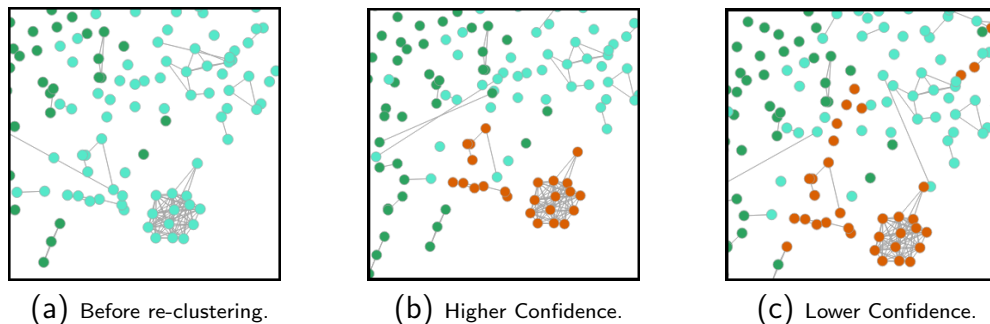


Figure 2.6: The impact of the confidence level on the clustering result. The *BBC Sport* dataset before (a) and after (b, c) sending re-clustering signal.

#### 2.4.2 Usage Scenario

The scenario presented here demonstrates how a user can change the topics and number of clusters in order to reflect her preferences in the process. The author of this thesis perform the usage scenario. Although the framework is very flexible and different users may choose to operate by employing different modules, we follow one possible set of steps to illustrate the process and the support our framework offers.

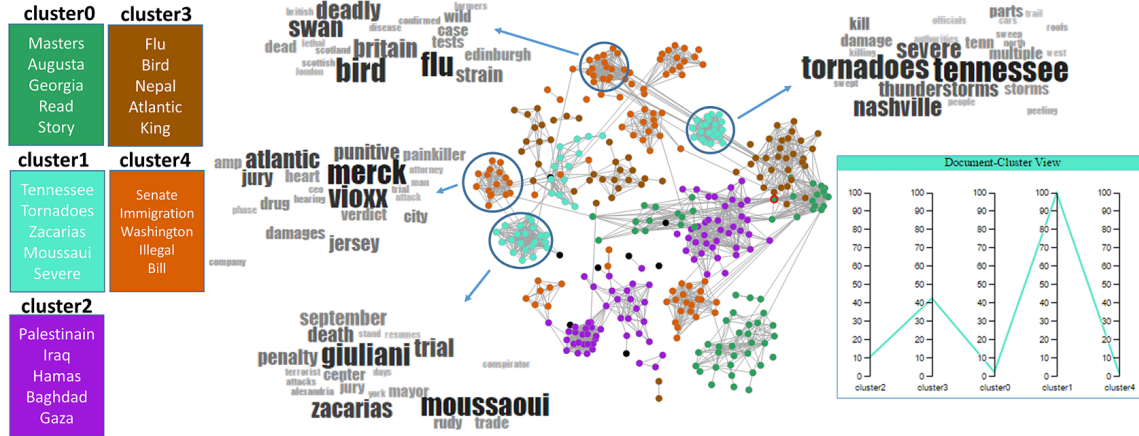


Figure 2.7: Use case 1: clustering *NewsSeparate* - Iteration 1: Term clouds of different subgraphs of “cluster1” and “cluster4” show that these clusters must be separated to obtain finer sub-clusters. Top five key-terms of each cluster are shown in the rectangles. Document-Cluster View shows that a single document belongs to “cluster1” and “cluster3” with different degrees of relevance.

To this end, we have employed *NewsSeparate* which contains news articles in 13 topics that can be identified as “Trial of Zacarias Moussaoui”, “Viondrug dispute”, “Tennessee Tornadoes”, “Stock Price”, “Iraq Suicide”, “Nepal Strikes”, “Masters Augusta (Golf)”, “Katie Courie”, “Immigration Bill”, “Hamas”, “Enron Insurance case”, “Bird Flu”, and “Da Vinci Code dispute”.

Table 2.6: The Silhouette indices of clusterings generated on *NewsSeparate* in each iteration. The Silhouette indices of t-SNE and Force Layout are independent of the user interaction. The Silhouette index of the clustering shows the quality of clustering.

	Average Silhouette			
	Iteration1	Iteration2	Iteration3	Iteration4
Clustering	0.0830	0.1718	0.2011	0.2224
t-SNE	0.0522	0.2305	0.3530	0.3385
Force Layout	0.0474	0.2423	0.3774	0.4681

Even though we know the number of classes, the initial clustering was generated with five clusters. The rectangles of the clusters, the graph representation of the clustering, and a term cloud of each subgraph are depicted in Fig. 2.7. Based on the key-terms of the rectangles, one can observe that the topics of clusters were “Golf”, “Tennessee Tornadoes”, “Hamas and Palestine”, “Bird Flu”, and “Immigration Bill”. The average Silhouette indices of this iteration are shown in Table 2.6.

The term clouds of the subgraphs clearly demonstrate that multiple disjoint groups of documents (based on similarity) were assigned to the same cluster. For instance, the term clouds of two subgraphs of “cluster1” correspond to the documents of “Trial of Zacarias Moussaoui” and “Tennessee Tornado”. This observation clearly implies that the number of clusters should increase.

We observed that a few documents were assigned to more than one cluster (black nodes in the graph). A part of these documents contain discussions about the economic aspect of bird flu and were placed into both “Stock Price” and “Bird Flu” clusters. The other documents cover two topics of “Tennessee Tornadoes” and “Bird Flu”.

After analyzing the results, we decided to add five new clusters for the next iteration. The topics of the new clusters were specified by using terms in the term clouds, term lists, and document contents. A re-clustering signal along with the term lists were sent to the clustering algorithm.

It is logical that the topics of document clusters alter after each iteration; for instance, “cluster1” in the following iterations will not represent the same topic as in the current iteration.

The results of the second iteration are depicted in Fig. 2.8. Documents were grouped into ten clusters and improvement was achieved based on the Silhouette index [Table 2.6]. The improvement confirms that increasing the number of clusters was successful. Nevertheless, there were still different groups of documents placed in the same cluster. We thus added two new clusters targeting at segregating these topics.

The output of the third iteration is shown in Fig. 2.9. We observed that some documents were mistakenly placed in cluster “Da Vinci Code dispute” while they actually belong to “Stock Price”. These topics share key-terms such as “London”, “legal”, and “expected”. We thus removed these common key-terms from the rectangles for the next iteration. The alternative was to keep them in the rectangles with different levels of relevance but we did not have enough knowledge about the collection to do so.

For the fourth iteration, we added two new clusters [Fig. 2.10] and we decided to stop at this point.



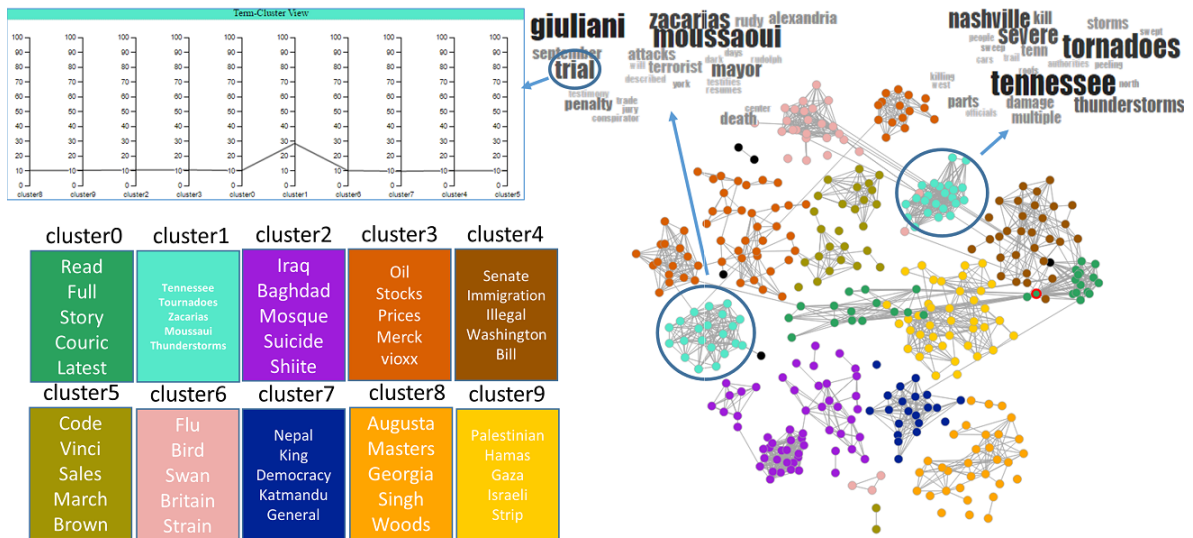


Figure 2.8: Use case 1: clustering NewsSeparate - Iteration 2: Except for “cluster1”, most clusters contain only one group of adjacent documents. There are two groups of documents in “cluster1”. The topic of one group is “Trial of Zacarias Moussaoui” and the topic of the other one is “Tennessee Tornadoes”. Based on the term clouds, there are some common terms like “death” in both groups. The Term-Cluster View shows the relevance of term “trial” in document clusters.

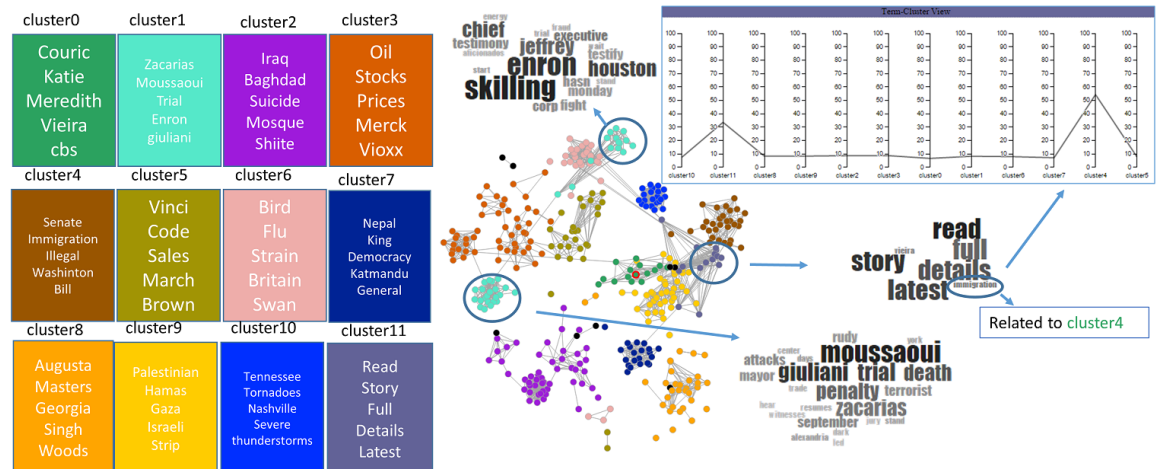


Figure 2.9: Use case 1: clustering NewsSeparate - Iteration 3: The term cloud of “cluster1” shows that a new cluster is needed to separate topics of “Enron Insurance case” and “Trial of Zacarias Moussaoui”. The Term-Cluster View of “immigration” shows that it is more related to “cluster4” than to “cluster11”.

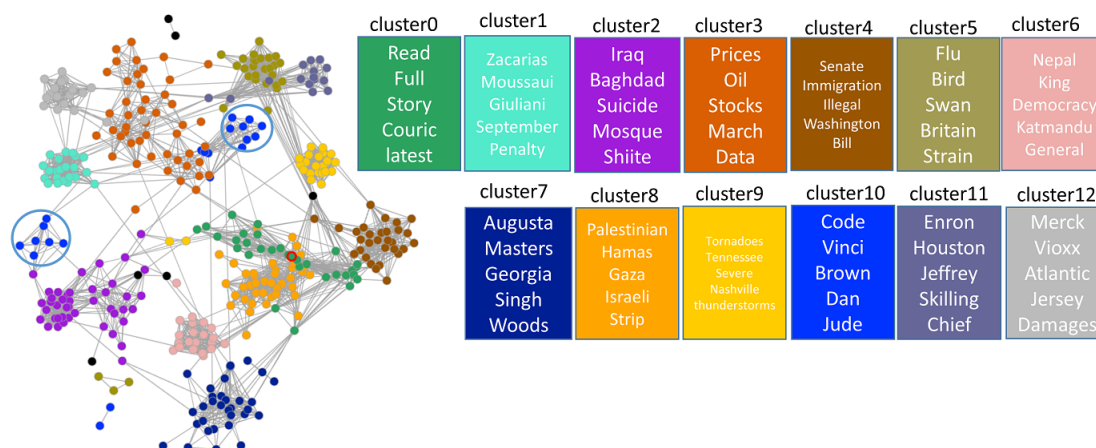


Figure 2.10: Use case 1: clustering *NewsSeparate* - Iteration 4: Result of clustering *NewsSeparate* after four iterations. Most groups of adjacent nodes of the graph are in distinct clusters. Some clusters, however, could be distilled further, for instance “cluster10”.

This use case shows the efficacy of the framework to generate user-desired clusterings by increasing the number of clusters and specifying the topics of interest. The same scenario is applicable to form the clusters in a top-down approach by merging clusters and reducing the number of clusters.

### 2.4.3 Use Case

We asked a computer science researcher with good knowledge about document clustering to cluster the *Yahoo Answers* dataset without mentioning the subject of the dataset to her. The use case took 40 minutes including 20 minutes for system introduction. We recorded the user interactions and have reported the summary of them in this paragraph. First, the user asked the clustering algorithm to generate 4 clusters which Fig. 2.11a demonstrates the initial result of the clustering. Second, the user checked the *Term Cloud* of clusters to know their subject and by looking at the *Graph view*, she noticed that there were two dense groups of documents in the cluster with the topics *Cancer*, *Syndrome*, and *Lymphoma*. Third, the user selected several nodes related to one of these groups of documents in the *Graph view* with the help of the *Keep function*. She noticed that these nodes were about *education* and *university* (The blue *Term Cloud* in Fig. 2.11a), so she selected the top three terms from *Term Cloud* and created the new cluster. The new clustering result after adding

a new cluster is depicted in Fig. 2.11b. The new cluster (blue color) now had two dense sets of nodes. Fourth, the user found out that terms such as *Education* were common between these two sets of documents by the help of *Term-cluster view*. On the other hand, one of the sets of documents was more about *Language, French, and Spanish* so she decided to create a new cluster. The user also removed the term *Education* from the blue color cluster to help the clustering algorithm to separate these two sets of documents. The result of clustering after the third interaction of the user is in Fig. 2.11c. The improvement in the Silhouette score after each interaction (see Table 2.7), gave the user more confidence about the usefulness of the user feedback to the clustering algorithm. The Silhouette score of the t-SNE is calculated based on the x and y coordinate of each document in the 2D space while for the Clustering is based on the bag of word representation of each document. Because of that, it is not meaningful to compare the Silhouette score of the t-SNE with the Clustering.

Table 2.7: The improvement of the Avg. Silhouette after each interaction.

Avg. Silhouette	Interaction 1	Interaction 2	Interaction 3
t-SNE	0.3493	0.3533	0.3544
Clustering	0.1267	0.1382	0.1530

#### 2.4.4 User Study

We invited 18 participants (9 male, 9 female) for a user study. The participants were computer science students with at least acceptable knowledge of document clustering and strong English comprehension skills. The study was in an office with a single monitor with 1920x1200 resolution. In this study, we had two research questions: 1) evaluate the impact of users' interactions on the quality of document clustering. 2) study if the visualization assists the users in obtaining better insight into the document collection and to improve the clustering result. To find the answer to these questions we designed two separate tasks. For each user, 20 minutes of training and 50 minutes for two tasks of the study was provided.

In the first task, we gave each of the 18 participants 30 minutes to cluster the *Yahoo Answers* dataset. Each document in this dataset was renamed to prevent the user from finding the correct cluster label of documents by their name. We did not

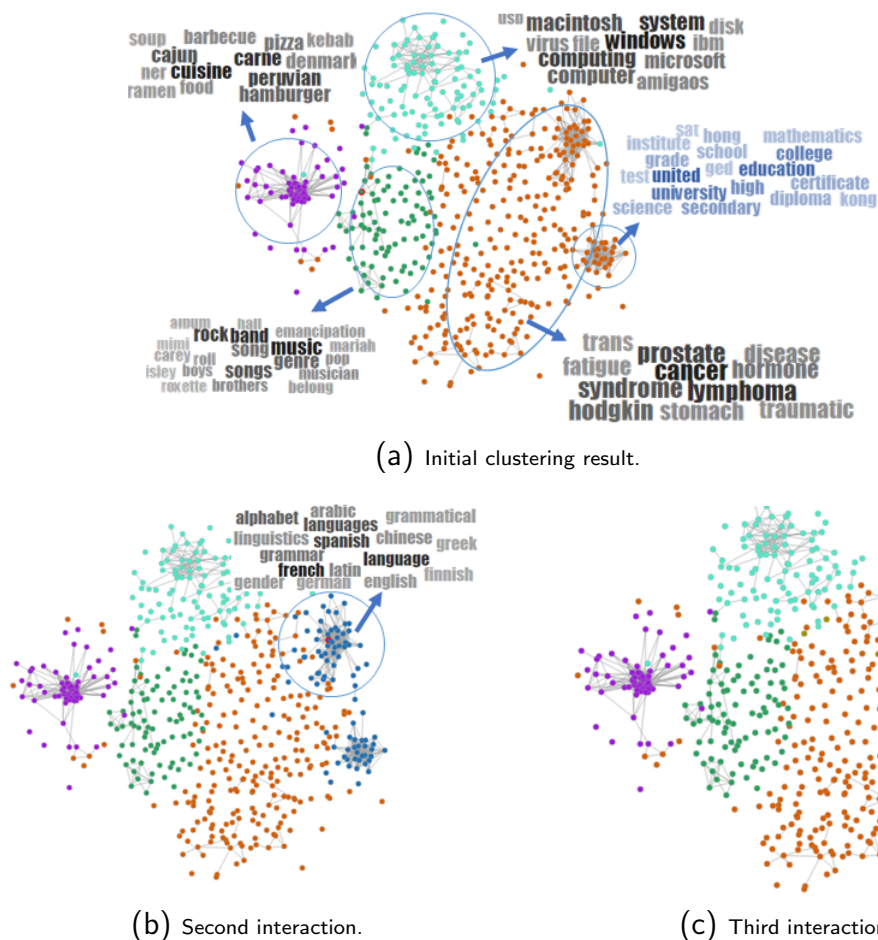


Figure 2.11: The screen-shots of different interaction rounds conducted by the user. The *Yahoo Answers* dataset and the *iKMeans* algorithm is used in this use case.

provide any information about the topics of the dataset and its number of clusters. To provide similar condition for each user, each one started with the result of random initialization of the clustering algorithm with 3 clusters. We recorded every operation that users conducted during this task. The average frequency of five important actions of the term-based interactive clustering that users conducted is in the middle section of Table 2.8. The following items are the summary of findings in the first task.

- The high standard deviation for the number of *Add term*, indicates that some users generated a longer list of key terms as the supervision to the clustering algorithm. There is a +0.42 (Pearson) correlation between the number of *Add term* and the *Homogeneity* score of the final clustering result after users' interactions.

Table 2.8: Statistics about the study. The left table demonstrates users’ vote for visualization modules. The middle table is average number of important users’ interactions in the first task. The right table highlights the most frequent users’ operations and the name of its parent module.

Module name	# Votes	Action name	Avg. Count	Operation description	Module name	Percent
Document-cluster & Document view	4	Re-clustering	$7.17 \pm 4.78$	Highlight documents containing the selected term	Graph view	11.45%
key-terms view & Term-cluster view	6	Add term	$25.39 \pm 26.42$	Click on a cluster to load its information on views	Cluster view	10.95%
Cluster view	5	Remove term	$5.61 \pm 5.77$	Mouse over a node to see its information (tool-tip)	Graph view	7.56%
Term cloud view	6	Add cluster	$4.22 \pm 2.05$	Creating terms cloud	Term Cloud view	6.86%
Graph view	14	Remove cluster	$0.33 \pm 0.77$	Click term in key-term view & load Term-cluster view	key-terms view & Term-cluster view	6.69%

- The users categorized this dataset to  $5.7 \pm 1.3$  clusters in average which is close to the actual number of classes in this dataset.
- The most frequent operations users did during this task is summarized in the right section of Table 2.8.
- The users were asked in the post-task questionnaires to vote for each module of the system (multiple votes were allowed). The approval of the popularity and the importance of the *Graph view* by the users is demonstrated in left part of Table 2.8.

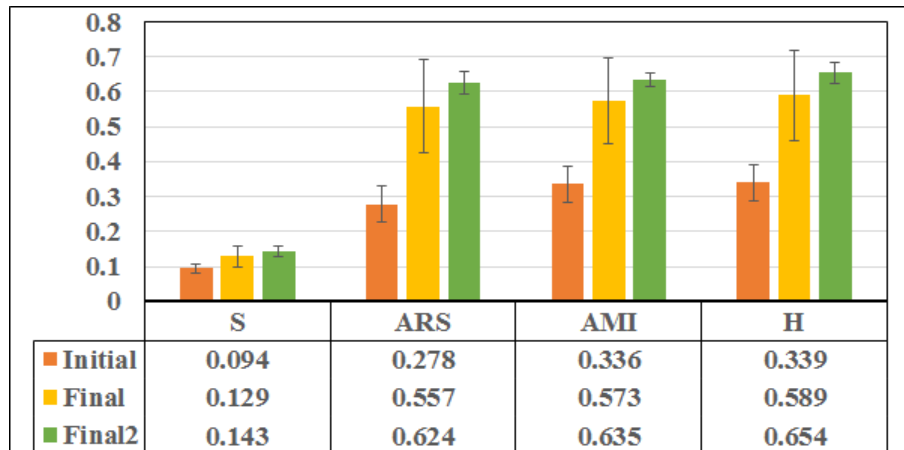


Figure 2.12: The comparison between the initial clustering and the final clustering after user interactions. The final2 is after removing outliers.

In order to investigate the impact of users’ interactions on the quality of the final clustering result, we compared the initial result of the clustering and the result of the

clustering after user interactions in the first task. The evaluation metrics are *Adjusted Random Score* (ARS), *Adjusted Mutual Information* (AMI), *Homogeneity* (H), and *Average Silhouette* (S) (See Fig. 2.12). Results demonstrate that user supervision significantly improved the result of clustering. During the study, four participants misunderstood the instructions and were not able to complete the task properly. A higher performance with less Standard Deviation was achieved after removing these participants (Final2 bar charts in Fig. 2.12).

Table 2.9: Comparing the result (number of correct answers) of the system with (Vis.) and without (Base) visualization and their significance differences based on Wilcoxon signed-rank test with  $p < 0.05$ .

Question	Base	Vis.	P-value
1-Provide a title for each cluster	14/18	16/18	0.2119
2-Give the names of two clusters with most similar topics	4/18	9/18	0.0184
3-For each cluster, provide at least 1 term belong to other clusters as well	2/18	9/18	0.0054
<b>The overall (for all questions)</b>	<b>20</b>	<b>34</b>	<b>0.0023</b>

In the second task, we studied the impact of the user interface on giving the user better insight into the document collection. The *NewsSeparate* dataset with 3 initial clusters is elaborated in this task. The users were unfamiliar with the content of the dataset prior to the experiments. A questionnaire was provided to each user, who had 10 minutes to complete it (See Table 2.9). We asked the user to answer these questions twice (two modes); once with and once without (base mode) the visualization. In the without visualization mode, the ordered list of the top terms of each cluster and the folder of documents related to each cluster was provided. The users were able to use the Windows file explorer feature and the Notepad++ application to dig into the clustering result in this mode. These are the usual tools that users use when using an automatic clustering algorithm without the visualization. We did not inform the users that in both modes they are answering the same questions about the same dataset. We randomly divided the participants into two halves. For the first half the clustering results with and for the second half without visualization was provided first. For the first question, the users were able to answer this questions properly in both modes. We believe the reason is that the ordered list of the top terms for each cluster is a

good description of each cluster and it was easily accessible in both modes. For the next two questions, the result of the visualization mode is statistically significantly better which shows the effectiveness of the visualization. The order of providing the visualization or the base mode at first did not have a statistically significant impact on the quality of users' answers. The overall comparison result in Table 2.9 demonstrates that the visualization is significantly better than the base mode.

Table 2.10: The Post-task questionnaire. Questions 1-10 are from Software Usability Scale (SUS) questionnaire [21]. All questions are in 5-point Likert scale agreement scores (the higher, the more agreement).

Question	Avg.
1-I think that I would like to use this system frequently	4.28±0.83
2-I found the system unnecessarily complex	2.28±0.83
3-I thought the system was easy to use	4.22±0.65
4-I think that I would need the support of a technical person to be able to use this system	2.22±1.06
5-I found the various functions in this system were well integrated	4.61±0.50
6-I found there was too much inconsistency in this system	1.44±0.62
7-I would imagine that most people would learn to use this system very quickly	4.17±0.92
8-I found the system very cumbersome to use	2.17±1.04
9-I felt very confident using the system	4.28±0.75
10-I needed to learn a lot of things before I could get going with this system	2.28±1.13
11-It is more meaningful to use phrases instead of single words to determining the clusters topics	3.89±0.83
12-Term based visualization and term labeling is a useful way in generating desired cluster topics	4.67±0.69
13-The user interface is a useful tool for document clustering in general	4.67±0.59
14-I would like to use the system in the future	4.61±0.70

Three more questions were asked from the users to test the *ease of use* and the *learnability* of the visualization. The first question asked which cluster has the highest number of documents and 83% of participants answered correctly. The second question was about the number of repetitive documents in the collection and 45% of

participants were able to answer this question correctly. For the third question, participants needed to give the number of documents with more than one cluster labels. The participants 67% answered this question correctly. The low correct answer rate for the complicated questions such as the second question indicates that we need to reduce the complexity of the interface and provide a more straightforward solution for the users.

The result of the post-task questionnaire is in Table 2.10. The goal of these questions was to get the users experience and opinion during the user study. The first 10 questions of Table 2.10 are selected from the Software Usability Scale (SUS) questionnaire [21]. The overall average result of answers of participants to the questions of Table 2.10 demonstrates the effectiveness and usefulness of the proposed system for interactive document clustering.

## 2.5 Conclusion

In this chapter, we introduced a novel solution for interactive document clustering. First, the proposed solution was evaluated in real world datasets by end users, demonstrating significant improvement in the quality of the clusters over fully automatic clustering. We built our system based on key-term interaction because of its intuitiveness for the user. Second, we introduced an interactive version of KMeans called iKMeans. The proposed method for iKMeans could be applied to other clustering algorithms and consequently could be employed in the proposed system. Third, we have combined the t-SNE algorithm with force directed display for improved projection of documents. The evaluation result demonstrates the effectiveness of the proposed system on improving the clustering result.



## Chapter 3

### Deterministic Seeding of KMeans

In chapter 2 a user study is conducted to evaluate the performance of the proposed interactive document clustering algorithm. Based on the result of the user study, we found out that random initialization of the clustering algorithm may confuse the user because of the inconsistency of results. In this chapter<sup>1</sup>, we try to improve the quality of KMeans by introducing a novel initialization method and at the same time provide consistence result. KMeans is usually initialized by random seeds that can drastically impact the final algorithm performance. There exist many random or order-sensitive methods that try to properly initialize KMeans but their problem is that their result is non-deterministic and unrepeatable. Thus KMeans needs to be initialized several times to get a better result which is a time-consuming operation. In this chapter, we introduce a novel deterministic seeding method for KMeans that is specifically designed for text document clustering. Due to its simplicity, it is fast and can be scaled to large datasets. Experimental results on several real-world datasets demonstrate that the proposed method has overall better performance compared to several deterministic, random, or order-sensitive methods of initializing KMeans in terms of clustering quality and runtime.

#### 3.1 Introduction

The objective of KMeans is to assign similar data points to the same cluster while they are dissimilar to other clusters. The gradient descent method is usually used for optimizing the objective function and due to the non-convex nature of KMeans, the initial seeds play an important role in the quality of the clustering. There are several research works that try to provide good seeds for the KMeans. These methods can be divided into two major categories of non-deterministic and deterministic methods [27].

---

<sup>1</sup>An earlier version of this chapter appeared as [94]

The non-deterministic methods are random or order-sensitive in nature. KMeans Plus Plus is a well known seeding method that incrementally selects initial seeds one at a time [5]. In each step, a data point is selected with a probability proportional to the minimum distance to the previously selected seeds. Because the first seed in KMeans++ is determined randomly and next seeds are selected based on a probabilistic method, the initial seeds are not repeatable. The KMC2 method improves the KMeans++ sampling step by Markov chain Monte Carlo based approximation [7]. Similarly to KMeans++, KMC2 starts with a uniformly random seed then the next seeds are selected by Markov chains of size  $m$ . The key factor for speeding up the KMC2 is that for each seed selection, it does not need to fully pass through all the data points and it only needs to compute the distance between  $m$  data points and previously selected seeds. The  $m$  is a fixed value, independent of the number of data points.

While there are many non-deterministic seeding methods, there exist few deterministic ones. The deterministic approaches need to be run only once and it makes them more practical for larger datasets. The comparison between different deterministic methods is presented by [26]. The KKZ method is one of the first deterministic seeding methods for KMeans [50]. It first sorts data points by their vector's norm and the one with the highest value is selected as the first seed. The next seeds will be selected from data points that have the largest distance to the closest previously selected seeds. The most important drawback of this method is that it is sensitive to outliers. To avoid selecting an outlier as the initial seed, the ROBIN approach [41] uses local outlier factor (LOF) method [20]. This method first starts with a reference point  $r$  that usually is the origin of data points. Then it sorts data points in decreasing order of their minimum distances from  $r$ . It then traverses the sorted list and selects the first non-outlier node, based on its LOF value. For the next steps, it sorts data points in decreasing order by their minimum distance to the previous seeds and, again, the first non-outlier node is the next seed. The LOF method is not applicable to high dimensional and sparse datasets, which is an important issue in textual document collections [2].

The PCA-part and VAR-part are two popular deterministic hierarchical initialization methods for KMeans [96]. They start with all data points as a single cluster and

then divide the data point into two halves based on Principle Component Analysis (PCA) [1].

This process continues and at each step, the half with largest average distance to its centroid is divided into two parts until the required number of seeds is reached. The result of the previous steps is an approximate clustering of data points; the centroid of the clusters are used for initializing KMeans.

There are some applications that require determinism. Interactive document clustering is a task that involves a human domain expert in the clustering procedure [13]. First, the clustering algorithm provides the user with initial clustering results, then the user provides feedback to reflect her idea of a meaningful clustering. If the initial result is non-deterministic, the user may get confused by the inconsistency clustering result. It is possible to store the initial data points to make the result of a non-deterministic method repeatable, but it may lead to a bad quality solution unless one initializes the clustering algorithm several times and then consider the one which has optimized the objective function the most which is a very time-consuming process. In a medical domain, such as cancer subtype prediction, it is essential to have deterministic clusters for making a consistent decision and be able to compare the clustering results with other clustering algorithms [72]. There is a particular treatment plan for each cancer subtype and in case that a subtype is clustered differently with different seeds it may impact the patients' treatment procedure.

In this chapter, we introduce a simple deterministic seeding method for the KMeans algorithm, called DSKM, with the target of text document clustering. The proposed method is not only deterministic and reproducible but also improves the overall clustering results. The proposed method tries to find initial seeds that are as diverse as possible which consequently lead to a better clustering result. The KMeans need to be initialized by DSKM only once and this makes it fast and applicable to large datasets. All the code and data is publicly available<sup>2</sup>.

In Section 3.2, we present our proposed initializing method DSKM. We provide a review of baseline methods in Section 3.3 and a detailed description of datasets and evaluation metrics in Section 3.4. Finally, we report extensive experimental results in Section 3.5.

---

<sup>2</sup><https://github.com/ehsansherkat/DSKM>

### 3.2 Proposed Method

The key idea of the proposed method is to select  $k$  data points that are far from each other and, at the same time, have a high  $L_1$  norm. These data points are used to initialize the KMeans algorithm. Steps of the proposed method is described in the following.

**Step 1** First the document vectors are created based on terms of document collection after removing numbers, punctuations and stop-words. The document-term matrix produced as a result of this step is the input of the Algorithm 3. Let  $D$  be the set of documents and  $d$  a document in  $D$ . The Tfidf weight of term  $w$  in document  $d$  is defined as Eq. 3.1, which is a smoothed variant of the classical TF-idf.

$$\text{Tf\_idf}(\mathbf{w}, \mathbf{d}, D) = f(\mathbf{w}, \mathbf{d}) \times \log \frac{|D| + 1}{|x \in D : w \in x| + 1} + 1. \quad (3.1)$$

where  $f(w, d)$  is the frequency of term  $w$  in document  $d$ . Each document vector is then normalized by the  $L_2$  norm. The high dimensionality of vectors may impact the results of the clustering algorithm. To reduce the dimensionality, we use a simple but effective approach for pruning: the terms with a lower *mean-Tf-idf score* than the average mean-Tf-idf of all terms.

For each term, the *mean-Tf-idf score* is calculated based on Eq. 3.2.

$$\text{mean\_Tf\_idf}(\mathbf{w}, D) = \frac{1}{|D|} \times \sum_{d \in D} \text{tf\_idf}(\mathbf{w}, \mathbf{d}, D). \quad (3.2)$$

**Step 2** The rows of the document-term matrix are sorted by  $L_1$  norm in a way that the first row of the matrix is the document with the highest  $L_1$  norm. Documents with a higher  $L_1$  norm have more impact on grouping similar documents because of having more key-terms. Therefore, we select the document with the highest  $L_1$  norm as the starting data point ( $s_0$ ). This procedure will generally not select an outlier document as a seed document.

**Step 3** In the third step, we find a data point that is far from the starting data point and consider it as the first seed.

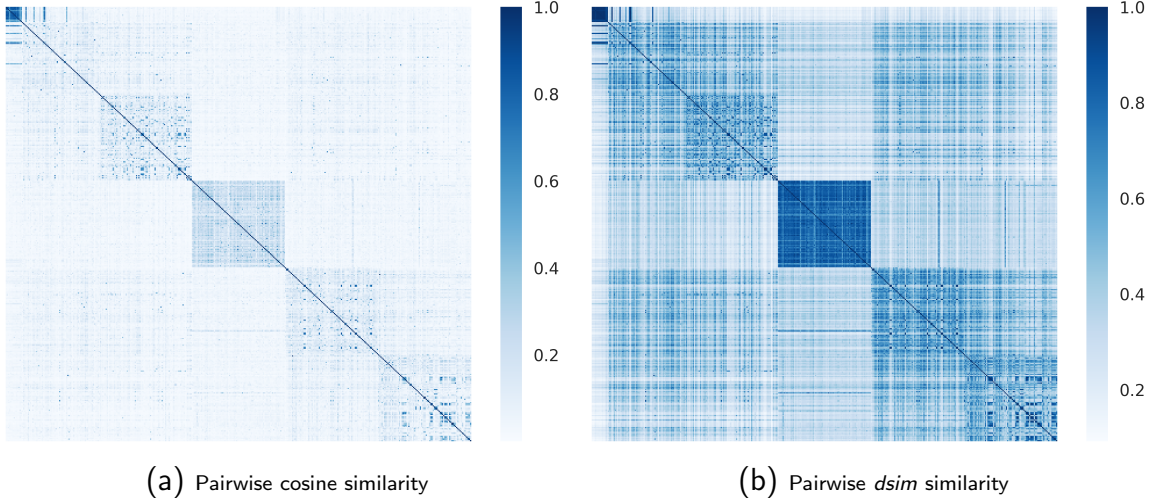


Figure 3.1: The comparative result of pairwise cosine and *dsim* similarity of Newsgroup5 dataset. The darker color indicates the higher similarity between two documents. The documents are sorted by their class labels and five clusters are clearly detectable in both heat maps.

Let  $C^{|D| \times |D|}$  be the pairwise cosine similarity matrix between each pair of documents. Let  $c_{d_i}$  be the  $i$ -th row of  $C$ .  $c_{d_i}$  corresponds to the vector of similarities of document  $d_i$  with every other document. It has been shown that the cosine similarity is a better similarity metric than Euclidean distance for comparing textual documents [12]. We define the double similarity (*dsim*) between the document  $d_i$  to document  $d_j$  as Eq. 3.3.

$$\mathbf{dsim}(d_i, d_j) = \frac{c_{d_i} \cdot c_{d_j}}{\|c_{d_i}\|_2 \|c_{d_j}\|_2}. \quad (3.3)$$

The insight for using *dsim* is that not only two documents, but also their similar documents, should be far from each other. Using *dsim* can help to achieve this goal. The comparison between heat maps of pairwise cosine and *dsim* similarity of Newsgroup5 dataset is depicted in Figure 3.1. The darker colors in the *dsim* heat map indicate that two documents may have considerable number of common similar documents. It means that, two documents may be more similar to each other if we compare their similar documents with each other than directly comparing them.

Let  $A$  be the list of document indexes sorted in decreasing order by their  $L_1$  norm. The goal of the third step is finding the first document which has *dsim* similarity less

than a specific threshold from the starting point ( $s_0$ ) by traversing from the first of list  $A$  (Lines 8-13 Algorithm 3).

Let  $S$  be the set of seed documents and  $s_i \in S$  be the document index of seed  $i$ . The similarity threshold (Lines 1-3 Algorithm 3) is calculated based on Eq. 3.4.

$$T(s_i) = \frac{1}{|D|} \sum_{j=1}^{|D|} \text{dsim}(d_j, S_i). \quad (3.4)$$

$T(s_0)$  is the threshold for finding the first seed based on the starting data point  $s_0$ . We do not consider the starting data point as the first seed but we will give the chance for it to be selected in the next steps. Using Eq. 3.4 as the threshold prevents to select documents that are at the very end of list  $A$  which have low  $L_1$  norm and less impact on grouping similar documents. After having found the first document  $s_1$  that passes the threshold, we stop considering other documents and we add it to the seed document set  $S$ . Now, the seed documents set has the size of 1.

**Step 4** We find  $k-1$  more seed documents in this step. We start from the beginning of list of  $A-S$  and find the first document which is far from every seed in set  $S$  based on the threshold defined by Eq. 3.4. We iterate this step until  $k$  seeds are determined (Lines 16-22 Algorithm 3). In the case that there is no document far from all the seeds in  $S$ , the following objective function is considered, with the goal of finding the document, which has the lowest cumulative  $\text{dsim}$  to every other seed document (Lines 23-25 Algorithm 3).

$$\text{argmin} \left( \sum_{j=1}^{|S|} \text{dsim}(d_i, s_j) \right), \quad 1 \leq i \leq |D|, \quad d_i \notin S. \quad (3.5)$$

This step ensures that the proposed method can always find  $k$  seed documents in every document collection.

After finding the initial seeds, we can directly initialize the KMeans algorithm. Based on our experiments, we can achieve a higher quality of result if for each seed document we find a few similar documents based on cosine similarity and then consider their centroid as the final seed. In our experiments, we extended each seed document with 15 similar documents for all datasets.

---

**Algorithm 3:** Deterministic seeding KMeans (DSKM)
 

---

```

input : k: Number of clusters
          Data|D|×|W|: document-term matrix // Step1
output: S:{s1, s2, ..., sk} = Set of seed documents index

1 Function T(si): // Threshold function
2 | return  $\frac{1}{|D|} \sum_{j=1}^{|D|} dsim(d_j, s_i)$ ;
3 end

4 C|D|×|D| ← pairwise-similarity(Data, 'cosine');
5 A:{a1, a2, ..., a|D|} ← sort(Data, 'L1 norm');
6 s0 ← C[a1] // Set starting point; Step 2;
7 S ← {}

8 for i ← 1 to |D| do // Step 3
9 | if dsim(C[s0], C[ai]) < T(s0) then
10 | | S ← ai;
11 | | break;
12 | end
13 end

14 while |S| < k do // Step 4
15 | found ← False;
16 | for i ← 1, ai ∉ S to |D| do
17 | | if dsim(C[sj], C[ai]) < T(sj), ∀sj ∈ S then
18 | | | S ← ai;
19 | | | found ← True;
20 | | | break;
21 | | end
22 | end
23 | if found == False then
24 | | S ← argmin( $\sum_{j=1}^{|S|} dsim(a_i, s_j)$ ), ∀ai ∈ A, ai ∉ S
25 | end
26 end
27 return S

```

---

**Complexity Analysis:** Let  $|D| = n$  be the number of documents and  $m$  the number of unique terms after applying Eq. 3.2 filter. The time complexity of sorting document-term-matrix and calculating the cosine similarity matrix is  $O(n \log n)$  and  $O(n^2m/2)$  while the time complexity of finding seed documents based on  $dsim$  is  $O(n^2k)$ . Calculating the cosine similarity matrix is the most time-consuming step of the proposed method but it could easily be processed in parallel. On the other hand, pairwise similarity matrix is a one time process and it could be calculated once in off-line processing. In reality, the size of  $m$  will be less than a few thousand even for large textual datasets after selecting important terms, which makes the proposed approach practically feasible.

### 3.3 Baseline Methods

We compare three random or order-sensitive seeding methods, Points, KMeans++, and KMC2 with the proposed method. In the Points method, uniformly  $k$  randomly selected data points are considered as the initial seeds for the KMeans algorithm. KMeans++ is one of the most widely used seeding methods which has been demonstrated to achieve better performance result than the Points method [5]. KMeans++ starts with a random seed, then it tries to find the next one as far as possible from the first seed based on a probability sampling method called  $D^2$ -sampling. In this sampling method, data points that have higher distance to the previously selected seeds will more likely be selected as the next seed. This process continues until  $k$  initial seeds are detected. KMC2 method is speeding up KMeans++ algorithm by Markov chain Monte Carlo sampling based approximation [7]. It has been reported that the KMC2 has a better quality of results and computational cost than the KMeans++ algorithm. In our experiments, we use the assumption-free version of KMC2 with  $m$  equals to 200.

Two widely used deterministic seeding methods of PCA-part and VAR-part are compared with the proposed method. The PCA-part method hierarchically divides the data points into two halves based on PCA. First, it starts with calculating the centroid of all data points as a single cluster, and the principal eigenvector of the cluster covariance matrix. Second, it passes through an hyperplane orthogonal to the principal eigenvector of the cluster which goes from the cluster centroid to create two



Table 3.1: Description of Datasets. The Eq. 5.2 is used for feature selection for the first 7 datasets and for the rest only stop-words and words with frequency less than 20 are removed.

#	Dataset	#Samples	#Dim.	#Classes
1	Newsgroup5	400	1450	5
2	Yahoo6	600	2206	6
3	R8	7674	1997	8
4	Newsgroup20	18846	11556	20
5	WebKB	4199	1578	4
6	NewsSeparate	381	380	13
7	SMS	5549	858	2
8	BBCsport	737	969	5
9	BBC	2225	3121	5
10	Wikilow	4986	15441	10
11	WikiHigh	5738	17311	6
12	Guardian	6520	10801	6
13	Irishtimes	3246	4823	7

sub-clusters. The sum distance of each data point in each sub-cluster to its centroid is calculated and the sub-cluster with a higher value is divided in the next step. Finally, this procedure is continued until  $k$  clusters are obtained. The VAR-part (variance partitioning) is an approximation to the PCA-part method [96]. In VAR-part the covariance matrix of the cluster is assumed to be diagonal. In each partitioning stage, the hyperplane is diagonal to the dimension with the largest variance. Based on our experiments, using the Euclidean distance leads to similar initialized seeds compared to cosine distance for VAR-par and PCA-part in all datasets; therefore we used the Euclidean distance for both methods.

In our experiments, we used the Spherical version of the KMeans algorithm. In Spherical KMeans the feature vectors are projected to the unit sphere equipped with the cosine similarity which performs better than Euclidean distance for text document clustering [33].

We compare the Spherical KMeans with different seeding methods with Fuzzy CMeans and Von Mises-Fisher Mixture methods. In the Fuzzy CMeans algorithm the data points can belong to more than one cluster with different membership values rather than distinct membership to only one cluster [15]. In our experiments, we

Table 3.2: Comparing precision of seeds. The average ( $\pm$  std) over 50 runs is reported for the Points, KMeans++, and KMC2 methods.

Dataset	DSKM	Points	KMeans++	KMC2
Newsgroup5	<b>0.800</b>	0.684 $\pm$ 0.145	0.636 $\pm$ 0.182	0.692 $\pm$ 0.134
Yahoo6	<b>1.000</b>	0.700 $\pm$ 0.115	0.613 $\pm$ 0.131	0.677 $\pm$ 0.070
R8	<b>0.750</b>	0.393 $\pm$ 0.120	0.495 $\pm$ 0.135	0.443 $\pm$ 0.137
Newsgroup20	<b>0.700</b>	0.634 $\pm$ 0.064	0.617 $\pm$ 0.072	0.638 $\pm$ 0.060
WebKB	<b>1.000</b>	0.660 $\pm$ 0.179	0.610 $\pm$ 0.151	0.655 $\pm$ 0.165
NewsSeparate	<b>0.846</b>	0.582 $\pm$ 0.084	0.563 $\pm$ 0.089	0.614 $\pm$ 0.103
SMS	<b>1.000</b>	0.620 $\pm$ 0.214	0.630 $\pm$ 0.219	0.610 $\pm$ 0.207
BBCsport	<b>0.800</b>	0.660 $\pm$ 0.140	0.576 $\pm$ 0.148	0.656 $\pm$ 0.133
BBC	<b>0.800</b>	0.668 $\pm$ 0.153	0.580 $\pm$ 0.146	0.688 $\pm$ 0.145
Wikilow	<b>0.800</b>	0.646 $\pm$ 0.090	0.556 $\pm$ 0.098	0.676 $\pm$ 0.111
WikiHigh	<b>0.833</b>	0.653 $\pm$ 0.152	0.627 $\pm$ 0.131	0.687 $\pm$ 0.123
Guardian	<b>1.000</b>	0.643 $\pm$ 0.105	0.577 $\pm$ 0.138	0.667 $\pm$ 0.120
Irishtimes	<b>0.857</b>	0.611 $\pm$ 0.114	0.509 $\pm$ 0.149	0.643 $\pm$ 0.112

use cosine similarity for the distance measure of the Fuzzy CMeans. The Von Mises-Fisher Mixture methods is a mixture model for clustering data distributed on the unit hypersphere based on Von Mises-Fisher distribution [9].

### 3.4 Datasets and Evaluation Metrics

**Datasets** Different text document clustering datasets with a diverse number of clusters and documents are adopted to evaluate the proposed method. The description of datasets is provided in Table 3.1.

We obtained dataset Newsgroup5 by selecting 5 categories of the Newsgroup20<sup>3</sup> dataset each containing 80 randomly chosen documents. The Newsgroups20 dataset consists of nearly 20,000 messages of Internet news articles with 20 categories. The Yahoo6 is a sub-collection of questions and answers extracted from the Yahoo! Answers website [28]. We used 6 sub-categories with 100 randomly selected question and answer pairs. R8 is a subset of *Reuters-21578* dataset containing 8 categories and can be downloaded from Ana Cachopo’s homepage<sup>4</sup>. The WebKB dataset consists of 4199 faculty, student, project, and course websites collected from the four universities on

<sup>3</sup><http://qwone.com/~jason/20Newsgroups/> — last accessed: Oct. 16, 2018

<sup>4</sup><http://ana.cachopo.org> — last accessed: Oct. 16, 2018

Table 3.3: Comparing clustering accuracy. For the deterministic approaches the McNemars test is used. The P-value less than 0.05 indicates that the clustering algorithm does not have the same error rate as DSKM approach. The average over 50 runs with standard deviation is reported for the random or order-sensitive methods in which the  $m$  shows the minimum and the  $M$  shows the maximum of 50 runs.

Dataset	KMeans (DSKM)	KMeans (PCA-part)	KMeans (VAR-part)	KMeans (Points)	KMeans (KMeans++)	KMeans (KMC2)	Fuzzy CMeans (Points)	Von Mises Fisher Mixture
NewsGroup5	<b>0.850</b>	0.740 p < 0.05	0.525 p < 0.05	0.687 ± 0.082 m:0.522 M:0.91	0.696 ± 0.095 m:0.555 M:0.922	0.706 ± 0.080 m:0.542 M:0.912	0.719 ± 0.056 m:0.505 M:0.785	0.666 ± 0.073 m:0.497 M:0.820
Yahoo6	<b>0.850</b>	0.827 p > 0.05	0.803 p < 0.05	0.756 ± 0.079 m:0.553 M:0.847	0.740 ± 0.072 m:0.577 M:0.850	0.746 ± 0.070 m:0.620 M:0.843	0.798 ± 0.062 m:0.633 M:0.830	0.645 ± 0.052 m:0.457 M:0.757
RS	<b>0.688</b>	0.411 p < 0.05	0.537 p < 0.05	0.468 ± 0.060 m:0.332 M:0.605	0.476 ± 0.052 m:0.381 M:0.585	0.474 ± 0.064 m:0.361 M:0.612	0.457 ± 0.045 m:0.368 M:0.539	0.431 ± 0.054 m:0.271 M:0.513
NewsGroup20	0.485	<b>0.517</b> p < 0.05	0.386 p < 0.05	0.478 ± 0.037 m:0.399 M:0.565	0.496 ± 0.041 m:0.378 M:0.605	0.484 ± 0.039 m:0.410 M:0.595	0.119 ± 0.003 m:0.114 M:0.126	0.343 ± 0.024 m:0.303 M:0.407
WebKB	<b>0.65</b>	0.609 p < 0.05	0.529 p < 0.05	0.609 ± 0.029 m:0.521 M:0.669	0.604 ± 0.033 m:0.539 M:0.661	0.605 ± 0.039 m:0.529 M:0.692	0.603 ± 0.041 m:0.514 M:0.660	-
NewsSeparate	<b>0.861</b>	0.711 p < 0.05	0.766 p < 0.05	0.727 ± 0.072 m:0.562 M:0.89	0.713 ± 0.059 m:0.583 M:0.861	0.748 ± 0.066 m:0.622 M:0.864	0.747 ± 0.048 m:0.627 M:0.874	0.679 ± 0.066 m:0.507 M:0.824
SMS	0.597	0.904 p < 0.05	<b>0.907</b> p < 0.05	0.675 ± 0.142 m:0.502 M:0.907	0.646 ± 0.139 m:0.502 M:0.907	0.667 ± 0.143 m:0.505 M:0.907	0.797 ± 0.037 m:0.721 M:0.839	-
BBCsport	0.856	0.670 p < 0.05	<b>0.951</b> p < 0.05	0.783 ± 0.115 m:0.521 M:0.961	0.789 ± 0.117 m:0.620 M:0.958	0.800 ± 0.124 m:0.514 M:0.958	0.869 ± 0.123 m:0.626 M:0.955	0.803 ± 0.122 m:0.528 M:0.955
BBC	0.956	<b>0.958</b> p > 0.05	0.953 p > 0.05	0.870 ± 0.116 m:0.654 M:0.962	0.817 ± 0.133 m:0.493 M:0.965	0.833 ± 0.142 m:0.443 M:0.965	0.948 ± 0.035 m:0.704 M:0.953	0.809 ± 0.108 m:0.539 M:0.953
Wikilow	0.763	<b>0.969</b> p < 0.05	0.834 p < 0.05	0.803 ± 0.101 m:0.466 M:0.968	0.771 ± 0.096 m:0.581 M:0.964	0.793 ± 0.097 m:0.477 M:0.967	0.843 ± 0.075 m:0.702 M:0.964	0.751 ± 0.067 m:0.590 M:0.870
WikiHigh	0.715	<b>0.861</b> p < 0.05	0.658 p < 0.05	0.774 ± 0.087 m: 0.544 M:0.88	0.774 ± 0.087 m:0.487 M:0.890	0.785 ± 0.069 m:0.655 M:0.874	0.851 ± 0.026 m:0.712 M:0.867	0.629 ± 0.062 m:0.496 M:0.730
Guardian	<b>0.951</b>	0.951 p > 0.05	0.950 p > 0.05	0.834 ± 0.104 m: 0.583 M:0.954	0.832 ± 0.108 m:0.574 M:0.955	0.837 ± 0.121 m:0.554 M:0.954	0.945 ± 0.013 m:0.856 M:0.947	0.851 ± 0.097 m:0.661 M:0.945
Irishtimes	<b>0.871</b>	0.772 p < 0.05	0.626 p < 0.05	0.695 ± 0.083 m: 0.518 M:0.871	0.671 ± 0.085 m:0.505 M:0.837	0.678 ± 0.084 m:0.498 M:0.827	0.784 ± 0.074 m:0.625 M:0.877	0.704 ± 0.059 m:0.574 M:0.837

January 1997<sup>5</sup>. The NewsSeparate dataset is a subset of RSS news feeds from BBC, CNN, Reuters and Associated Press manually categorized into 13 categories [68]. The SMS dataset is a set of labeled SMS messages for spam research<sup>6</sup>.

Datasets number 8 to 13 are taken from [38] and can be downloaded from their web-page<sup>7</sup>. The BBCsport, BBC, Irishtimes, and Guardian are news articles and WikiHigh and Wikilow are a subset of a Wikipedia dump from January 2014.

**Evaluation Metrics** The clustering quality is measured by two widely used document clustering evaluation metrics of Normalized Mutual Information (NMI) and Accuracy (Acc) [22]. These metrics generate values between 0 and 1 in which values closer to 1 shows better performance. To match the predicted labels with actual labels for calculating the accuracy, we used the Hungarian method [53].

We compare the precision of initial seeds of methods defined by Eq. 3.6. The true

<sup>5</sup><http://cs.cmu.edu/afs/cs/project/theo-20/www/data/> — last accessed: Oct. 16, 2018

<sup>6</sup><http://dt.fee.unicamp.br/~tiago/smsspamcollection/> — last accessed: Oct. 16, 2018

<sup>7</sup><http://mlg.ucd.ie/howmanytopics/index.html> — last accessed: Oct. 16, 2018

Table 3.4: Comparing clustering NMI score. The average 50 runs with standard deviation is reported for the random or order-sensitive approaches in which the  $m$  shows the minimum and the  $M$  shows the maximum of 50 runs.

Dataset	KMeans (DSKM)	KMeans (PCA-part)	KMeans (VAR-part)	KMeans (Points)	KMeans (KMeans++)	KMeans (KMC2)	Fuzzy CMeans (Points)	Von Mises Fisher Mixture
NewsGroup5	<b>0.781</b>	0.742	0.513	0.663 ± 0.075 m:0.437 M:0.829	0.667 ± 0.074 m:0.511 M:0.821	0.665 ± 0.066 m:0.550 M:0.815	0.663 ± 0.032 m:0.537 M:0.706	0.622 ± 0.069 m:0.442 M:0.777
Yahoo6	<b>0.704</b>	0.684	0.645	0.631 ± 0.043 m:0.492 M:0.700	0.621 ± 0.044 m:0.538 M:0.693	0.629 ± 0.041 m:0.532 M:0.694	0.664 ± 0.028 m:0.585 M:0.678	0.538 ± 0.03 m:0.449 M:0.615
RS	<b>0.575</b>	0.534	0.509	0.515 ± 0.032 m:0.420 M:0.567	0.520 ± 0.027 m:0.460 M:0.580	0.527 ± 0.029 m:0.453 M:0.600	0.480 ± 0.032 m:0.425 M:0.548	0.397 ± 0.057 m:0.260 M:0.495
NewsGroup20	<b>0.539</b>	0.533	0.467	0.498 ± 0.023 m:0.453 M:0.554	0.509 ± 0.028 m:0.439 M:0.578	0.501 ± 0.024 m:0.456 M:0.567	0.234 ± 0.002 m:0.232 M:0.239	0.412 ± 0.018 m:0.365 M:0.45
WebKB	<b>0.388</b>	0.320	0.353	0.362 ± 0.017 m:0.324 M:0.396	0.362 ± 0.017 m:0.316 M:0.395	0.363 ± 0.016 m:0.322 M:0.394	0.349 ± 0.023 m:0.307 M:0.377	-
NewsSeparate	<b>0.872</b>	0.809	0.829	0.819 ± 0.035 m:0.729 M:0.899	0.813 ± 0.033 m:0.742 M:0.882	0.833 ± 0.031 m:0.777 M:0.893	0.826 ± 0.022 m:0.767 M:0.894	0.77 ± 0.036 m:0.679 M:0.868
SMS	0.123	0.409	<b>0.414</b>	0.120 ± 0.128 m:0.000 M:0.414	0.135 ± 0.115 m:0.002 M:0.413	0.140 ± 0.119 m:0.000 M:0.414	0.267 ± 0.043 m:0.165 M:0.317	-
BBCsport	0.761	0.716	<b>0.858</b>	0.742 ± 0.077 m:0.583 M:0.881	0.742 ± 0.079 m:0.578 M:0.876	0.752 ± 0.089 m:0.570 M:0.876	0.816 ± 0.066 m:0.692 M:0.869	0.743 ± 0.091 m:0.461 M:0.876
BBC	0.865	<b>0.871</b>	0.857	0.806 ± 0.072 m:0.663 M:0.880	0.772 ± 0.091 m:0.536 M:0.891	0.774 ± 0.090 m:0.557 M:0.889	0.851 ± 0.020 m:0.708 M:0.856	0.718 ± 0.086 m:0.494 M:0.859
Wikilow	0.867	<b>0.934</b>	0.897	0.862 ± 0.040 m:0.730 M:0.933	0.853 ± 0.037 m:0.781 M:0.930	0.862 ± 0.039 m:0.740 M:0.931	0.879 ± 0.027 m:0.825 M:0.927	0.774 ± 0.031 m:0.713 M:0.832
WikiHigh	0.723	<b>0.740</b>	0.642	0.707 ± 0.047 m:0.580 M:0.761	0.702 ± 0.043 m:0.548 M:0.764	0.704 ± 0.037 m:0.633 M:0.759	0.721 ± 0.016 m:0.620 M:0.727	0.552 ± 0.041 m:0.479 M:0.667
Guardian	<b>0.862</b>	<b>0.862</b>	0.861	0.805 ± 0.054 m:0.627 M:0.870	0.803 ± 0.056 m:0.647 M:0.872	0.807 ± 0.062 m:0.644 M:0.871	0.852 ± 0.015 m:0.748 M:0.856	0.786 ± 0.055 m:0.639 M:0.848
Irishtimes	<b>0.783</b>	0.720	0.642	0.681 ± 0.049 m:0.575 M:0.759	0.672 ± 0.055 m:0.564 M:0.761	0.680 ± 0.052 m:0.573 M:0.761	0.741 ± 0.032 m:0.666 M:0.780	0.672 ± 0.036 m:0.595 M:0.740

label of each initial seed is used to find the diversity of label of seeds. The method with more diverse (their true labels be different) initial seeds is better because it is able to introduce a better representative seed for each cluster. The comparative result of seed precision of evaluation methods is given in Table 3.2. The PCA-part and VAR-part produce initial centroids instead of initial seeds so it is not possible to evaluate their seed precision.

$$SeedPrecision = \frac{\#diverse\ labels}{k}. \quad (3.6)$$

### 3.5 Experimental Results

The accuracy of the DSKM in comparison to other methods is summarized in Table 3.3. For random or order-sensitive methods, we report the average over 50 runs with its standard deviation, the minimum, and the maximum result. In order to have a fair comparison, we only initialize KMeans once for the non-deterministic methods. For the PCA-part and VAR-part methods, the McNemar’s test is applied to determine whether their clustering result has the same error rate as DSKM. The

Table 3.5: Running time (seconds) of seeding methods. A random single run of KMeans++ and KMC2 is reported.

Dataset	DSKM	PCA-part	VAR-part	KMeans++	KMC2
Newsgroup5	<b>0.03</b>	5.27	<b>0.03</b>	0.05	<b>0.03</b>
Yahoo6	0.02	10.08	0.04	<b>0.01</b>	0.02
R8	3.22	172.67	0.90	<b>0.25</b>	0.47
Newsgroup20	55.96	19712.72	39.56	8.28	<b>6.28</b>
WebKB	0.72	-	-	0.11	<b>0.06</b>
NewsSeparate	0.03	0.74	0.03	0.02	<b>0.01</b>
SMS	0.77	7.92	0.11	<b>0.02</b>	0.03
BBCsport	0.06	4.79	0.03	<b>0.01</b>	<b>0.01</b>
BBC	0.38	94.93	0.39	0.08	<b>0.07</b>
Wikilow	7.02	6849.23	5.62	1.45	<b>0.70</b>
WikiHigh	8.75	8725.6	5.59	1.21	<b>0.71</b>
Guardian	6.02	3681.96	3.88	0.82	<b>0.44</b>
Irishtimes	0.99	410.3	1.25	0.22	<b>0.14</b>

Hungarian algorithm is used to map the cluster labels to actual labels. The deterministic approaches are superior in accuracy score compared to the average score of random or order-insensitive methods. Better performance for deterministic methods on non-textual and Synthetic datasets has been reported by [27]. A possible reason is that the deterministic methods are running once and the seeding step can be viewed as an approximate clustering of data points. The DSKM method has similar or even better accuracy compared to the maximum accuracy score of the random or order-sensitive methods on Yahoo6, R8, WebKB, NewsSeparate, BBC, Guardian, and Irishtimes. The SMS dataset is an unbalanced dataset and DSKM does not perform well on it although it was able to find 100% diverse initial seeds (Table 3.2). PCA-part, and VAR-part performed well on the SMS dataset which demonstrates their effectiveness for unbalanced datasets. Fuzzy CMeans has the best average and Von Mises Fisher Mixture the lowest accuracy score on most of the datasets among random or order-sensitive methods. On the Newsgroup20 dataset, Fuzzy CMeans does not perform well, which indicates that this method has difficulty on large datasets with a high number of clusters. The Points, KMeans++, and KMC2 have similar average accuracy result on most datasets. This shows that KMeans++ and KMC2 are performing better for very large datasets which it is a case for Newsgroup20 and R8 datasets.

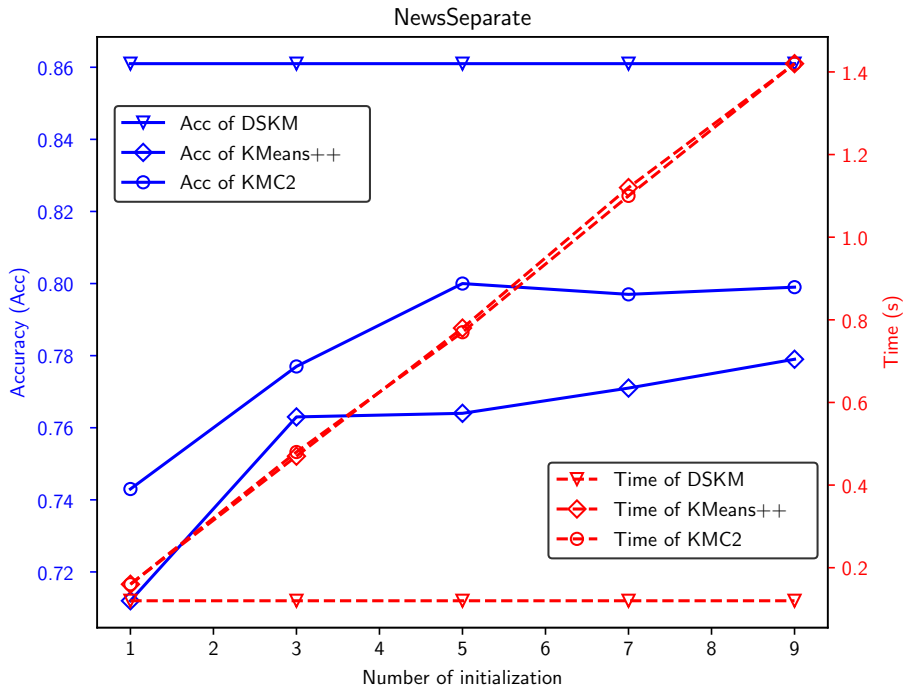


Figure 3.2: The impact of number of initialization on the Accuracy performance and running time. Each initialization of the KMeans++ and KMC2 is the result of average 50 runs.

The NMI score of the proposed method compared to other methods is summarized in Table 3.4. The DSKM is outperformed in most of the datasets. The same trend of performance similar to the accuracy score can be observed for NMI score as well. KMC2 has slightly better NMI score compared to KMeans++ and Points.

We compare the running time of the seeding methods in Table 3.5. Although the PCA-part has better performance result than the VAR-part, its running time makes it not practical for large datasets. The DSKM method has acceptable running time even for large datasets. The KMC2 is the fastest seeding algorithm compared to the others and based on its accuracy and NMI performance, it is the best random or order-sensitive method. Due to the random nature of the KMeans++ and KMC2, the Kmeans is initialized several times by them and the clustering which optimizes the KMeans objective function is selected. The impact of the number of initializations on the accuracy performance of the KMeans++ and KMC2 for NewsSeparate is depicted in Figure 3.2. In order to have stable results, we reported the average of 50 runs for KMC2 and KMeans++. As the number of initialization increases, the accuracy of

the KMC2 and KMeans++ increases and converges to a stable value. On the other hand, the running time increases as the number of initializations is increased. This indicates that the DSKM method could be even faster than the random or order-sensitive methods in practice because it does not need to run several times.

### 3.6 Conclusion

In this chapter, a new deterministic seeding algorithm for the KMeans algorithm called DSKM is proposed. The key idea of the DSKM is that the initial seeds should be as far as possible from each other. Two data points that not only themselves but their similar documents are less similar to each other are good candidates and that is why we have defined the *dsim* similarity. For finding seeds we start from documents with higher  $L_1$  norm. Experimental results on several real world textual datasets shows that DSKM outperforms other deterministic, random or order-sensitive methods in terms of clustering accuracy and NMI score. The proposed methods have an acceptable running time even for large datasets. In the future, we will incorporate the temporal aspect in the process of clustering and will use the similar initialization method that proposed in this chapter for initializing the proposed method for interactive temporal document clustering.

## Chapter 4

### Interactive Temporal Document Clustering

In chapter 2 we have incorporated KMeans-like document clustering algorithms in the loop of interactive document clustering. Considering the temporal aspect in the process of clustering is needed when one wants to extract the trends and evolution of clusters over time. In this chapter<sup>1</sup>, we aim to incorporate the temporal aspect in the procedure of document clustering. First, a time-based similarity measure is introduced and following that a novel temporal interactive document clustering algorithm is presented in detail. Second, we explain how to consider the temporal aspect in the visualization. Finally, we discuss the case study of clustering a real world dataset belonging to the Canadian Society of Respiratory Therapists.

#### 4.1 Incorporating Temporal Aspect in the Clustering Algorithm

In this section, we will first introduce a temporal based similarity for textual documents. This similarity is the combination of the content and temporal similarity between documents. Experimental results show that considering the temporal similarity could improve the overall clustering performance in some datasets. Second, we have designed a new clustering algorithm that can incorporate the user key-term interactions in the objective function of the clustering algorithm and at the same time consider the temporal similarity between documents. Notations used in this section is explained in Table 4.1.

##### 4.1.1 Temporal Similarity

It is possible to consider the temporal similarity from different aspects. One could extract the temporal related terms such as week days, dates, and time from the document content and then place the documents in the same time interval close

---

<sup>1</sup>Part of this chapter is published in [90]



Table 4.1: List of notations

$D$	Set of data points (documents).
$d_i \in D$	A data point (document)
$k$	Number of clusters
$\pi_i$	List of documents in cluster $i$
$\mu_i$	Centroid of cluster $i$
$n =  D $	Number of data points in $D$
$Dis(d_i, d_j)$	Distance between $d_i$ and $d_j$
$t_i$	Term $i$ in document-term matrix
$v_{t_i}$	Vector representation of term $t_i$ (the column $t_i$ of document-term matrix)
$v_{d_i}$	Vector representation of document $d_i$ (the row $d_i$ of document-term matrix)
$M$	The set of must-link constraints
$time(d_i, d_j)$	The creation time difference between two documents
$T_{d_i}$	The creation time of document $d_i$

to each other [66]. This needs several different Natural Language Processing steps such as Name Entity Detection and is beyond the scope of this research. In our experiments, we assume that each document has a creation time and based on that we calculate the temporal similarity between documents. This assumption will simplify the task and let us only focus on the impact of the temporal similarity on the result of the clustering. Let  $T_{d_f}$  and  $T_{d_l}$  be the creation time of the first and last document in the collection. We define the temporal distance between two documents in Eq. 4.1.

$$\mathbf{time}(d_i, d_j) = \frac{|T_{d_i} - T_{d_j}|}{|T_{d_f} - T_{d_l}|} \quad (4.1)$$

The time difference can be in days, months, weeks, or years and can be determined by the user and it is related to the characteristic of the dataset. We not only want to consider the temporal similarity but we also need to consider the content similarity. Based on this, we combine the the temporal and content similarity based on Eq. 4.2.

$$\mathbf{Dis}(d_i, d_j) = (1 - \lambda * \mathbf{time}(d_i, d_j)) * (1 - \mathbf{Cosine}(d_i, d_j)) \quad (4.2)$$

In Eq. 4.2,  $\lambda$  is a hyper parameter to balance between temporal and content similarity which can be adjusted by the user. This temporal-content similarity is inspired by the similarity introduced by M. Rizoiu et al. [83], which has been used for non-textual datasets. We made some amendments to make it appropriate for the textual datasets. The Cosine similarity between two documents is defined in Eq. 4.3.

$$\text{Cosine}(d_i, d_j) = \frac{d_i \cdot d_j}{\|d_i\| \|d_j\|} = \frac{\sum_{k=1}^v d_{it_k} d_{jt_k}}{\sqrt{\sum_{k=1}^v d_{it_k}^2} \sqrt{\sum_{k=1}^v d_{jt_k}^2}} \quad (4.3)$$

Table 4.2: Statistics of the Datasets used in the Temporal Similarity experiments.

Name	Type/Language	# Documents	# Categories	Categories
NewsCat	news/English	8018	5	Arts; Education Tech; Crime; Science
NewsCat2	news/English	8022	4	Religion; College; Travel; World News
BrazilNews	news/Portuguese	5749	4	Comida; TV; Folhinha; Turismo
BrazilNews2	news/Portuguese	4864	3	Ilustrissima; Ciencia; Educacao

In our experiments, we use two datasets, NewsCat and BrazilNews. The statistical information about these datasets is in Table 4.2. These datasets had the creation date for each document in terms of day, month and year and this is the reason we have chosen them. NewsCat dataset contains around 125k news headlines from 2013 to 2018 extracted from HuffPost website<sup>2</sup>. BrazilNews is a news dataset containing about 167K news articles gathered between January 2015 and September 2017 from Folha de São Paulo newspaper<sup>3</sup>. We have selected a subset of categories in these datasets.

We compare the performance of clustering based on different similarity measures in Table 4.3. Normalized Mutual Information (NMI), Homogeneity, and Adjusted Random Index (ARI) are used to evaluate the results. For each similarity, the KMeans algorithm has been initialized 100 times by random seeds. In order to have a fair comparison, the same set of seeds is used for each similarity method. In most cases, the combination of the temporal and content (Cosine) similarity leads to better clustering result. We also studied the impact of using only the temporal similarity. The temporal similarity results in Table 4.3 indicates that even by using different seeds the clustering algorithm will merge to the same set of clusters. It is an interesting result and it can be used to automatically determine the  $\lambda$  parameter of Eq. 4.2. For example, calculate the Silhouette score of the clusters when using only the temporal similarity and based on that regulate the  $\lambda$  parameter. The higher value of the Silhouette score the closer value of the  $\lambda$  to 1. For those datasets that the Temporal similarity by itself led to a good clustering result, the combination of it with the

<sup>2</sup><http://kaggle.com/rmisra/news-category-dataset> — last accessed: Oct. 16, 2018

<sup>3</sup><http://kaggle.com/marlesson/news-of-the-site-folhau01> — last accessed: Oct. 16, 2018

Table 4.3: The comparison between different similarity methods and their impact on the clustering performance. Normalized Mutual Information (NMI), Homogeneity, and Adjusted Random Index (ARI) are used to evaluate the results. For each similarity, the KMeans algorithm has been initialized 100 times by random seeds. The number after +/- indicates the standard deviation. In order to have a fair comparison, the same set of seeds is used for each similarity method. The values in parentheses indicate the average percentage of improvements over the Cosine similarity measure. In NewsCat and NewsCat2 the  $\lambda$  is set to 1 and for the BrazilNews and BrazilNews2 is set to 0.3 based on grid search approach.

Similarity	Measure	NewsCat	NewsCat2	BrazilNews	BrazilNews2
Cosine	NMI	0.36 +/- 0.05	0.32 +/- 0.07	0.31 +/- 0.07	0.45 +/- 0.12
	Homogeneity	0.36 +/- 0.05	0.31 +/- 0.07	0.3 +/- 0.08	0.4 +/- 0.12
	ARI	0.32 +/- 0.11	0.31 +/- 0.08	0.23 +/- 0.1	0.37 +/- 0.16
Temporal	NMI	0.12 +/- 0	0.3 +/- 0	0.08 +/- 0	0.01 +/- 0
	Homogeneity	0.12 +/- 0	0.3 +/- 0	0.08 +/- 0	0.01 +/- 0
	ARI	0.06 +/- 0	0.29 +/- 0	0.04 +/- 0	0.01 +/- 0
Cosine & Temporal	NMI	0.4 +/- 0.05 (+11.27%)	0.46 +/- 0.08 (+47.92%)	0.34 +/- 0.07 (+12.99%)	0.43 +/- 0.11 (-1.48%)
	Homogeneity	0.4 +/- 0.06 (+12.45%)	0.45 +/- 0.09 (+50.05%)	0.33 +/- 0.08 (+14.47%)	0.39 +/- 0.11 (-1.04%)
	ARI	0.37 +/- 0.11 (+17.89%)	0.45 +/- 0.11 (+53.47%)	0.26 +/- 0.09 (+20.04%)	0.35 +/- 0.14 (-0.9%)

content similarity (Cosine) was effective. For example in BrazilNews2, the temporal similarity had the worst performance and consequently there is no improvement when the temporal similarity and the Cosine similarity are combined.

We repeat the same set of experiments by using the DSKM algorithm for initializing the KMeans algorithm in Table 4.4. Results indicate that using the DSKM improved the performance result of the BrazilNews and BrazilNews2 datasets and for the NewsCat and NewsCat2 results did not improve. In the DSKM algorithm, we only need to initialize the clustering algorithm once.

#### 4.1.2 Interactive Temporal Document Clustering

It is possible to interact with the KMeans algorithm in different ways. One can consider the interactions as seed documents (similar to iKmeans) or assign a penalty in the objective function of the KMeans in case of violating the must-link or cannot link constraints [17, 32]. In this section, we will introduce a novel clustering algorithm that considers both user interactions and the temporal similarity between documents.

Table 4.4: The comparison between different similarity methods and their impact on the clustering performance. The DSKM method has been used to initialize the KMeans algorithm. The values in parentheses indicates the percentage improvements over the Cosine similarity measure.

Similarity	Measure	NewsCat	NewsCat2	BrazilNews	BrazilNews2
Cosine	NMI	0.35	0.321	0.389	0.403
	Homogeneity	0.334	0.289	0.394	0.377
	ARI	0.365	0.284	0.349	0.312
Cosine & Temporal	NMI	0.363 (+3.71%)	0.458 (+42.68%)	0.38 (-2.31%)	0.481 (+19.35%)
	Homogeneity	0.352 (+5.39%)	0.409 (+41.52%)	0.385 (-2.28%)	0.439 (+16.45%)
	ARI	0.39 (+6.85%)	0.41 (+44.37%)	0.301 (-13.75%)	0.446 (+42.95%)

The user interactions are over of key-terms. The key-term interactions will be mapped to must-link constraints and in case of violating them there will be a penalty. In our proposed method we assumed the following assumptions.

- The document-term matrix contains the normalized *tf-idf* value of terms inside each document.
- Interaction between the user and the clustering algorithm is based on key-terms. Only must-link constraints are considered. For example, the user wants to have two terms to be in a cluster.
- Let the user assign  $m$  key-terms to a cluster. The must-link terms will be converted to the equivalent must-link documents as Eq. 4.4 where  $\mathbb{1}$  is the indicator function and  $\Sigma$  is the sum of each element of two vectors.  $v_{t_i}$  is vector representation of term  $t_i$  (the column  $t_i$  of document-term matrix). Basically, the result of Eq. 4.4 is a set of documents that contains at least one of the user specified key-terms. In case that a document belongs to more than one set of must-links, the one that is more similar based on the Cosine similarity will be chosen.

$$\sum_{i=1}^m \mathbb{1}[v_{t_i} > 0] \quad (4.4)$$

- If a document is clustered differently from the designated user cluster, we assign a penalty. The penalty in case of violating a must-link constraint is considered

to be the distance between the centroid of the cluster that the document must be belongs to and the centroid of the new assigned cluster.

- In the first iteration of the algorithm, we assumed that there are no must-link constraints.

The KMeans algorithm has two major steps, assignment and update. In the assignment step, the distance between each document and cluster centroids is calculated first, then the documents are assigned to the closet cluster centroid. In the update step, the centroid vectors will be updated. In the main KMeans algorithm, the centroid of a cluster will be updated by calculating the element-wise average of all documents assigned to that cluster.

Let  $\alpha$  be a hyper parameter to regulate the penalty term of the objective function. Based on the above assumptions, we will define the objective function of the KMeans algorithm as Eq. 4.5.

$$\mathcal{J} = \sum_{j=1}^k \sum_{d_i \in \pi_j} \left[ \frac{1}{2}(\mu_j - d_i)^2 + \alpha \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} \frac{1}{2}(\mu_j - \mu_k)^2 \right] \quad (4.5)$$

Let  $(iter - 1)$  be the result of previous iteration. The Cluster assignment for document  $d_i$  is:

$$\operatorname{argmin}_{j=1, \dots, k} \left[ \frac{1}{2}(\mu_j^{(iter-1)} - d_i)^2 + \alpha \sum_{\substack{d_k \notin \pi_j^{(iter-1)} \\ (d_k, d_i) \in M}} \frac{1}{2}(\mu_j^{(iter-1)} - \mu_k^{(iter-1)})^2 \right] \quad (4.6)$$

Based on Eq. 4.6, a document will be assigned to the closest cluster centroid. The role of the penalty term is forcing a document not to be clustered differently from the cluster the user assigned to it. As described before, the user can assign documents to clusters indirectly by assigning a set of key-terms for the clusters. In case that there exists a cluster centroid such that a document is closer to it than the the user specified cluster centroid, the penalty term in Eq. 4.6 is playing the decision making role.

The next step is updating cluster centroids. Let  $L$  be:

$$L = \sum_{d_i \in \pi_j} \left[ \frac{1}{2}(\mu_j - d_i)^2 + \alpha \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} \frac{1}{2}(\mu_j - \mu_k)^2 \right] \quad (4.7)$$

Optimizing  $L$  will optimize the  $\mathcal{J}$  as well. Based on that, the cluster center ( $j$ ) will be updated as the following:

$$\begin{aligned} \frac{\partial L}{\partial \mu_j} = 0 &\Rightarrow \frac{\partial \left( \sum_{d_i \in \pi_j} \left[ \frac{1}{2}(\mu_j - d_i)^2 + \alpha \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} \frac{1}{2}(\mu_j - \mu_k)^2 \right] \right)}{\partial \mu_j} = 0 \\ &\Rightarrow \sum_{d_i \in \pi_j} \left[ \frac{1}{2} \frac{\partial(\mu_j - d_i)^2}{\partial \mu_j} + \alpha \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} \frac{1}{2} \frac{\partial(\mu_j - \mu_k)^2}{\partial \mu_j} \right] = 0 \\ &\Rightarrow \sum_{d_i \in \pi_j} (\mu_j - d_i) + \alpha \sum_{d_i \in \pi_j} \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} (\mu_j - \mu_k) = 0 \\ &\Rightarrow |\pi_j| \mu_j - \sum_{d_i \in \pi_j} d_i + \alpha |\pi_j| \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} \mu_j - \alpha \sum_{d_i \in \pi_j} \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} \mu_k = 0 \\ &\Rightarrow |\pi_j| \mu_j + \left( \alpha \sum_{\substack{d_i \in \pi_j \\ d_k \notin \pi_j}} \mathbb{1}[(d_k, d_i) \in M] \right) \mu_j - \sum_{d_i \in \pi_j} d_i - \alpha \sum_{d_i \in \pi_j} \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} \mu_k = 0 \end{aligned}$$

Let  $Y_j$  be:

$$Y_j = |\pi_j| + \alpha \sum_{\substack{d_i \in \pi_j \\ d_k \notin \pi_j}} \mathbb{1}[(d_k, d_i) \in M] \quad (4.8)$$

Then:

$$\mu_j = \frac{\sum_{d_i \in \pi_j} d_i}{Y_j} + \frac{\alpha \sum_{d_i \in \pi_j} \sum_{\substack{d_k \notin \pi_j \\ (d_k, d_i) \in M}} \mu_k}{Y_j} \quad (4.9)$$

Based on Eq. 4.9, a cluster centroid is updated by the sum of all assigned documents vectors plus the summation of the centroid vectors of violated constraints. This value is normalized by the number of assigned documents and the number of times that a must-link constraint has been violated.

---

**Algorithm 4:** Proposed Interactive Temporal Document Clustering Algorithm

---

```

1 if firstIteration then
2   | provide and visualize initial clusters for the user;
3 else
4   | get user feedback (key-terms) for each cluster;
5   | generate must-link document constraints using Eq. 4.4;
6   | while !Converged do
7     | assign data points to each cluster using Eq. 4.6;
8     | update centroid of clusters based on Eq. 4.9;
9   | end
10  | visualize the clusters for the user;
11 end

```

---

The overall pseudocode of the proposed interactive temporal document clustering algorithm is explained in Algorithm 4. In the first iteration, the clustering algorithm provides the initial result for the user. It is possible to use the DSKM algorithm explained in section 3.2 for finding the initial seed documents. In the second step, the user can provide feedback for each cluster by assigning a set of key-terms for each cluster. The key-term interactions will be converted to a set of must-link documents using Eq. 4.4. In the next step, each data point will be assigned to the closest cluster based on Eq. 4.6. Finally, the centroid of clusters will be updated using Eq. 4.9. The data point assignment and centroids update steps will be executed iteratively until the clustering algorithm converges to a solution. The algorithm is converged when the label of clusters does not change from the previous iteration. This way of solving an optimization function is usually called Coordinate descent [103].

## Evaluation

In order to evaluate the proposed interactive clustering algorithm, first we ran the algorithm without any interactions, then we initialized the algorithm with the centroids of output clusters and a set of keywords as an interaction. This is exactly the same procedure when an end users want to interact with an interactive clustering algorithm. In order to find the relevant keywords for each cluster we used the Chi Square test using the true label of each document. For each cluster we selected a single keyword. In this way, we can simulate the interactions with the clustering algorithm using these keywords.

Table 4.5: Comparing the Accuracy of the proposed key-term based interactive clustering algorithm. The average of 50 runs of the algorithm by random seeds is reported. The number after +/- indicates the standard deviation. In order to have the fair comparison, the same set of seeds is used for each similarity method. See Table 3.1 for more information about Newsgroup5 dataset.

Dataset	Similarity	Before Interaction	After Interaction	Improvement	Alpha	keywords
Newsgroup5	Cosine	0.706 +/- 0.09	0.719 +/- 0.09	+2.09%	0.15	chastity; zoo; aramis; georgia; politics
NewsCat	Cosine	0.434 +/- 0.06	0.439 +/- 0.09	+1.51%	0.1	art; police; education; teachers; apple
NewsCat	Cosine & Temporal	0.458 +/- 0.03	0.459 +/- 0.03	+0.28%	0.05	art; police; education; teachers; apple
NewsCat2	Cosine	0.430 +/- 0.04	0.440 +/- 0.05	+2.41%	0.1	college; pope; korea; travel
NewsCat2	Cosine & Temporal	0.532 +/- 0.04	0.547 +/- 0.05	+3.38%	0.4	college; pope; korea; travel

The comparison result of the accuracy of the proposed method on different datasets is provided in Table 4.5. The clustering quality is measured by Accuracy [22]. To match the predicted labels with actual labels for calculating the accuracy, we used the Hungarian method [53]. The reason that we selected the accuracy is to prevent the impact of the order of key-terms on evaluation results. Because we are simulating the interactivity we do not know in which order to assign the keyword to each cluster. For the sack of simplicity, in Eq. 4.5, we considered  $(d_k, d_i) \in M$  where  $k = i$ . We evaluated the proposed method with and without considering the temporal similarity. The result shows that in most cases the combination of the key-term interaction and the temporal similarity leads to the best clustering performance result. For the NewsCat dataset, the improvement of using both temporal and key-term interaction



does not significantly change the result. It means that a single key-term is not enough for this dataset and more interaction is needed.

## 4.2 Incorporating Temporal Aspect in the Visualization

In this section, we explain how to consider the temporal aspect in the visualization and finally evaluated the usefulness of visual components by conducting a case study. In the case study we use the combination of the DSKM algorithm (Chapter 3) and the iKMeans algorithm (Chapter 5) as the clustering algorithm.

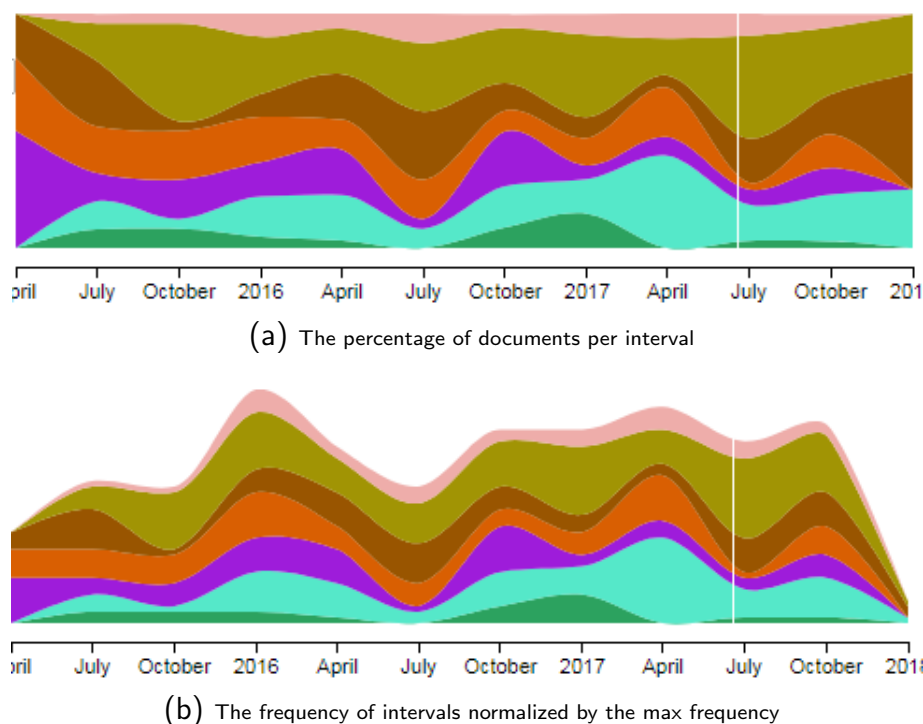


Figure 4.1: The two modes of the Temporal view. The user can switch between these modes on demand.

### 4.2.1 Visualization Modules

**Sentence Cloud View** To make the reading of the content of documents faster, we decide to introduce the Sentence Cloud instead of a plain display of document textual content. The idea of the sentence cloud is highlighting the most important sentences of the document by changing the font size. The more important sentences will have a larger font size and consequently, the user can quickly skim the document

content and find out directly its most important parts. In order to create the Sentence Cloud, first, we use a sentence tokenizer and extracted the sentences of the document. Second, we aggregate the *tf-idf* value of terms of each sentence from the document-term matrix. Based on the sentence score, we assign the HTML font size from 1 to 7 to each sentence of the document.

**Temporal View** It is important and beneficial to depict the clusters' change over time. We use a ThemeRiver [42] style visualization for illustration of changes. First, the document collection is divided into equal time intervals. The time intervals are on a monthly basis between 1 to 6 months and the user can change the interval in real time. Second, we count the number of documents belonging to each cluster in each interval. This frequency is used to initialize the ThemeRiver in two modes. The first mode shows the percentage of documents per interval (Fig. 4.1a), and the second one is the frequency of intervals normalized by the max frequency (Fig. 4.1b). In the first mode, the user can find out what percentage of the document clusters is changing over each time interval and in the second one, the user can see cluster evolution based on the maximum frequency of the number of documents in all time intervals. Monotonic interpolation is used to smoothly connect intervals together. The color of each layer of ThemeRiver is the same as the color of the associated cluster. Whenever the user hovers over each cluster in each time interval the list of top terms of the documents in that cluster and interval will be shown to the user. These top terms are extracted by Eq. 2.13. This view is for datasets containing documents that include their creation time.

#### 4.2.2 Case Study

In this section, we explain the details of a case study conducted by domain experts. The case study employs the system for evidence-based decision making over email conversations in the field of Respiratory Therapy. The different between the Case study with the user study is that, in the case study we ask domain experts (usually one or two) about their opinions while in the user study, the number of participants are much higher and they do not necessary need to be domain expert.

## Case Study Introduction

Email is one of the popular formal communication methods used by healthcare professionals. Often, healthcare providers belonging to the same community join list-servers to share their questions, ideas, and concerns about different topics. The tacit knowledge in the email conversations complements formal textbooks and research publications. Current email applications are not adequate for efficiently exploring, retrieving, and acquiring knowledge from past email conversations. The goal of this case study is to employ the proposed interactive document clustering system for visual analytics of listserver content. We apply our method to the real world dataset belonging to the Canadian Society of Respiratory Therapists. The dataset consists of email conversations among members of the Respiratory Therapist community who subscribe to the mailing list since 2015. First, we introduce how we prepared and processed the dataset. Second, we report the result of the case study conducted on the respiratory therapy dataset. Besides the cases previously analyzed in [92], the study shows that, with proper consideration for the vocabulary involved, the framework is useful even for more complex data sets such as email.

## Dataset Preparation

First, we extract email subject, sender name, date, and body, then sorted them by email date. The timezone differences were considered while sorting email messages. This process resulted in 1058 email messages from January 2015 to January 2018. Second, we merge emails with the same subject title in chronological order and created email threads. Email messages with the same subject but not in the same time period are considered as forming new email threads. Each email thread is considered as a single document and saved by the name of its first email message date concatenated by its subject. Third, the repetitive parts of emails included as a result of replying or forwarding the original email are removed based on a rule-based approach. Fourth, default text messages after the emails signature, such as the organization's privacy rules or the listserv regulations, were removed. While the text inside the signature section of emails contains useful information, for the process of creating the document-term matrix, we do not consider the signatures. The signature can make unrelated email messages similar if they have been sent by the same person or from the same

organization.

A simple, but effective, heuristic was used to distinguish the signature from the body of an email. First, we create a different combination of the email sender name such as just the first or last name, last name followed by the first name or first and last name together. Second, we spot these names in the email message. If we spot more than one name combination, the one which is longer and have the larger offset index value is selected. Finally, the text followed by the targeted name was considered as the email signature. The reason that this heuristic is effective is that most email messages, especially emails in the scientific listservers, end with the sender name and it is unusual to have the sender name inside the body of the email message.

The language of the email messages was mostly formal and had very few spelling and grammatical errors. Abbreviations are common in most of the emails, so we expand the abbreviations by using reference textbooks about Respiratory Therapy. The most frequent sense is used for disambiguating the abbreviations with the same surface form. In order to create the document term matrix, we decided to use the controlled vocabulary approach. The reason is that there are several greeting and signature common words in email messages that make unrelated email threads look similar to each other based on the bag of words model. The controlled vocabulary was created from the index and glossary section of several reference Respiratory Therapy textbooks. To expand the vocabulary, we used named entity recognition<sup>4</sup> (NER) to extract organization, persons, tools, locations, and drug names. The vocabulary contains single-word as well as multi-word terms (noun phrases). All terms that appeared in the controlled vocabulary, were extracted by the NER method and used to create the final document-term matrix. The reason we prepare the document-term matrix in this way is to support evidence-based decision making. Domain-specific terminologies are very helpful for this task.

## Case Study Results

The case study was conducted by a Registered Respiratory Therapist (RRT) with more than 20 years of experience and a faculty member of a Canadian University. The user started with 8 clusters. Then, by looking at the Cluster view, and based on

---

<sup>4</sup><https://spacy.io/> - last access: Oct. 16, 2018

Table 4.6: Top terms of final clustering result of the email list dataset. Each line shows the top related phrase or term of each cluster.

laryngectomy emergency tracheostomy algorithms airway	fabian sipap arabella biphasic efficacy	end tidal co2 tidal end capnography monitoring	flow nasal high flow nasal cannula humidifier delivered	program pft lab pulmonary function
days staffing hospital tertiary following	methacholine challenge particle conducting methacholine challenge testing	filters filter expiratory occlusion acid	invasive non non invasive ventilation niv ventilation	sputum sputum induction induction saline hypertonic
nitric nitric oxide oxide inhaled nitric oxide pregnant	birth born congenital decelerations defects	sedation procedural suite rrt endoscopy	balloon esophageal balloon esophageal procedure peep	ett etts endotracheal tube securing endotracheal
respiratory respiratory therapy health position therapy	bronchoscope cleaning mdr scope point	machine circuit gas tubing bag	aggregate delivery room institutional institutional review board irb	arterial line insertion lines art

her expert knowledge, she increased the number of clusters to 20. The new clustering result was more satisfying to her but she decided to increase the number of clusters to 40 to find out if there would be finer clusters. The clustering result with 40 clusters was not as satisfying as the clustering with 20 clusters so she switched back to 20 clusters. The deterministic feature of the system was very helpful for the user to reproduce the previous clustering result. During the case study, the user searched the term “CPAP” and the related documents were filtered in the Graph view. The result of this search not only helped the user retrieve the related documents, but she was able to find the most similar cluster to this term. This feature enabled the user to identify similar topics and email threads that may not contain “CPAP” but are actually related to the user query. The user found the list of documents saved by the name of their related email thread in the sentence cloud very helpful and informative.

Emails with similar subjects are grouped as a single thread and sorted according to the time they were sent. They are preprocessed by removing the repetitive content from forwarded or replied emails, and the sender’s signature. These features helped the user to find her required content quickly and efficiently.

By looking at the Temporal view, the user found out that January and September are two months that have the highest number of email conversations. It may be related to the start of the academic semesters. The Temporal view also showed the

same email rates per year. Some clusters and topics can be observed in most time intervals while some of the clusters emerged or faded out in different time intervals. The user found it useful to be able to change the time intervals in the Temporal view. Based on this feature, the user is able to change the interval from 1 month to 6 months.

The overall final clustering result of the Canadian Society of Respiratory Therapy email list is summarized in Table 4.6. The final clustering result was very convincing and informative to the user. The user mentioned that, by using this system, she was able to faster and better retrieve the related email threads about a specific topic than the traditional search in email applications. This is an important feature in support of the process of evidence-based decision making. We asked another domain expert who was head of a Respiratory Therapy department in a Canadian University to give his idea about the results of Table 4.6 and he confirmed that the results in this table are meaningful and valuable to understand the major topics in email conversations. During the case study, we automatically logged every operation that the expert conducted (see Table 4.7). We did not inform the expert about logging to prevent behavior change of the expert. The most frequent module that the expert focused on was the Graph view. The Cluster view and temporal view are in the second and third place. The quality of clustering based on the Silhouette score after each re-clustering request is reported in Table 4.8. The low Silhouette score of the clustering indicates that there is a very diverse set of topics in the email conversations and as the number of clusters increases, the score is also increased. The heat map of the expert clicks and mouse movements during the case study is depicted in Fig. 4.2.

Table 4.7: Most frequent operations that the expert conducted during the case study. During the case study we automatically logged every interaction of the user with the system.

Operation description	Module name	Percent
Hover a mouse on the Graph view nodes	Graph view	36.3
Click on a cluster in Cluster view	Cluster view	12.79
Hover mouse over Temporal view	Temporal view	11.19
Highlighting documents containing the selected term in Graph view	Cluster view and Cluster Key-terms view	8.45
Changing the Cosine distance threshold of a Graph view	Graph view	5.25
A node inside Graph view is clicked	Graph view	4.57
A term inside Cluster view is clicked	Cluster view	3.88

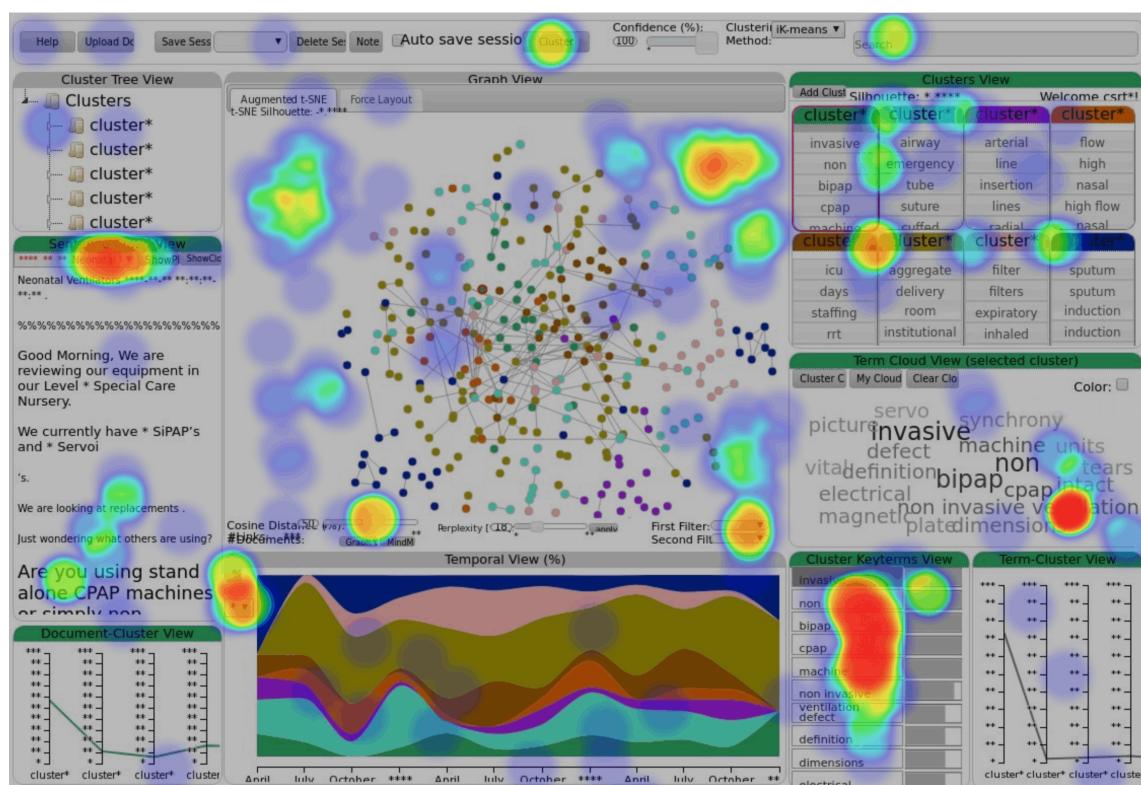


Figure 4.2: The heat map of the expert clicks and mouse movements. The red areas are most frequently clicked areas. The expert mostly focused on the Sentence view, the Graph view, Cluster key-term view and Cluster view. The clicks on the white area of the Graph view are cases that the expert decided to de-select nodes in the graph.

### Expert Interview:

The case study was conducted in about 70 minutes, including 20 minutes introducing the system to the expert, 40 minutes conducting the case study, and 10 minutes interviewing the expert. We summarize the interview result into three categories.

**Clustering algorithm** The expert was impressed by the accuracy of the clustering algorithm but she needed a better mechanism to find the proper number of clusters as she commented “*some trial and error is required in order to determine the ideal number of clusters required in order to achieve a relevant clustering*”

**Interactive visualization** All in all, the expert was satisfied by the interactive visualization and found most of the modules useful in practice. The expert found the

Table 4.8: Clustering Silhouette after each re-clustering request from the expert.

Evaluation measure	Re-clustering 1 (8 Clusters)	Re-clustering 2 (40 Clusters)	Re-clustering 3 (20 Clusters)
Clustering Silhouette	0.0125	0.0582	0.0472

Sentence view as one of the most practical ones but she preferred to have the same font size for all portions of an email thread. She commented about the Temporal view that she needs to conduct some trial and error to optimally visualize what topics/clusters were most prominent during specific time intervals. The expert was uncertain about the functionality of the relationship between clusters in the Graph view in this Case study. Finally, the search bar in the interactive visualization was very helpful for the expert to search for all e-mail threads related to a specific topic or theme.

**Overall goal** We quote the expert response for her goal of conducting the case study and the result she achieved in the following. “*The objective of this project [Case Study] was to extract from a Listserv, topics, and trends related to respiratory therapy practice in Canada. The interactive clustering tool that was created enabled the user to easily find e-mail threads related to specific topics, see clustered topics and identify discussion trends during the last four years.*”

### 4.3 Conclusion

In this chapter, we have incorporated the temporal aspect in the procedure of clustering. First, a temporal based similarity measure is introduced. The experimental results indicate that the combination of the temporal and content similarity can lead to better clustering in most of the datasets. Second, a novel temporal interactive document clustering algorithm is explained in detail. In the proposed method, we have added a penalty term to the objective function of the KMeans and have considered the key-terms as generating must-link constraints. The result indicates that the proposed method can apply the user preferences to clustering and consequently improve the quality of the clusters. Third, we have explained how to consider the temporal aspect in the visualization. Finally, we have discussed the case study conducted on real world dataset belonging to the Canadian Society of Respiratory Therapists.



## Chapter 5

### Wikipedia Based Semantic Similarity

In addition to involving human in the process of clustering, considering semantic similarity can improve the result of document clustering. This idea is mostly studied in this chapter<sup>1</sup> where we introduce a method for representing Wikipedia concepts with low dimensional vectors. These Wikipedia concept vectors can be used for linking concepts of documents to the related Wikipedia page. In that case, each document can be represented by a bag of concepts. The higher quality of document clustering is reported by considering the bag of concepts [44]. In the first step, we use word embedding methods for representing Wikipedia concepts with a vector. These vectors can be used in the process of disambiguation and consequently help to better quality bag of concepts. The bag of concepts can be used similar to bag of words in the process of clustering.

Using neural network model for different machine learning tasks such as word embedding has recently gained a lot of researchers' attention. Word embedding is the task of mapping words or phrases to a low dimensional numerical vector. In this chapter, we use skip-gram model to embed Wikipedia concepts and entities. The English version of Wikipedia contains more than five million pages, which suggest its capability to cover many English entities, phrases, and concepts. Each Wikipedia page is considered as a concept. Some concepts correspond to entities, such as a person's name, an organization or a place. Contrary to word embedding, Wikipedia concepts embedding is not ambiguous, so there are different vectors for concepts with similar surface form but different mentions. The results show that the proposed approaches have the performance comparable and in some cases even higher than the state-of-the-art methods.

---

<sup>1</sup>Part of this chapter is published in [91]

Table 5.1: Top similar terms to “amazon” based on Word2Vec and GloVe.

<b>Word2Vec</b>	itunes	play.com	cli	adobe_acrobat	amiga	canada
<b>GloVe</b>	amazon.com	rainforest	amazonian	kindle	jungle	deforestation

## 5.1 Vector Embedding of Wikipedia Concepts

### 5.1.1 Introduction

Recently, many researchers [69, 78] showed the capabilities of deep learning for natural language processing tasks such as word embedding. Word embedding is the task of representing each term with a low-dimensional (typically less than 1000) numerical vector. Distributed representation of words showed better performance than traditional approaches for tasks such as word analogy [69]. Some words are Entities, i.e. name of an organization, Person, Movie, etc. On the other hand, some terms and phrases have a page or definition in a knowledge base such as Wikipedia, which are called concepts. For example, there is a page in Wikipedia for Data Mining or Computer Science concepts. Both concepts and entities are valuable resources for getting semantic and better sense making of a text. In this chapter, we used deep learning to represent Wikipedia concepts and entities with numerical vectors. We make the following contributions in this chapter:

- Wide coverage of words and concepts: about 1.7 million Wikipedia concepts and nearly 2 million English words were embedded in this research, which is one of the highest numbers of embedded concepts that currently exists, to the best of our knowledge. The concept and words vectors are also publicly available for research purposes<sup>2</sup>. We also used one of the latest versions of the Wikipedia English dump to learn word embedding. Over time, each term may appear in different contexts, and as a result, it may have different embeddings so this is why we used one of the recent versions of Wikipedia.
- Unambiguous word embedding: Existing word embedding approaches suffer from the problem of ambiguity. For example, top nine similar terms to ‘Amazon’ based on pre-trained Google’s vectors in Word2Vec [69] and GloVe [78] models

---

<sup>2</sup><https://github.com/ehsansherkat/ConVec>

are in Table 5.1. Word2Vec and GloVe are the two first pioneer approaches for word embedding. In a document, 'Amazon' may refer to the name of a jungle and not the name of a company. In the process of embedding, all different meanings of the word 'Amazon' are embedded in a single vector. Producing distinct embedding for each sense of the ambiguous terms could lead to better representation of documents. One way to achieve this is using unambiguous resources such as Wikipedia and learning the embedding separately for each entity and concept.

- We compared the quality versus the size of the corpus on the quality of trained vectors. We demonstrated that much smaller corpora with more accurate textual content is better than very large text corpora with less accuracy in the content for the concept and phrase embedding.
- We studied the impact of fine-tuning weights of network by pre-trained word vectors from very large text corpora in tasks of Phrase Analogy and Phrase Similarity. Fine-tuning is the task of initializing the weights of the network by pre-trained vectors instead of random initialization.
- Proposing different approaches for Wikipedia concept embedding and comparing results with the state-of-the-art methods on the standard datasets.

### 5.1.2 Related Work

Word2Vec and GloVe are two pioneer approaches for word embedding. Recently, other methods have been introduced that try to improve both the performance and quality of the word embedding [35] by using multilingual correlation. A method based on Word2Vec is proposed by Mikolov et al. for phrase embedding. [69]. In the first step, they find the words that appear more frequently together than separately, and then they replace them with a single token. Finally, the vector for phrases is learned in the same way as single word embedding. One of the features of this approach is that both words and phrases are in the same vector space.

Graph embedding methods [24] using Deep Neural Networks are similar to the goals of this paper. Graph representation has been used for information management in many real world problems. Extracting deep information from these graphs is

important and challenging. One solution is using graph embedding methods. The word embedding methods use linear sequences of words to learn a word representation. For graph embedding, the first step is converting the graph structure to an extensive collection of linear sequences. Perozzi presented a uniform sampling method named Truncated Random Walk for converting the graph structure to a linear sequences [79]. In the second step, a word embedding method such as Word2Vec is used to learn the representation for each graph vertex. Wikipedia can also be represented by a graph, and the links are the inter citation between Wikipedia’s pages, called anchors.

A graph embedding method for Wikipedia using a similarity inspired by the HITS algorithm [52] is presented in [89]. The output of this approach for each Wikipedia concept is a fixed length list of similar Wikipedia pages and their similarity score, which represents the dimension name of the corresponding Wikipedia concepts. The difference between this method and deep learning based methods is that each dimension of a concept embedding is meaningful and understandable by the human.

Milne and Witten [71] proposed a Wikipedia concept similarity index based on in-links and out-links of a page. In their similarity method, two Wikipedia pages are more similar to each other if they share more common in- and out-links. This method is used to compare the result of the Concept Similarity task with the proposed approaches.

The idea of using Anchor texts inside Wikipedia for learning phrase vectors is being used in some other research [98, 23] as well. In this research, we proposed different methods to use anchor texts and evaluated the results in standard tasks. We also compared the performance of the proposed methods with top notch methods.

The link structure (citation) between scientific papers is used in *cite2vec* [14] to help users better exploring such documents. In this system, all the citations in documents are replaced by a unique identifier, and it then jointly learned a semantic embedding of words and documents. As a result of this step, each pair of word and document has a single vector in the same vector space. In the interface, the documents are depicted in a scatter plot and on top of that the most similar term for each document. The user can filter the scatter plot by providing desired keywords and as a result, all similar terms and documents will be highlighted in the plot.

### 5.1.3 Distributed Representation of Concepts

From this point on, we describe how we trained our word embedding. At first we describe the steps for preparing the Wikipedia dataset and then describe different methods we used to train words and concepts vectors.

**Preparing Wikipedia dataset:** In this research, the Wikipedia English text from the Wikipedia dump on May 01, 2016 is used. In the first step, we developed a toolkit<sup>3</sup> using several open source Python libraries (described in Appendix A) to extract all pages in English Wikipedia, and as a result 16,527,332 pages were extracted. Not all of these pages are valuable, so we pruned the list using several rules (check Appendix B for more information).

As a result of pruning, 5,001,168 unique Wikipedia pages, pointed to by the anchors, were extracted. For the next step, the plain text of all these pages was extracted in such a way that anchors belonging to the pruned list of Wikipedia pages were replaced (using developed toolkit) with their Wikipedia page ID (the redirects were also handled), and for other anchors, their surface form was substituted. We merged the plain text of all pages in a single text file in which each line is a clean text of a Wikipedia page. This dataset contains 2.1 billion tokens.

**ConVec:** The Wikipedia dataset obtained as a result of previous steps was used for training a Skip-gram model [69] with negative sampling instead of hierarchical softmax. We called this approach ConVec. The Skip-gram model is a type of Artificial Neural Network, which contains three layers: input, projection, and output. Each word in the dataset is inputted to this network, and the output is a prediction of the surrounding words within a fixed window size. We used a window size of 10 because we been able to get higher accuracy based on this window size. Skip-gram has been shown to give a better result in comparison to the Bag of Words (CBOW) model [69]. CBOW gets the surrounding words of a word and tries to predict the word (the reverse of the Skip-gram model).

As a result of running the Skip-gram model on the Wikipedia dataset, we got 3,274,884 unique word embeddings, of which 1,707,205 are Wikipedia concepts. Words

---

<sup>3</sup><https://github.com/ehsansherkat/ConVec>

and Anchors with a frequency of appearance in Wikipedia pages less than five are not considered. The procedure of training both words and concepts in the same time, results in that concepts and words belonging to the same vector space. This feature enables not only finding similar concepts to a concept but also finding similar words to that concept.

**ConVec Fine-Tuned:** In image processing approaches, it is customary to fine-tune the weights of a neural network with pre-trained vectors over a large dataset. Fine-tuning is the task of initializing the weights of the network by pre-trained vectors instead of random initialization. We tried to investigate the impact of fine-tuning the weights for textual datasets as well. In this case, we tried to fine-tune the vectors with GloVe 6B dataset trained on Wikipedia and Gigaword datasets [78]. The weights of the the skip-gram model initialized with GloVe 6B pre-trained word vectors and then the training continued with the Wikipedia dataset prepared in the previous step. We called the concept vectors trained based on this method ConVec Fine-Tuned.

Table 5.2: Comparing the results of three different versions of ConVec (trained on Wikipedia 2.1B tokens) with Google Freebase pre-trained vectors over the Google-100B-tokens news dataset in the Phrase Analogy task. The Accuracy (All) shows the coverage and performance of each approach for answering questions. The accuracy for common questions (Accuracy (Commons)) is for fair comparison of each approach. #phrases shows the number of top frequent words of each approach that are used to calculate the accuracy. #found is the number of questions where all 4 words are present in the approach dictionary.

Embedding Name	#phrases	Accuracy (All)		Accuracy (Commons)	
		#found	Accuracy	#found	Accuracy
Google Freebase	Top 30,000	1048	55.7%	89	52.8%
	Top 300,000	1536	47.0%	800	48.5%
	Top 3,000,000	1838	42.1%	1203	42.7%
ConVec	Top 30,000	202	81.7%	89	82.0%
	Top 300,000	1702	68.0%	800	72.1%
	Top 3,000,000	2238	56.4%	1203	61.1%
ConVec (Fine-Tuned)	Top 30,000	202	80.7%	89	79.8%
	Top 300,000	1702	68.3%	800	73.0%
	Top 3,000,000	2238	56.8%	1203	63.6%
ConVec (Heuristic)	Top 30,000	242	81.4%	89	80.9%
	Top 300,000	1804	65.6%	800	68.9%
	Top 3,000,000	2960	46.6%	1203	58.7%

**ConVec Heuristic:** We hypothesize that the quality of concept vectors can improve with the size of training data. The sample data is the anchor text inside each Wikipedia page. Based on this assumption, we experimented with a heuristic to increase the number of anchor texts in each Wikipedia page. It is a Wikipedia policy that there is no self-link (anchor) in a page. It means that no page links to itself. On the other hand, it is common that the title of the page is repeated inside the page. The heuristic is to convert all exact mentions of the title of a Wikipedia page to anchor text with a link to that page. By using this heuristic, 18,301,475 new anchors were added to the Wikipedia dataset. This method is called ConVec Heuristic.

**ConVec Only-Anchors:** The other experiment is to study the importance and role of the non-anchored words in Wikipedia pages in improving the quality of phrase embeddings. In that case, all the text in a page, except anchor texts were removed and then the same Skip-gram model with negative sampling and the window size of 10 is used to learn phrase embeddings. This approach (ConVec Only-Anchors) is similar to ConVec except that the corpus only contains anchor texts.

An approach called Doc2Vec was introduced by Mikolov et al. [56] for Document embedding. In this embedding, the vector representation is for the entire document instead of a single term or a phrase. Based on the vector embeddings of two documents, one can check their similarity by comparing their vector similarity (e.g. using Cosine distance). We tried to embed a whole Wikipedia page (concept) with its content using Doc2Vec and then consider the resulting vector as the concept vector. The results of this experiment were far worse than the other approaches so we decided not to compare it with other methods. The reason is mostly related to the length of Wikipedia pages. As the size of a document increases, the Doc2Vec approach for document embedding results in a lower performance.

#### 5.1.4 Evaluation

Phrase Analogy and Phrase Similarity tasks are used to evaluate the different embedding of Wikipedia concepts. In the following, detailed results of this comparison are provided.

**Phrase Analogy Task:** To evaluate the quality of the concept vectors, we used the phrase analogy dataset in [69] which contains 3,218 questions. The Phrase analogy task involves questions like “*Word1* is to *Word2* as *Word3* is to *Word4*”. The last word (*Word4*) is the missing word. Each approach is allowed to suggest the one and only one word for the missing word (*Word4*). The accuracy is calculated based on the number of correct answers. In word embedding the answer is finding the closest word vector to the Eq. 5.1.  $V$  is the vector representation of the corresponding Word.

$$V_{Word2} - V_{Word1} + V_{Word3} = V_{Word4} \quad (5.1)$$

$V$  is the vector representation of the corresponding Word. The cosine similarity is used for majoring the similarity between vectors on each side of the above equation.

In order to calculate the accuracy in the Phrase Analogy, all four words of a question should be present in the dataset. If a word is missing from a question, the question is not included in the accuracy calculation. Based on this assumption, the accuracy is calculated using Eq. 5.2.

$$Accuracy = \frac{\#CorrectAnswers}{\#QuestionsWithPhrasesInsideApproachVectorsList} \quad (5.2)$$

We compared the quality of three different versions of ConVec with Google Freebase<sup>4</sup> phrase vectors pre-trained over the Google-100B-token news dataset. The Skip-gram model with negative sampling is used to train the vectors in Google Freebase. The vectors in this dataset have 1000 dimensions in length. For preparing the embedding for phrases, the authors used a statistical approach to find words that appear more together than separately and then considered them as a single token. In the next step, they replaced these tokens with their corresponding freebase ID. Freebase is a knowledge base containing millions of entities and concepts, mostly extracted from Wikipedia pages.

In order to have a fair comparison, we reported the accuracy of each approach in two ways in Table 5.2. The first accuracy is to compare the coverage and performance of each approach over all the questions in the test dataset (Accuracy All). Based on the training corpus and the frequency of each word vector inside the corpus, each

---

<sup>4</sup><https://code.google.com/archive/p/word2vec>



Table 5.3: Comparing the results in Phrase Similarity dataset. Rho is Spearman’s correlation to the human evaluators. !Found is the number of pairs not found in each approach dataset. The average scores are weighted average score of the approach for each of the datasets. The weights are number of found pairs for each dataset.

Datasets			Wikipedia Miner		Google Freebase		ConVec		ConVec (Heuristic)	
#	Dataset Name	#Pairs	!Found	Rho	!Found	Rho	!Found	Rho	!Found	Rho
1	WS-REL [36]	251	114	0.6564	87	0.3227	104	0.5594	57	0.5566
2	SIMLEX [43]	961	513	0.2166	369	0.1159	504	0.3406	357	0.2152
3	WS-SIM [36]	200	83	0.7505	58	0.4646	81	0.7524	41	0.6101
4	RW [64]	1182	874	0.2714	959	0.1777	753	0.2678	469	0.2161
5	WS-ALL [36]	349	142	0.6567	116	0.4071	136	0.6348	74	0.5945
6	RG [80]	62	35	0.7922	14	0.3188	36	0.6411	25	0.5894
7	MC [70]	28	15	0.7675	9	0.3336	16	0.2727	12	0.4706
8	MTurk [40]	283	155	0.6558	123	0.5132	128	0.5591	52	0.5337
-	Average	414	241	0.4402	217	0.2693	219	0.4391	136	0.3612

approach is able to answer a different subset of questions from the list of all questions inside the phrase analogy dataset. An approach can answer a question if all four words of a question are present in the dataset. For example, the ConVec base model is able to answer 2,328 questions out of 3,218 questions of the phrase analogy dataset for the top 3,000,000 phrases. The second accuracy is to compare the methods over only common questions (Accuracy commons). Common questions are the subset of questions where all four approaches in Table 5.2 are able to answer them.

Each approach tries to answer as much as possible the 3,218 questions inside the Phrase Analogy dataset in *Accuracy-for-All* scenario. For the top 30,000 frequent phrases, Google Freebase was able to answer more questions, but for the top 3,000,000 frequent phrases ConVec was able to answer more questions with higher accuracy. Fine-tuning of the vectors does not have impact on the coverage of ConVec; this is why the #found is similar to the base model. We used the Wikipedia ID of a page instead of its surface name. The heuristic version of ConVec has more coverage to answering questions in comparison with the base ConVec model. The accuracy of the heuristic ConVec is somehow similar to the base ConVec for the top 300,000 phrases, but it will drop down for the top 3,000,000. It seems that this approach is efficient to increase the coverage without significantly sacrificing the accuracy, but probably it needs to be more conservative by adding more regulations and restrictions in the process of adding new anchor texts.

Only common questions between each method are used to compare the *Accuracy-for-Commons* scenario. The results in the last column of Table 5.2 show that the fine-tuning of vectors does not have a significant impact on the quality of the vectors embedding. The result of the ConVec Heuristic for the common questions, argues that this heuristic does not have a significant impact on the quality of the base ConVec model and it just improved the coverage (added more concepts to the list of concept vectors). The most important message of the third column of Table 5.2 is that even a very small dataset (Wikipedia 2.1 B tokens) can produce good vectors embedding in comparison with the Google freebase dataset (100B tokens) and consequently, the quality of the training corpus is more important than its size.

**Phrase Similarity Task:** The next experiment is evaluating vector quality in the Phrase similarity datasets (Check Table 5.3). In these datasets, each row consists of two words with their relatedness assigned by a human. The Spearman’s correlation is used for comparing the result of different approaches with the human evaluated results. These datasets contain words and not the Wikipedia concepts. We replaced all the words in these datasets with their corresponding Wikipedia pages if their surface form and the Wikipedia concept match. We used the simple but effective most frequent sense disambiguation method to disambiguate words that may correspond to several Wikipedia concepts. This method of assigning words to concepts is not error prone but this error is considered for all approaches.

Wikipedia Miner [71] is a well-known approach to find the similarity between two Wikipedia pages based on their input and output links. Results show that our approach for learning concepts embedding can embed the Wikipedia link structure properly since its results are similar to the structural based similarity approach of Wikipedia Miner (See Table 5.3). The average correlation for the heuristic based approach is less than the other approaches, but the average of not-found entries in this approach is much less than in the others. It shows that using the heuristic can increase the coverage of the Wikipedia concepts.

To have a fair comparison between different approaches, we extracted all common entries of all datasets and then re-calculated the correlation (Table 5.4). We also compared the results with another structural based similarity approach called

Table 5.4: Comparing the results in the Phrase Similarity datasets for the common entries between all approaches. Rho is Spearmans’s correlation.

Datasets			Wikipedia Miner	HitSim	ConVec	ConVec (Heuristic)	ConVec (Only Anchors)
#	Dataset Name	#Pairs	Rho	Rho	Rho	Rho	Rho
1	WS-REL	130	0.6662	0.5330	0.6022	0.6193	0.6515
2	SIMLEX	406	0.2405	0.3221	0.3011	0.3087	0.2503
3	WS-MAN [36]	224	0.6762	0.6854	0.6331	0.6371	0.6554
4	WS-411 [36]	314	0.7311	0.7131	0.7126	0.7136	0.7308
5	WS-SIM	108	0.7538	0.6968	0.7492	0.7527	0.7596
6	RWD	268	0.3072	0.2906	0.1989	0.1864	0.1443
7	WS-ALL	192	0.6656	0.6290	0.6372	0.6482	0.6733
8	RG	20	0.7654	0.7805	0.6647	0.7338	0.6301
9	MC	9	0.3667	0.5667	0.2667	0.2167	0.2833
10	MTurk	122	0.6627	0.5175	0.6438	0.6453	0.6432
-	Average	179	0.5333	0.5216	0.5114	0.5152	0.5054

HitSim [89]. The comparable result of our approach to structural based methods is another proof that we could embed the Wikipedia link structure properly. The result of the heuristic based approach is slightly better than our base model. This shows that without sacrificing the accuracy, we could increase the coverage. This means that with the proposed heuristic, we have a vector representation of more Wikipedia pages.

Results for the only anchors version of ConVec (see the last column of Table 5.4) show that in some datasets this approach is better than other approaches, but the average result is less than the other approaches. This shows it is better to learn Wikipedia’s concepts vector in the context of other words (words that are not anchored) and as a result to have the same vector space for both concepts and words.

## 5.2 Conclusion

In this chapter, several approaches for embedding Wikipedia concepts are introduced. The higher importance of the quality of the corpus than its quantity (size) is demonstrated and the idea of the larger corpus will not always lead to better word embedding is argued. Although the proposed approaches only use inter Wikipedia links (anchors), they have a performance as good as or even higher than the state of the

arts approaches for Concept Analogy and Concept Similarity tasks. Contrary to word embedding, Wikipedia concepts embedding is not ambiguous, so there is a different vector for concepts with similar surface forms but different mentions. This feature is important for many NLP tasks such as Named Entity Recognition, Text Similarity, and Document Clustering or Classification.

## Chapter 6

### Conclusion and Future Research

#### 6.1 Conclusion

The evaluation of the interactive document clustering system indicates that users want two capabilities from an interactive document clustering system. First, the ability to effectively interact with the clustering algorithm. Second, a visualization that enables them to explore the document collection and the result of clustering efficiently.

To address the first need, we need to replace the conventional non-interactive clustering algorithms by interactive ones. We selected key-terms as an interaction between the user and the clustering algorithm. In key-term interaction, the user assigns a set of key-terms to each cluster to guide the clustering algorithm. While key-term interaction is more intuitive and convenient for the users, some user still wants to have document interaction beside it. In document interaction, the user assigns a set of documents to each cluster. Sometimes the user wants to have a more sophisticated interaction. For example, asking the clustering algorithm to divide a cluster based on the sentiment of documents. Ability to support more diverse and complicated interactions may help the user better interact with the clustering algorithm. However not every types of user feedback may lead to a better clustering result. There are some cases that the user provides key-terms that may not be coherent to each other. To prevent this problem, first we guide the user with different visualization modules to better select the key-terms and second, in the clustering algorithm, key-terms that not have a high *tf-idf* value will have less impact on the clustering result. We still need to work on this feature to prevent negative impact of low-quality user feedback on clustering quality. The rollback feature of the proposed system can be helpful to revert back every unwanted changes.

For the second need, we designed several visualization modules to give the user a comprehensive view of the document collection. While most of the users find most of

these modules helpful, some of them complained about the difficulty of understanding so many features in a single screen. This problem can be handled by visualizing only the important features of the document cluster and show the other features on demand.

The main goal of supporting two different clustering algorithms of (LDC and iKMeans) was to indicate that the visual interface is designed independent of the clustering algorithm and any key-term based clustering algorithm can be combined with it. During the user study, we briefly explained the differences between two clustering algorithms and made the iKMeans the default algorithm. Only one of the users switched between the clustering algorithms and the rest decided to stay with the default one. Because of the lack of evidence, we cannot clearly decide whether if users effectively switched between clustering algorithms. This observation indicates that the switch between algorithms needs expert knowledge and users do not feel confident to be involved in this procedure.

The evaluation results on general and domain-specific datasets indicate that it is possible to use the proposed system in diverse domains. In some cases, it is necessary to change some part of the system to better adapt to a domain. For example, we changed the preprocessing of the system to better extract the email conversations in the Respiratory Therapy case study.

During the user study we have notified that user needs deterministic results. If the initial result of the clustering is non-deterministic, the user may get confused by the inconsistent clustering. It is possible to store the initial data points to make the result of a non-deterministic method repeatable, but it may lead to a bad quality solution unless one initializes the clustering algorithm several times and then considers the one which has optimized the objective function the most, which is a very time-consuming process. To tackle this problem, we introduced the DSKM algorithm that not only deterministically initializes the KMeans but it also improves the clustering time and performance. The proposed method is optimized for the textual datasets; expanding the proposed idea to non-textual datasets could be beneficial.

The importance of considering the temporal aspect of document clustering is explained in Chapter 4. In our experiments, we assumed that each document has a creation time and we designed a similarity measure that considers both temporal

and content similarity between documents. Evaluation indicates that in most of the datasets using this similarity measure leads to the better clustering result. We combined this similarity measure with an interactive document clustering algorithm and we thus introduced a novel interactive temporal document clustering algorithm.

The overall goal of chapter 5 was about using Wikipedia knowledge base for improving the quality of different text mining tasks such as document clustering. We demonstrated the higher importance of the quality of the corpus than its quantity (size) and argued that the larger corpus will not always lead to better word embedding. Although the proposed approaches only use inter Wikipedia links (anchors), they have a performance as good as or even higher than the state of the art approaches for Concept Analogy and Concept Similarity tasks. Contrary to word embeddings, Wikipedia concept embeddings are not ambiguous, so there is a different vector for concepts with similar surface forms but different meaning.

## 6.2 Future Research

**Clusters change follow up** It is important to effectively visualize how the clusters change over each interaction of the user with the clustering algorithm. This will help the user to observe her/his changes on the clustering and at the same time be able to roll back her/his changes partially. In the current system, we will save result of the clustering before each interaction of the user with the system as a session. It is possible to use these session to visualize the cluster changes as a Sankey diagram [82].

**Information Retrieval** In the era of search engines, users like to find their questions by searching them on search engines. In the proposed visual interface there is a search bar for users to start searching their interested keywords inside the collection. Currently, a basic string matching method is used for finding relevant documents for the user. It would be beneficial if we use more advanced information retrieval methods such as OKapi BM25 method [84] and elaborate ideas of total recall problem [85].

**Scalability** the size of the datasets increases, it will be more challenging to effectively visualize the clustering result. One of the difficulties is depicting all documents

in the Graph View (see Section 2.3.4). A possible solution is grouping similar documents as a single node and whenever the user clicks on that node, it expands and show the documents. The other future direction is clustering a huge datasets interactively with multiple users. In this case, users collaborate (remotely) with each other to cluster the same dataset. This idea needs methods for resolving conflicts and preventing any possible deadlocks as a result of multi-user interaction.

**Temporal Clustering** Currently, we consider the temporal feature of documents in the similarity measure. It is possible to incorporate the temporal aspect in other cases as well. For example, separately cluster documents in different time intervals and then try to link clusters in each time interval. This may leads to help the user to extract the evolution of clusters over each time stamp.

**Wikification** Combining Wikipedia concepts with bag of word model can improve the result of document clustering [48]. In order to use Wikipedia for getting more semantic from the text, the Wikification of text is needed. One of the important steps of Wikification is disambiguation of Wikipedia concepts. The Wikipedia concepts embeddings as a result of our proposed method in chapter 5, can be used for disambiguating Wikipedia concepts. In this approach, the mention of a concept surface form is the one which has the closer vector to the context of the document.

In order to improve the quality of Wikipedia concept vector, we plan to use multiple resources such as Infoboxes, multilingual version of a Wikipedia page, Categories and syntactical features of a page to improve the quality of Wikipedia concepts embedding.



## Bibliography

- [1] Herv Abdi and Lynne J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4):433–459, 2010.
- [2] Charu C. Aggarwal and Philip S. Yu. Outlier Detection for High Dimensional Data. In *Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data*, SIGMOD '01, pages 37–46, New York, NY, USA, 2001. ACM.
- [3] E. Alexander, J. Kohlmann, R. Valenza, M. Witmore, and M. Gleicher. Serendip: Topic model-driven visual exploration of text corpora. In *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 173–182, Oct 2014.
- [4] David Andrzejewski, Xiaojin Zhu, and Mark Craven. Incorporating Domain Knowledge into Topic Modeling via Dirichlet Forest Priors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 25–32, New York, NY, USA, 2009. ACM.
- [5] David Arthur and Sergei Vassilvitskii. K-means++: The Advantages of Careful Seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.
- [6] Pranjali Awasthi, Maria-Florina Balcan, and Konstantin Voevodski. Local Algorithms for Interactive Clustering. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages II–550–II–558. JMLR.org, 2014.
- [7] Olivier Bachem, Mario Lucic, Hamed Hassani, and Andreas Krause. Fast and Provably Good Seedings for k-Means. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 55–63. Curran Associates, Inc., 2016.
- [8] Maria-Florina Balcan and Avrim Blum. Clustering with Interactive Feedback. In *Proceedings of the 19th International Conference on Algorithmic Learning Theory*, ALT '08, pages 316–328, Berlin, Heidelberg, 2008. Springer-Verlag.
- [9] Arindam Banerjee, Inderjit S. Dhillon, Joydeep Ghosh, and Suvrit Sra. Clustering on the Unit Hypersphere Using Von Mises-Fisher Distributions. *J. Mach. Learn. Res.*, 6:1345–1382, December 2005.
- [10] Josh Barnes and Piet Hut. A hierarchical  $O(N \log N)$  force-calculation algorithm. *nature*, 324(6096):446–449, 1986.

- [11] Sugato Basu, Arindam Banerjee, and Raymond J. Mooney. Semi-supervised Clustering by Seeding. In *Proceedings of the Nineteenth International Conference on Machine Learning, ICML '02*, pages 27–34, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.
- [12] Sugato Basu, Ian Davidson, and Kiri Wagstaff. *Constrained Clustering: Advances in Algorithms, Theory, and Applications*. Chapman & Hall/CRC, 1 edition, 2008.
- [13] Ron Bekkerman, Hema Raghavan, James Allan, and Koji Eguchi. Interactive Clustering of Text Collections According to a User-specified Criterion. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI'07*, pages 684–689, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.
- [14] Matthew Berger, Katherine McDonough, and Lee M. Seversky. Cite2Vec: Citation-Driven Document Exploration via Word Embeddings. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):691–700, January 2017.
- [15] James C. Bezdek. A Convergence Theorem for the Fuzzy ISODATA Clustering Algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2(1):1–8, January 1980.
- [16] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1981.
- [17] Mikhail Bilenko, Sugato Basu, and Raymond J. Mooney. Integrating Constraints and Metric Learning in Semi-supervised Clustering. In *Proceedings of the Twenty-first International Conference on Machine Learning, ICML '04*, pages 11–, New York, NY, USA, 2004. ACM.
- [18] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.
- [19] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D3 Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, December 2011.
- [20] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. LOF: Identifying Density-based Local Outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, SIGMOD '00*, pages 93–104, New York, NY, USA, 2000. ACM.
- [21] John Brooke et al. SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.
- [22] D. Cai, X. He, and J. Han. Document clustering using locality preserving indexing. *IEEE Transactions on Knowledge and Data Engineering*, 17(12):1624–1637, Dec 2005.

- [23] José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. Nasari: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. *Artificial Intelligence*, 240:36–64, 2016.
- [24] Shaosheng Cao, Wei Lu, and Qiongkai Xu. Deep neural networks for learning graph representations. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 1145–1152. AAAI Press, 2016.
- [25] Ana Cardoso-Cachopo. Improving Methods for Single-label Text Categorization. PdD Thesis, Instituto Superior Tecnico, Universidade Tecnica de Lisboa, 2007.
- [26] M. Emre Celebi and Hassan A. Kingravi. *Partitional Clustering Algorithms*, chapter Linear, Deterministic, and Order-Invariant Initialization Methods for the K-Means Clustering Algorithm, pages 79–98. Springer International Publishing, Cham, 2015.
- [27] M. Emre Celebi, Hassan A. Kingravi, and Patricio A. Vela. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Systems with Applications*, 40(1):200 – 210, 2013.
- [28] M. Chang, L. Ratinov, D. Roth, and V. Srikumar. Importance of Semantic Representation: Dataless Classification. In *AAAI*, 7 2008.
- [29] Jaegul Choo, Changhyun Lee, Chandan K. Reddy, and Haesun Park. UTOPIAN: User-Driven Topic Modeling Based on Interactive Nonnegative Matrix Factorization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1992–2001, December 2013.
- [30] Jason Chuang, Christopher D. Manning, and Jeffrey Heer. Termite: Visualization Techniques for Assessing Textual Topic Models. In *Proceedings of the International Working Conference on Advanced Visual Interfaces, AVI '12*, pages 74–77, New York, NY, USA, 2012. ACM.
- [31] Mark Craven, Dan DiPasquo, Dayne Freitag, Andrew McCallum, Tom Mitchell, Kamal Nigam, and Seán Slattery. Learning to Extract Symbolic Knowledge from the World Wide Web. In *Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence, AAAI '98/IAAI '98*, pages 509–516, Menlo Park, CA, USA, 1998. American Association for Artificial Intelligence.
- [32] Ian Davidson and S. S. Ravi. Clustering with Constraints: Feasibility Issues and the k-Means Algorithm. In *SDM*, 2005.
- [33] Inderjit S. Dhillon and Dharmendra S. Modha. Concept Decompositions for Large Sparse Text Data Using Clustering. *Mach. Learn.*, 42(1-2):143–175, January 2001.

- [34] Wenwen Dou, Li Yu, Xiaoyu Wang, Zhiqiang Ma, and William Ribarsky. HierarchicalTopics: Visually Exploring Large Text Collections Using Topic Hierarchies. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2002–2011, December 2013.
- [35] Manaal Faruqui and Chris Dyer. Improving vector space word representations using multilingual correlation. Association for Computational Linguistics, 2014.
- [36] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppín. Placing Search in Context: The Concept Revisited. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pages 406–414, New York, NY, USA, 2001. ACM.
- [37] Thomas M. J. Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164, 1991.
- [38] D. Greene, D. O’Callaghan, and P. Cunningham. How Many Topics? Stability Analysis for Topic Models. In *Proc. European Conference on Machine Learning (ECML’14)*, 2014.
- [39] Derek Greene and Pádraig Cunningham. Producing Accurate Interpretable Clusters from High-dimensional Data. In *Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, PKDD’05, pages 486–494, Berlin, Heidelberg, 2005. Springer-Verlag.
- [40] Guy Halawi, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. Large-scale learning of word relatedness with constraints. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1406–1414. ACM, 2012.
- [41] Mohammad Al Hasan, Vineet Chaoji, Saeed Salem, and Mohammed J. Zaki. Robust Partitional Clustering by Outlier and Density Insensitive Seeding. *Pattern Recognition Letters*, 30(11):994 – 1002, 2009.
- [42] Susan Havre, Beth Hetzler, and Lucy Nowell. ThemeRiverTM: In search of trends, patterns, and relationships. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):9–20, 2002.
- [43] Felix Hill, Roi Reichart, and Anna Korhonen. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 2016.
- [44] Xiaohua Hu, Xiaodan Zhang, Caimei Lu, Eun K Park, and Xiaohua Zhou. Exploiting Wikipedia as external knowledge for document clustering. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 389–396. ACM, 2009.

- [45] Yeming Hu, Evangelos E. Milios, James Blustein, and Shali Liu. Personalized Document Clustering with Dual Supervision. In *Proceedings of the 2012 ACM Symposium on Document Engineering, DocEng '12*, pages 161–170, New York, NY, USA, 2012. ACM.
- [46] Yifan Hu. Efficient, high-quality force-directed graph drawing. *Mathematica Journal*, 10(1):37–71, 2005.
- [47] Yuening Hu, Jordan Boyd-Graber, and Brianna Satinoff. Interactive Topic Modeling. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, HLT '11*, pages 248–257, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.
- [48] Anna Huang, David Milne, Eibe Frank, and Ian H. Witten. Clustering Documents Using a Wikipedia-Based Concept Representation. In Thanaruk Theeramunkong, Boonserm Kijirikul, Nick Cercone, and Tu-Bao Ho, editors, *Advances in Knowledge Discovery and Data Mining*, pages 628–636, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.
- [49] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.
- [50] I. Katsavounidis, C. C. Jay Kuo, and Zhen Zhang. A new initialization technique for generalized Lloyd iteration. *IEEE Signal Processing Letters*, 1(10):144–146, Oct 1994.
- [51] M. Kim, K. Kang, D. Park, J. Choo, and N. Elmqvist. TopicLens: Efficient Multi-Level Visual Topic Exploration of Large-Scale Document Collections. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):151–160, Jan 2017.
- [52] Jon M Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.
- [53] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.
- [54] Chia-Tung Kuo, SS Ravi, Thi-Bich-Hanh Dao, Christel Vrain, and Ian Davidson. A Framework for Minimal Clustering Modification via Constraint Programming. In *AAAI*, pages 1389–1395, 2017.
- [55] Ken Lang. Newsweeder: Learning to filter netnews.
- [56] Quoc V Le and Tomas Mikolov. Distributed Representations of Sentences and Documents. In *ICML*, volume 14, pages 1188–1196, 2014.

- [57] Hanseung Lee, Jaeyeon Kihm, Jaegul Choo, John Stasko, and Haesun Park. iVisClustering: An Interactive Visual Document Clustering via Topic Modeling. *Computer Graphics Forum*, 31(3pt3):1155–1164, 2012.
- [58] Kevin J Lee, Frank F Tu, Huong G Nghiem, and Andrew I Sokol. Promises and pitfalls of the AAGL LISTSERV: a descriptive analysis. *Journal of minimally invasive gynecology*, 17(4):407–410, 2010.
- [59] Y. Li, C. Luo, and S. M. Chung. Text Clustering with Feature Selection by Using Statistical Data. *IEEE Transactions on Knowledge and Data Engineering*, 20(5):641–652, May 2008.
- [60] Bing Liu, Xiaoli Li, Wee Sun Lee, and Philip S. Yu. Text Classification by Labeling Words. In *Proceedings of the 19th National Conference on Artificial Intelligence*, AAAI’04, pages 425–430. AAAI Press, 2004.
- [61] S. Liu, X. Wang, J. Chen, J. Zhu, and B. Guo. TopicPanorama: A full picture of relevant topics. In *2014 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 183–192, Oct 2014.
- [62] Tao Liu, Shengping Liu, Zheng Chen, and Wei-Ying Ma. An Evaluation on Feature Selection for Text Clustering. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML’03, pages 488–495. AAAI Press, 2003.
- [63] Samantha Long, Desleigh De Jonge, Jenny Ziviani, and Alison Jones. Paediatricots: utilisation of an Australian list serve to support occupational therapists working with children. *Australian occupational therapy journal*, 56(1):63–71, 2009.
- [64] Thang Luong, Richard Socher, and Christopher D Manning. Better word representations with recursive neural networks for morphology. In *CoNLL*, pages 104–113, 2013.
- [65] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.
- [66] Juha Makkonen, Helena Ahonen-Myka, and Marko Salmenkivi. Simple Semantics in Topic Detection and Tracking. *Inf. Retr.*, 7(3-4):347–368, September 2004.
- [67] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.
- [68] Rafael Messias Martins, Danilo Barbosa Coimbra, Rosane Minghim, and A.C. Telea. Visual analysis of dimensionality reduction quality for parameterized projections. *Computers and Graphics*, 41:26 – 42, 2014.

- [69] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.
- [70] George A Miller and Walter G Charles. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28, 1991.
- [71] David Milne and Ian H. Witten. Learning to Link with Wikipedia. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM '08*, pages 509–518, New York, NY, USA, 2008. ACM.
- [72] N. Nidheesh, K.A. Abdul Nazeer, and P.M. Ameer. An enhanced deterministic K-Means clustering algorithm for cancer subtype prediction from gene expression data. *Computers in Biology and Medicine*, 91:213 – 221, 2017.
- [73] Seyednaser Nourashrafeddin, Evangelos Milios, and Dirk Arnold. Interactive Text Document Clustering Using Feature Labeling. In *Proceedings of the 2013 ACM Symposium on Document Engineering, DocEng '13*, pages 61–70, New York, NY, USA, 2013. ACM.
- [74] Seyednaser Nourashrafeddin, Evangelos Milios, and Drik V. Arnold. An Ensemble Approach for Text Document Clustering Using Wikipedia Concepts. In *Proceedings of the 2014 ACM Symposium on Document Engineering, DocEng '14*, pages 107–116, New York, NY, USA, 2014. ACM.
- [75] Seyednaser Nourashrafeddin, Ehsan Sherkat, Rosane Minghim, and Evangelos E. Milios. A Visual Approach for Interactive Keyterm-Based Clustering. *ACM Trans. Interact. Intell. Syst.*, 8(1):6:1–6:35, February 2018.
- [76] Pentti Paatero and Unto Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.
- [77] F. V. Paulovich, L. G. Nonato, R. Minghim, and H. Levkowitz. Least Square Projection: A Fast High-Precision Multidimensional Projection Technique and Its Application to Document Mapping. *IEEE Transactions on Visualization and Computer Graphics*, 14(3):564–575, May 2008.
- [78] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global Vectors for Word Representation. In *EMNLP*, volume 14, pages 1532–43, 2014.
- [79] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710. ACM, 2014.

- [80] Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research*, 11(Sep):2487–2531, 2010.
- [81] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. Labeled LDA: A Supervised Topic Model for Credit Attribution in Multi-labeled Corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 248–256, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.
- [82] P. Riehmann, M. Hanfler, and B. Froehlich. Interactive Sankey diagrams. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pages 233–240, Oct 2005.
- [83] Marian-Andrei Rizoiu, Julien VELCIN, Stéphane Bonnevey, and Stéphane Lallich. ClusPath: a temporal-driven clustering to infer typical evolution paths.
- [84] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. Okapi at TREC-3. *Nist Special Publication Sp*, 109:109, 1995.
- [85] Adam Roegiest, Gordon V Cormack, Maura R Grossman, and Charles Clarke. TREC 2015 total recall track overview. *Proc. TREC-2015*, 2015.
- [86] Tony Rose, Mark Stevenson, and Miles Whitehead. The Reuters Corpus Volume 1-from Yesterday's News to Tomorrow's Language Resources. In *LREC*, volume 2, pages 827–832, 2002.
- [87] Andrew Rosenberg and Julia Hirschberg. V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure. In *EMNLP-CoNLL*, volume 7, pages 410–420, 2007.
- [88] Peter Rousseeuw. Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *J. Comput. Appl. Math.*, 20(1):53–65, November 1987.
- [89] Armin Sajadi, Evangelos E Milios, Vlado Kešelj, and Jeannette CM Janssen. Domain-Specific Semantic Relatedness from Wikipedia Structure: A Case Study in Biomedical Text. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 347–360. Springer, 2015.
- [90] Ehsan Sherkat, Evangelos Milios, and Rosane Minghim. A Visual Analytics Approach for Interactive Document Clustering. In *Journal of ACM Transactions on Interactive Intelligent Systems (TIIS 2019) - (in press)*.
- [91] Ehsan Sherkat and Evangelos E. Milios. Vector Embedding of Wikipedia Concepts and Entities. In Flavius Frasincar, Ashwin Ittoo, Le Minh Nguyen, and Elisabeth Métais, editors, *Natural Language Processing and Information Systems*, pages 418–428, Cham, 2017. Springer International Publishing.



- [92] Ehsan Sherkat, Seyednaser Nourashrafeddin, Evangelos E. Milios, and Rosane Minghim. Interactive Document Clustering Revisited: A Visual Analytics Approach. In *23rd International Conference on Intelligent User Interfaces, IUI '18*, pages 281–292, New York, NY, USA, 2018. ACM.
- [93] Ehsan Sherkat, Seyednaser Nourashrafeddin, Rosane Minghim, and Evangelos Milios. A Visual Approach for Interactive Expertise Finding and Exploration. In *CIKM 2016 Workshop on Data-Driven Talent Acquisition*, 2016.
- [94] Ehsan Sherkat, Julien Velcin, and Evangelos E. Milios. Fast and Simple Deterministic Seeding of KMeans for Text Document Clustering. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 76–88, Cham, 2018. Springer International Publishing.
- [95] Kathy Spurr, Gail Dechman, Kelly Lackie, and Robert Gilbert. Creation of a Tool for Assessing Knowledge in Evidence-Based Decision-Making in Practicing Health Care Providers. *Journal of Continuing Education in the Health Professions*, 36(3):164–170, 2016.
- [96] Ting Su and Jennifer G. Dy. In Search of Deterministic Methods for Initializing K-means and Gaussian Mixture Clustering. *Intell. Data Anal.*, 11(4):319–338, December 2007.
- [97] Teresa L Cervantez Thompson and Barbara Penprase. RehabNurse-L: An Analysis of the Rehabilitation Nursing LISTSERV Experience. *Rehabilitation Nursing*, 29(2):56–61, 2004.
- [98] Chen-Tse Tsai and Dan Roth. Cross-lingual Wikification Using Multilingual Embeddings. In *HLT-NAACL*, pages 589–598, 2016.
- [99] Laurens Van Der Maaten. Accelerating t-SNE Using Tree-based Algorithms. *J. Mach. Learn. Res.*, 15(1):3221–3245, January 2014.
- [100] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. *J. Mach. Learn. Res.*, 11:2837–2854, December 2010.
- [101] Kiri Wagstaff, Claire Cardie, Seth Rogers, and Stefan Schrödl. Constrained K-means Clustering with Background Knowledge. In *Proceedings of the Eighteenth International Conference on Machine Learning, ICML '01*, pages 577–584, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.
- [102] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.
- [103] Stephen J. Wright. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, Jun 2015.

- [104] Yi Yang, Shimei Pan, Yangqiu Song, Jie Lu, and Mercan Topkara. User-directed Non-Disruptive Topic Model Update for Effective Exploration of Dynamic Content. In *Proceedings of the 20th International Conference on Intelligent User Interfaces, IUI '15*, pages 158–168, New York, NY, USA, 2015. ACM.
- [105] Yiming Yang and Jan O. Pedersen. A Comparative Study on Feature Selection in Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning, ICML '97*, pages 412–420, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.

## Appendix A

### Evaluation Metrics

- **Adjusted Rand Index/Score (ARI/ARS) [49]:**

Assume  $C$  is a ground truth class assignment and  $K$  the clustering result. The Rand index is given by:

$$RI = \frac{a + b}{C_2^{n_{samples}}}$$

$n$  is the total number of samples.  $a$  is the number of pairs of elements that are in the same set in  $C$  and  $K$ .  $b$  is the number of pairs of elements that are in different set in  $C$  and  $K$ . Let  $E[RI]$  be the expected value of  $RI$ , the ARI is defined as:

$$ARI = \frac{RI - E[RI]}{\max(RI) - E[RI]}$$

- **Normalized and Adjusted Mutual Information (NMI and AMI) [100]:**

Let  $U$  and  $V$  be two sets of label assignment of  $n$  data points. The entropy of these sets is defined by:

$$En(U) = - \sum_{i=1}^{|U|} P(i) \log(P(i))$$

$$En(V) = - \sum_{j=1}^{|V|} P'(j) \log(P'(j))$$

Where  $P(i) = |U_i|/N$  is the probability randomly selected data points from  $U$  falls into class  $U_i$ . . The Mutual Information (MI) between  $U$  and  $V$  is given by:

$$\text{MI}(U, V) = \sum_{i=1}^{|U|} \sum_{j=1}^{|V|} P(i, j) \log \left( \frac{P(i, j)}{P(i)P'(j)} \right)$$

The Normalized Mutual Information now can be defined as:

$$\text{NMI}(U, V) = \frac{\text{MI}(U, V)}{\text{mean}(En(U), En(V))}$$

Let  $E[\text{MI}]$  be the expected value of MI, then the AMI is given by:

$$\text{AMI} = \frac{\text{MI} - E[\text{MI}]}{\text{mean}(En(U), En(V)) - E[\text{MI}]}$$

- **Homogeneity [87]:**

$$\text{Homogeneity} = 1 - \frac{En(C|K)}{En(C)}$$

$En(C|K)$  is the entropy of the classes given cluster assignments and is defined as:

$$En(C|K) = - \sum_{c=1}^{|C|} \sum_{k=1}^{|K|} \frac{n_{c,k}}{n} \cdot \log \left( \frac{n_{c,k}}{n_k} \right)$$

$En(C)$  is the entropy of the classes and is formulated as:

$$En(C) = - \sum_{c=1}^{|C|} \frac{n_c}{n} \cdot \log \left( \frac{n_c}{n} \right)$$

$n$  is the total number of samples,  $n_c$  is the number of samples in class  $c$ ,  $n_k$  is the number of samples belonging to cluster  $k$ , and  $n_{c,k}$  is the number of samples from class  $c$  assigned to cluster  $k$ .

- **Silhouette [88]:**

Let  $a$  be the average distance between a data point and all other points in the same class and  $b$  be the average distance between a data point and all other points in the next nearest cluster. The Silhouette is defined as:

$$\text{Silhouette} = \frac{b - a}{\max(a, b)}$$

- **Accuracy:**

Let  $Y$  be the true labels and  $Y'$  be the predicted labels. The accuracy is defined as:

$$\text{Accuracy}(Y, Y') = \frac{1}{n} \sum_{i=0}^{n-1} \mathbb{1}(Y'_i = Y_i)$$

$n$  is the total number of samples.  $\mathbb{1}$  is the indicator function.

## Appendix B

### User Study Design

#### B.1 Screening Questionnaire

Identification number: .....

1. At what level do you think your understanding of written English is?

- Excellent
- Very good
- Good
- Acceptable
- Bad
- Very bad
- None

2. What is the highest level of education you have completed?

- Little or no formal education
- High school or equivalent
- College or university
- Master
- Doctoral
- Post-Doctoral

3. How often do you read newspapers or magazines?

- Every day
- Once two days

- Once four days
- Once a week
- Once a month
- Once a year
- Never

4. How often do you read academic papers?

- Every day
- Once two days
- Once four days
- Once a week
- Once a month
- Once a year
- Never

5. What is your primary area of study?

- Computer Science
- Information technology
- Internetworking
- Other.....

6. How much are you familiar with the document clustering?

- Excellent
- Very good
- Good
- Acceptable
- Bad
- Very bad
- None

## B.2 Demographic Questionnaire

Identification number: \_\_\_\_\_

Gender: \_\_\_\_Male \_\_\_\_Female \_\_\_\_Other

1. On the average, how much time do you spend per week on a computer?
  - Less than one hour
  - One to less than 4 hours
  - 4 to less than 10 hours
  - 10 to less than 20 hours
  - 20 to less than 40 hours
  - Over 40 hours
  
2. How often do you use interactive user interfaces such as dragging objects from one place to another?
  - Extremely often
  - Very often
  - Often
  - Not often
  - Seldom
  - Never
  
3. How comfortable are you at using interactive user interface?
  - Extremely comfortable
  - Very comfortable
  - Comfortable
  - Uncomfortable
  - Very uncomfortable
  - Extremely uncomfortable



### B.3 Interface Features Questionnaire

1	It is useful to highlight the documents containing certain term in the Graph View.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
2	It is useful to see the corresponding node of selected document in the Graph View.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
3	It is NOT useful to see/load the document content from the Graph View.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
4	It is useful to see top 5 terms of each document in the Graph View.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
5	The keep functionality plus selecting neighbors documents in the Graph View is useful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
6	The ability to create a Term Cloud from selected documents in the Graph View is useful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
7	The ability to create a Term Cloud of the selected document is NOT useful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
8	It is useful to see the Term Cloud of each cluster.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
9	It is useless to interact with Graph View by changing the Cosine Distance slider.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
10	It is useful to have two different projections (t-SNE vs Force layout) in the Graph View.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
11	Changing Force layout parameters is helpful	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
12	It is NOT useful to filter the Graph View based on the name of documents.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
13	It is useful to change the name of the clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
14	It is useful to change the color of clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
15	The searching and adding terms from search bar is necessary.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
16	It is helpful to add terms from Term Cloud View to the clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
17	It is NOT helpful to add terms from Cluster key-terms View to the clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
18	It is helpful to add terms from Document View to the clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
19	It is helpful to add term from one cluster to another cluster in Cluster View.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
20	Cluster Key terms view is useful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
21	The bar charts in Cluster Key terms is NOT useful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree

		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
22	Parallel coordinate diagram of terms relatedness in Term-Cluster view is useful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
23	It is useful to compare two or more terms relatedness in Term-Cluster view.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
24	Parallel coordinate diagram of document relatedness in Document-Cluster view is useful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
25	It is helpful to have an access to the list of documents of each cluster in order of their relatedness in Document view.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
26	It is easy to identify the topic of a document cluster from the key-terms in the corresponding list of clusters in Cluster View.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
27	It is easy to add a new cluster.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
28	The Silhouette score of Clustering, t-SNE and Force layout was helpful to find the quality of clustering after each interaction.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
29	It is useful to be able to save/reload (save and load sessions) the clustering results.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
30	It is useful to write note about each session of clustering.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
31	It is useful to have all the clusters in a single view beside each other (Cluster View).	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
32	It is necessary to have an access to the text of documents in Document View.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
33	It is NOT useful to have the cluster hierarchy in Cluster Tree View	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
34	It is useful to save the result of clustering as ZIP file	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
36	It is useful to save the result of clustering in MindMap format	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
37	It is helpful to suggest the user the initial number of clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
38	It is essential to suggest the user the initial number of clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
39	It is more meaningful to use phrases instead of single words to determining the clusters topics.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
40	The upload section was helpful for processing user dataset.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
41	It is necessary to follow the user determined cluster name after each re-clustering.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
42	It is NOT useful for the user to follow clusters name after each re-clustering.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
43	It is essential for the user to follow clusters name after each re-clustering.	1	2	3	4	5

		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
44	The hierarchical clustering feature is useful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
45	The Silhouette Chart View was helpful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
46	The user confidence level slider was helpful.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
47	Which of the following views do you think is more useful?					
		Document view + document cluster view	Cluster key-terms view plus term-clusterview	Cluster view	Term cloud View	Graph View

## B.4 Interface Rating Questionnaire

1	The automatically generated clusters are near to what you desired?	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
2	It is necessary to be able to change the topics of document clusters generated automatically.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
3	It is necessary to be able to change the number of clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
4	Term based visualization is a useful way in exploring the topics of a collection.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
5	Term based visualization and term labeling is a useful way in generating desired cluster topics.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
6	Term based visualization is a useful way to find a desired number of clusters.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
7	The user interface is a useful tool for document clustering in general.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
8	Was the documentation and training sufficient for you to learn how to use the interface?	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree
9	I would like to use the interface in the future.	1	2	3	4	5
		Strongly Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Strongly Agree

## B.5 Dalhousie Ethic Board's Letter of Approval

### REB # 2016-4012 Letter of Approval

1 message

angela.hersey@dal.ca <angela.hersey@dal.ca>  
 To: "Ehsan Sherkat (Principal Investigator)" <ehsansherkat@dal.ca>  
 Cc: "Evangelos Milios (Supervisor)" <EMILIOS@dal.ca>, angela.hersey@dal.ca

Fri, Dec 23, 2016 at 9:31AM



#### Social Sciences & Humanities Research Ethics Board Letter of Approval

December 23, 2016

Ehsan Sherkat  
 Computer Science\Computer Science

Dear Ehsan,

**REB #:** 2016-4012  
**Project Title:** User evaluation of the interactive document clustering system  
**Effective Date:** December 23, 2016  
**Expiry Date:** December 23, 2017

The Social Sciences & Humanities Research Ethics Board has reviewed your application for research involving humans and found the proposed research to be in accordance with the Tri-Council Policy Statement on *Ethical Conduct for Research Involving Humans*. This approval will be in effect for 12 months as indicated above. This approval is subject to the conditions listed below which constitute your on-going responsibilities with respect to the ethical conduct of this research.

Sincerely,



Dr. Karen Beazley, Chair

---

#### Post REB Approval: On-going Responsibilities of Researchers

After receiving ethical approval for the conduct of research involving humans, there are several ongoing responsibilities that researchers must meet to remain in compliance with University and Tri-Council policies.

##### 1. Additional Research Ethics approval

Prior to conducting any research, researchers must ensure that all required research ethics approvals are secured (in addition to this one). This includes, but is not limited to, securing appropriate research ethics approvals from: other institutions with whom the PI is affiliated; the research institutions of research team members; the institution at which participants may be recruited or from which data may be collected; organizations or groups (e.g. school boards, Aboriginal communities, correctional services, long-term care facilities, service agencies and community groups) and from any other responsible review body or bodies at the research site

## 2. Reporting adverse events

Any significant adverse events experienced by research participants must be reported **in writing** to Research Ethics **within 24 hours** of their occurrence. Examples of what might be considered “significant” include: an emotional breakdown of a participant during an interview, a negative physical reaction by a participant (e.g. fainting, nausea, unexpected pain, allergic reaction), report by a participant of some sort of negative repercussion from their participation (e.g. reaction of spouse or employer) or complaint by a participant with respect to their participation. The above list is indicative but not all-inclusive. The written report must include details of the adverse event and actions taken by the researcher in response to the incident.

## 3. Seeking approval for protocol / consent form changes

Prior to implementing any changes to your research plan, whether to the protocol or consent form, researchers must submit a description of the proposed changes to the Research Ethics Board for review and approval. This is done by completing an Amendment Request (available on the website). Please note that no reviews are conducted in August.

## 4. Submitting annual reports

Ethics approvals are valid for up to 12 months. Prior to the end of the project's approval deadline, the researcher must complete an Annual Report (available on the website) and return it to Research Ethics for review and approval before the approval end date in order to prevent a lapse of ethics approval for the research. Researchers should note that no research involving humans may be conducted in the absence of a valid ethical approval and that allowing REB approval to lapse is a violation of University policy, inconsistent with the TCPS (article 6.14) and may result in suspension of research and research funding, as required by the funding agency.

## 5. Submitting final reports

When the researcher is confident that no further data collection or participant contact will be required, a Final Report (available on the website) must be submitted to Research Ethics. After review and approval of the Final Report, the Research Ethics file will be closed.

## 6. Retaining records in a secure manner

Researchers must ensure that both during and after the research project, data is securely retained and/or disposed of in such a manner as to comply with confidentiality provisions specified in the protocol and consent forms. This may involve destruction of the data, or continued arrangements for secure storage. Casual storage of old data is not acceptable.

It is the Principal Investigator's responsibility to keep a copy of the REB approval letters. This can be important to demonstrate that research was undertaken with Board approval, which can be a requirement to publish (and is required by the Faculty of Graduate Studies if you are using this research for your thesis).

Please note that the University will securely store your REB project file for 5 years after the study closure date at which point the file records may be permanently destroyed.

## 7. Current contact information and university affiliation

The Principal Investigator must inform the Research Ethics office of any changes to contact information for the PI (and supervisor, if appropriate), especially the electronic mail address, for the duration of the REB approval. The PI must inform Research Ethics if there is a termination or interruption of his or her affiliation with Dalhousie University.

## 8. Legal Counsel

The Principal Investigator agrees to comply with all legislative and regulatory requirements that apply to the project. The Principal Investigator agrees to notify the University Legal Counsel office in the event that he or she receives a notice of non-compliance, complaint or other proceeding relating to such requirements.

## 9. Supervision of students

Faculty must ensure that students conducting research under their supervision are aware of their responsibilities as described above, and have adequate support to conduct their research in a safe and ethical manner.

**REB # 2016-4012 Annual Renewal - Approval**

1 message

**do-not-reply-DAL@researchservicesoffice.com** <do-not-reply-DAL@researchservicesoffice.com>

Wed, Dec 6, 2017 at  
10:50 AM

To: "Ehsan Sherkat (Principal Investigator)" <eh379022@dal.ca>

Cc: "Evangelos Miliios (Supervisor)" <EMILIOS@dal.ca>, do-not-reply-DAL@researchservicesoffice.com

*\*\*\*This was sent from a no-reply address. To respond to this message, please reply directly to Keerthi Luthra at [Keerthi.Luthra@dal.ca](mailto:Keerthi.Luthra@dal.ca).*



**Social Sciences & Humanities Research Ethics Board  
Annual Renewal - Letter of Approval**

December 06, 2017

Ehsan Sherkat  
Computer Science\Computer Science

Dear Ehsan,

**REB #:** 2016-4012

**Project Title:** User evaluation of the interactive document clustering system

**Expiry Date:** December 23, 2018

The Social Sciences & Humanities Research Ethics Board has reviewed your annual report and has approved continuing approval of this project up to the expiry date (above).

REB approval is only effective for up to 12 months (as per TCPS article 6.14) after which the research requires additional review and approval for a subsequent period of up to 12 months. Prior to the expiry of this approval, you are responsible for submitting an annual report to further renew REB approval. Forms are available on the Research Ethics website.

I am also including a reminder (below) of your other on-going research ethics responsibilities with respect to this research.

Sincerely,



Dr. Karen Beazley, Chair

---

**Post REB Approval: On-going Responsibilities of Researchers**

After receiving ethical approval for the conduct of research involving humans, there are several ongoing responsibilities that researchers must meet to remain in compliance with University and Tri-Council policies.

1. Reporting adverse events

Any significant adverse events experienced by research participants must be reported **in writing** to Research Ethics within **24 hours** of their occurrence. Examples of what might be considered "significant" include: an emotional breakdown of a participant during and interview, a negative physical reaction by a participant (e.g. fainting, nausea, unexpected pain, allergic reaction), report by a participant of some sort of negative repercussion from their participation (e.g. reaction of spouse or employer) or a complaint by a participant with respect to their participation. The above list is indicative but not all-inclusive. The written report must include details of the adverse event and actions taken by the researcher in response to the incident.

#### 2. Seeking approval for protocol / consent form changes

Prior to implementing any changes to your research plan, whether to the study design, methods, consent form, or study instruments, researchers must submit a description of proposed changes to the Research Ethics Board for review and approval. This is done by completing an Amendment Request (available on the Research Ethics website). Please note that no reviews are conducted in August.

#### 3. Submitting annual reports

Ethics approvals are valid for up to 12 months. Prior to the end of the project's approval deadline, the researcher must complete an Annual Report (available on the website) and return it to Research Ethics for review and approval before the approval end date in order to prevent a lapse of ethics approval for the research. Researchers should note that no research involving humans may be conducted in the absence of a valid ethical approval and that allowing REB approval to lapse is a violation of University policy, inconsistent with the TCPS (article 6.14) and may result in suspension of research and research funding, as required by the funding agency.

#### 4. Submitting final reports

When the researcher is confident that no further data collection or participant contact will be required, a Final Report (available on the website) must be submitted to Research Ethics. After review and approval of the Final Report, the Research Ethics file will be closed.

#### 5. Retaining records in a secure manner

Researchers must ensure that both during and after the research project, data is securely retained and/or disposed of in such a manner as to comply with confidentiality provisions specified in the protocol and consent forms. This may involve destruction of the data, or continued arrangements for secure storage. Casual storage of old data is not acceptable.

It is the Principal Investigator's responsibility to keep a copy of the REB approval letters. This can be important to demonstrate that research was undertaken with Board approval, which can be a requirement to publish.

Please note that the University will securely store your REB project file for 5 years after the study closure date at which point the file records may be permanently destroyed.

#### 6. Current contact information and university affiliation

The Principal Investigator must inform the Research Ethics office of any changes to contact information for the PI (and supervisor, if appropriate), especially the electronic mail address, for the duration of the REB approval. The PI must inform Research Ethics if there is a termination or interruption of his or her affiliation with Dalhousie University.

#### 7. Legal Counsel

The Principal Investigator agrees to comply with all legislative and regulatory requirements that apply to the project. The Principal Investigator agrees to notify the University Legal Counsel office in the event that he or she receives a notice of non-compliance, complaint or other proceeding relating to such requirements.

#### 8. Supervision of students

Faculty must ensure that students conducting research under their supervision are aware of their responsibilities as described above, and have adequate support to conduct their research in a safe and ethical manner.



## Appendix C

### Copyright Permissions

In this section, the copyright permissions of the publications is provided.

## Consent to Publish

### Lecture Notes in Computer Science



Title of the Book or Conference Name: Natural Language Processing and Information Systems . . .  
 Volume Editor(s): Flavius Frasincar, Ashwin. Ittoo, Le Minh Nguyen, and Elisabeth Métais  
 Title of the Contribution: Vector Embedding of Wikipedia Concepts and Entities . . . . .  
 Author(s) Name(s): Ehsan Sherkat, Evangelos E. Milios . . . . .  
 Corresponding Author's Name, Address, Affiliation and Email: Ehsan Sherkat . . . . .  
 .ehsansherkat@dal.ca, Dalhousie University, Halifax, Canada . . . . .  
 . . . . .

When Author is more than one person the expression "Author" as used in this agreement will apply collectively unless otherwise indicated.

#### § 1 Rights Granted

Author hereby grants and assigns to Springer International Publishing AG, Cham (hereinafter called Springer) the exclusive, sole, permanent, world-wide, transferable, sub-licensable and unlimited right to reproduce, publish, distribute, transmit, make available or otherwise communicate to the public, translate, publicly perform, archive, store, lease or lend and sell the Contribution or parts thereof individually or together with other works in any language, in all revisions and versions (including soft cover, book club and collected editions, anthologies, advance printing, reprints or print to order, microfilm editions, audiograms and videograms), in all forms and media of expression including in electronic form (including offline and online use, push or pull technologies, use in databases and networks for display, print and storing on any and all stationary or portable end-user devices, e.g. text readers, audio, video or interactive devices, and for use in multimedia or interactive versions as well as for the display or transmission of the Contribution or parts thereof in data networks or search engines), in whole, in part or in abridged form, in each case as now known or developed in the future, including the right to grant further time-limited or permanent rights. For the purposes of use in electronic forms, Springer may adjust the Contribution to the respective form of use and include links or otherwise combine it with other works. For the avoidance of doubt, Springer has the right to permit others to use individual illustrations and may use the Contribution for advertising purposes.

The copyright of the Contribution will be held in the name of Springer. Springer may take, either in its own name or in that of copyright holder, any necessary steps to protect these rights against infringement by third parties. It will have the copyright notice inserted into all editions of the Contribution according to the provisions of the Universal Copyright Convention (UCC) and dutifully take care of all formalities in this connection in the name of the copyright holder.

#### § 2 Regulations for Authors under Special Copyright Law

The parties acknowledge that there may be no basis for claim of copyright in the United States to a Contribution prepared by an officer or employee of the United States government as part of that person's official duties. If the Contribution was performed under a United States government contract, but Author is not a United States government employee, Springer grants the United States government royalty-free permission to reproduce all or part of the Contribution and to authorize others to do so for United States government purposes.

If the Contribution was prepared or published by or under the direction or control of Her Majesty (i.e., the constitutional monarch of the Commonwealth realm) or any Crown government department, the copyright in the Contribution shall, subject to any agreement with Author, belong to Her Majesty.

If the Contribution was created by an employee of the European Union or the European Atomic Energy Community (EU/Euratom) in the performance of their duties, the regulation 31/EEC, 11/EAEC (Staff Regulations) applies, and copyright in the Contribution shall, subject to the Publication Framework Agreement (EC Plug), belong to the European Union or the European Atomic Energy Community.

If Author is an officer or employee of the United States government, of the Crown, or of EU/Euratom, reference will be made to this status on the signature page.

#### § 3 Rights Retained by Author

Author retains, in addition to uses permitted by law, the right to communicate the content of the Contribution to other scientists, to share the Contribution with them in manuscript form, to perform or present the Contribution or to use the content for non-commercial internal and educational purposes, provided the Springer publication is mentioned as the

original source of publication in any printed or electronic materials. Author retains the right to republish the Contribution in any collection consisting solely of Author's own works without charge subject to ensuring that the publication by Springer is properly credited and that the relevant copyright notice is repeated verbatim.

Author may self-archive an author-created version of his/her Contribution on his/her own website and/or the repository of Author's department or faculty. Author may also deposit this version on his/her funder's or funder's designated repository at the funder's request or as a result of a legal obligation. He/she may not use the publisher's PDF version, which is posted on SpringerLink and other Springer websites, for the purpose of self-archiving or deposit. Furthermore, Author may only post his/her own version, provided acknowledgment is given to the original source of publication and a link is inserted to the published article on Springer's website. The link must be provided by inserting the DOI number of the article in the following sentence: "The final publication is available at Springer via [http://dx.doi.org/\[insert DOI\]](http://dx.doi.org/[insert DOI])". The DOI (Digital Object Identifier) can be found at the bottom of the first page of the published paper.

Prior versions of the Contribution published on non-commercial pre-print servers like ArXiv/CoRR and HAL can remain on these servers and/or can be updated with Author's accepted version. The final published version (in pdf or html/xml format) cannot be used for this purpose. Acknowledgment needs to be given to the final publication and a link must be inserted to the published Contribution on Springer's website, by inserting the DOI number of the article in the following sentence: "The final publication is available at Springer via [http://dx.doi.org/\[insert DOI\]](http://dx.doi.org/[insert DOI])".

Author retains the right to use his/her Contribution for his/her further scientific career by including the final published paper in his/her dissertation or doctoral thesis provided acknowledgment is given to the original source of publication. Author also retains the right to use, without having to pay a fee and without having to inform the publisher, parts of the Contribution (e.g. illustrations) for inclusion in future work, and to publish a substantially revised version (at least 30% new content) elsewhere, provided that the original Springer Contribution is properly cited.

#### § 4 Warranties

Author warrants that the Contribution is original except for such excerpts from copyrighted works (including illustrations, tables, animations and text quotations) as may be included with the permission of the copyright holder thereof, in which case(s) Author is required to obtain written permission to the extent necessary and to indicate the precise sources of the excerpts in the manuscript. Author is also requested to store the signed permission forms and to make them available to Springer if required.

Author warrants that he/she is entitled to grant the rights in accordance with Clause 1 "Rights Granted", that he/she has not assigned such rights to third parties, that the Contribution has not heretofore been published in whole or in part, that the Contribution contains no libelous statements and does not infringe on any copyright, trademark, patent, statutory right or proprietary right of others, including rights obtained through licenses; and that Author will indemnify Springer against any costs, expenses or damages for which Springer may become liable as a result of any breach of this warranty.

#### § 5 Delivery of the Work and Publication

Author agrees to deliver to the responsible Volume Editor (for conferences, usually one of the Program Chairs), on a date to be agreed upon, the manuscript created according to the Springer Instructions for Authors. Springer will undertake the reproduction and distribution of the Contribution at its own expense and risk. After submission of the Consent to Publish form Signed by the Corresponding Author, changes of authorship, or in the order of the authors listed, will not be accepted by Springer.

#### § 6 Author's Discount

Author is entitled to purchase for his/her personal use (directly from Springer) the Work or other books published by Springer at a discount of 40% off the list price as long as there is a contractual arrangement between Author and Springer and subject to applicable book price regulation. Resale of such copies or of free copies is not permitted.

#### § 7 Governing Law and Jurisdiction

This agreement shall be governed by, and shall be construed in accordance with, the laws of Switzerland. The courts of Zug, Switzerland shall have the exclusive jurisdiction.

Corresponding Author signs for and accepts responsibility for releasing this material on behalf of any and all Co-authors.

**Signature of Corresponding Author:**

**Date:**

March 20, 2017

I'm an employee of the US Government and transfer the rights to the extent transferable (Title 17 §105 U.S.C. applies)

I'm an employee of the Crown and copyright on the Contribution belongs to Her Majesty

I'm an employee of the EU or Euratom and copyright on the Contribution belongs to EU or Euratom

## § 2 Rights Retained by Author

Author retains, in addition to uses permitted by law, the right to communicate the content of the Contribution to other research colleagues, to share the Contribution with them in manuscript form, to perform or present the Contribution or to use the content for non-commercial internal and educational purposes, provided the original source of publication is cited according to the current citation standards in any printed or electronic materials. Author retains the right to republish the Contribution in any collection consisting solely of Author's own works without charge, subject to ensuring that the publication of the Publisher is properly credited and that the relevant copyright notice is repeated verbatim. Author may self-archive an author-created version of his/her Contribution on his/her own website and/or the repository of Author's department or faculty. Author may also deposit this version on his/her funder's or funder's designated repository at the funder's request or as a result of a legal obligation. He/she may not use the Publisher's PDF version, which is posted on the Publisher's platforms, for the purpose of self-archiving or deposit. Furthermore, Author may only post his/her own version, provided acknowledgment is given to the original source of publication and a link is inserted to the published article on the Publisher's website. The link must be provided by inserting the DOI number of the article in the following sentence: "The final authenticated version is available online at [https://doi.org/\[insert DOI\]](https://doi.org/[insert DOI])." The DOI (Digital Object Identifier) can be found at the bottom of the first page of the published paper.

Prior versions of the Contribution published on non-commercial pre-print servers like ArXiv/CoRR and HAL can remain on these servers and/or can be updated with Author's accepted version. The final published version (in pdf or html/xml format) cannot be used for this purpose. Acknowledgment needs to be given to the final publication and a link must be inserted to the published Contribution on the Publisher's website, by inserting the DOI number of the article in the following sentence: "The final authenticated publication is available online at [https://doi.org/\[insert DOI\]](https://doi.org/[insert DOI])".

Author retains the right to use his/her Contribution for his/her further scientific career by including the final published paper in his/her dissertation or doctoral thesis provided acknowledgment is given to the original source of publication. Author also retains the right to use, without having to pay a fee and without having to inform the Publisher, parts of the Contribution (e.g. illustrations) for inclusion in future work. Authors may publish an extended version of their proceedings paper as a journal article provided the following principles are adhered to: a) the extended version includes at least 30% new material, b) the original publication is cited, and c) it includes an explicit statement about the increment (e.g., new results, better description of materials, etc.).

## § 3 Warranties

Author agrees, at the request of Publisher, to execute all documents and do all things reasonably required by Publisher in order to confer to Publisher all rights intended to be granted under this Agreement. Author warrants that the Contribution is original except for such excerpts from copyrighted works (including illustrations, tables, animations and text quotations) as may be included with the permission of the copyright holder thereof, in which case(s) Author is required to obtain written permission to the extent necessary and to indicate the precise sources of the excerpts in the manuscript. Author is also requested to store the signed permission forms and to make them available to Publisher if required.

Author warrants that Author is entitled to grant the rights in accordance with Clause 1 "Rights Granted", that Author has not assigned such rights to third parties, that the Contribution has not heretofore been published in whole or in part, that the Contribution contains no libellous or defamatory statements and does not infringe on any copyright, trademark, patent, statutory right or proprietary right of others, including rights obtained through licences; and that Author will indemnify Publisher against any costs, expenses or damages for which Publisher may become liable as a result of any claim which, if true, would constitute a breach by Author of any of Author's representations or warranties in this Agreement.

Author agrees to amend the Contribution to remove any potential obscenity, defamation, libel, malicious falsehood or otherwise unlawful part(s) identified at any time. Any such removal or alteration shall not affect the warranty and indemnity given by Author in this Agreement.

## § 4 Delivery of Contribution and Publication

Author agrees to deliver to the responsible Volume Editor (for conferences, usually one of the Program Chairs), on a date to be agreed upon, the manuscript created according to the Publisher's Instructions for Authors. Publisher will undertake the reproduction and distribution of the Contribution at its own expense and risk. After submission of the Consent to Publish form signed by the Corresponding Author, changes of authorship, or in the order of the authors listed, will not be accepted by the Publisher.

**§ 5 Author's Discount for Books and Electronic Access**

Author is entitled to purchase for his/her personal use (if ordered directly from Publisher) the Work or other books published by Publisher at a discount of 40% off the list price for as long as there is a contractual arrangement between Author and Publisher and subject to applicable book price regulation.

Resale of such copies is not permitted.

**§ 6 Governing Law and Jurisdiction**

If any difference shall arise between Author and Publisher concerning the meaning of this Agreement or the rights and liabilities of the parties, the parties shall engage in good faith discussions to attempt to seek a mutually satisfactory resolution of the dispute. This agreement shall be governed by, and shall be construed in accordance with, the laws of Switzerland. The courts of Zug, Switzerland shall have the exclusive jurisdiction.


Corresponding Author signs for and accepts responsibility for releasing this material on behalf of any and all Co-Authors.

**Signature of Corresponding Author:**

Ehsan Sherkat

**Date:**

June 17, 2018

.....  .....

- I'm an employee of the US Government and transfer the rights to the extent transferable (Title 17 §105 U.S.C. applies)
- I'm an employee of the Crown and copyright on the Contribution belongs to the Crown

*For internal use only:*

Legal Entity Number: 1128 Springer International Publishing AG  
Springer-C-CTP-01/2018

## ACM Copyright and Audio/Video Release

**Title of the Work:** Interactive Document Clustering Revisited: A Visual Analytics Approach

**Submission ID:**fp1180

**Author/Presenter(s):** Ehsan Sherkat: ; Seyednaser Nourashrafeddin: ; Evangelos Milios: ; Rosane Minghim:

**Type of material:**Full Paper

**Publication and/or Conference Name:** IUI'18: 23rd International Conference on Intelligent User Interfaces Proceedings

### I. Copyright Transfer, Reserved Rights and Permitted Uses

\* Your Copyright Transfer is conditional upon you agreeing to the terms set out below.

Copyright to the Work and to any supplemental files integral to the Work which are submitted with it for review and publication such as an extended proof, a PowerPoint outline, or appendices that may exceed a printed page limit, (including without limitation, the right to publish the Work in whole or in part in any and all forms of media, now or hereafter known) is hereby transferred to the ACM (for Government work, to the extent transferable) effective as of the date of this agreement, on the understanding that the Work has been accepted for publication by ACM.

#### Reserved Rights and Permitted Uses

(a) All rights and permissions the author has not granted to ACM are reserved to the Owner, including all other proprietary rights such as patent or trademark rights.

(b) Furthermore, notwithstanding the exclusive rights the Owner has granted to ACM, Owner shall have the right to do the following:

(i) Reuse any portion of the Work, without fee, in any future works written or edited by the Author, including books, lectures and presentations in any and all media.

(ii) Create a "[Major Revision](#)" which is wholly owned by the author

(iii) Post the Accepted Version of the Work on (1) the Author's home page, (2) the Owner's institutional repository, (3) any repository legally mandated by an agency funding the research on which the Work is based, and (4) any non-commercial repository or aggregation that does not duplicate ACM tables of contents, i.e., whose patterns of links do not substantially duplicate an ACM-copyrighted volume or issue. Non-commercial repositories are here understood as repositories owned by non-profit organizations that do not charge a fee for accessing deposited articles and that do not sell advertising or otherwise profit from serving articles.

(iv) Post an "[Author-Izer](#)" link enabling free downloads of the Version of Record in the ACM Digital Library on (1) the Author's home page or (2) the Owner's institutional repository;

(v) Prior to commencement of the ACM peer review process, post the version of the Work as submitted to ACM ("[Submitted Version](#)" or any earlier versions) to non-peer reviewed servers;

(vi) Make free distributions of the final published Version of Record internally to the Owner's employees, if applicable;

(vii) Make free distributions of the published Version of Record for Classroom and Personal Use;

(viii) Bundle the Work in any of Owner's software distributions; and



(ix) Use any Auxiliary Material independent from the Work.

When preparing your paper for submission using the ACM TeX templates, the rights and permissions information and the bibliographic strip must appear on the lower left hand portion of the first page.

The new [ACM Consolidated TeX template Version 1.3 and above](#) automatically creates and positions these text blocks for you based on the code snippet which is system-generated based on your rights management choice and this particular conference.

*Please copy and paste the following code snippet into your TeX file between `\begin{document}` and `\maketitle`, either after or before CCS codes.*

```
\copyrightyear{2018}
\acmYear{2018}
\setcopyright{acmcopyright}
\acmConference[IUI'18]{23rd International Conference on Intelligent User
Interfaces}{March 7--11, 2018}{Tokyo, Japan}
\acmBooktitle{IUI'18: 23rd International Conference on Intelligent User
Interfaces, March 7--11, 2018, Tokyo, Japan}
\acmPrice{15.00}
\acmDOI{10.1145/3172944.3172964}
\acmISBN{978-1-4503-4945-1/18/03}
```

ACM TeX template .cls version 2.8, automatically creates and positions these text blocks for you based on the code snippet which is system-generated based on your rights management choice and this particular conference.

*Please copy and paste the following code snippet into your TeX file between `\begin{document}` and `\maketitle`, either after or before CCS codes.*

```
\CopyrightYear{2018}
\setcopyright{acmcopyright}
\conferenceinfo{IUI'18.}{March 7--11, 2018, Tokyo, Japan}
\isbn{978-1-4503-4945-1/18/03}\acmPrice{$15.00}
\doi{https://doi.org/10.1145/3172944.3172964}
```

*If you are using the ACM Microsoft Word template, or still using an older version of the ACM TeX template, or the current versions of the ACM SIGCHI, SIGGRAPH, or SIGPLAN TeX templates, you must copy and paste the following text block into your document as per the instructions provided with the templates you are using:*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

IUI'18, March 7–11, 2018, Tokyo, Japan  
 © 2018 Association for Computing Machinery.  
 ACM ISBN 978-1-4503-4945-1/18/03...\$15.00  
<https://doi.org/10.1145/3172944.3172964>

*NOTE: Make sure to include your article's DOI as part of the bibstrip data; DOIs will be registered and become active shortly after publication in the ACM Digital Library*

A. Assent to Assignment. I hereby represent and warrant that I am the sole owner (or authorized agent of the copyright owner(s)), with the exception of third party materials detailed in section III below. I have obtained permission for any third-party material included in the Work.

B. Declaration for Government Work. I am an employee of the National Government of my country and my Government claims rights to this work, or it is not copyrightable (Government work is classified as Public Domain in U.S. only)

Are any of the co-authors, employees or contractors of a National Government?  Yes  No

---

## II. Permission For Conference Recording and Distribution

\* Your Audio/Video Release is conditional upon you agreeing to the terms set out below.

I hereby grant permission for ACM to include my name, likeness, presentation and comments in any and all forms, for the Conference and/or Publication.

I further grant permission for ACM to record and/or transcribe and reproduce my presentation as part of the ACM Digital Library, and to distribute the same for sale in complete or partial form as part of an ACM product on CD-ROM, DVD, webcast, USB device, streaming video or any other media format now or hereafter known.

I understand that my presentation will not be sold separately as a stand-alone product without my direct consent. Accordingly, I give ACM the right to use my image, voice, pronouncements, likeness, and my name, and any biographical material submitted by me, in connection with the Conference and/or Publication, whether used in excerpts or in full, for distribution described above and for any associated advertising or exhibition.

Do you agree to the above Audio/Video Release?  Yes  No

## III. Auxiliary Material

Do you have any Auxiliary Materials?  Yes  No

## IV. Third Party Materials

In the event that any materials used in my presentation or Auxiliary Materials contain the work of third-party individuals or organizations (including copyrighted music or movie excerpts or anything not owned by me), I understand that it is my responsibility to secure any necessary permissions and/or licenses for print and/or digital publication, and cite or attach them below.



- We/I have not used third-party material.  
 We/I have used third-party materials and have necessary permissions.

#### **V. Artistic Images**

If your paper includes images that were created for any purpose other than this paper and to which you or your employer claim copyright, you must complete Part V and be sure to include a notice of copyright with each such image in the paper.

- We/I do not have any artistic images.  
 We/I have any artistic images.

#### **VI. Representations, Warranties and Covenants**

The undersigned hereby represents, warrants and covenants as follows:

- (a) Owner is the sole owner or authorized agent of Owner(s) of the Work;
- (b) The undersigned is authorized to enter into this Agreement and grant the rights included in this license to ACM;
- (c) The Work is original and does not infringe the rights of any third party; all permissions for use of third-party materials consistent in scope and duration with the rights granted to ACM have been obtained, copies of such permissions have been provided to ACM, and the Work as submitted to ACM clearly and accurately indicates the credit to the proprietors of any such third-party materials (including any applicable copyright notice), or will be revised to indicate such credit;
- (d) The Work has not been published except for informal postings on non-peer reviewed servers, and Owner covenants to use best efforts to place ACM DOI pointers on any such prior postings;
- (e) The Auxiliary Materials, if any, contain no malicious code, virus, trojan horse or other software routines or hardware components designed to permit unauthorized access or to disable, erase or otherwise harm any computer systems or software; and
- (f) The Artistic Images, if any, are clearly and accurately noted as such (including any applicable copyright notice) in the Submitted Version.

I agree to the Representations, Warranties and Covenants

#### **Funding Agents**

1. Natural Sciences and Engineering Research Council of Canada (NSERC) award number(s):  
 CRDPJ 499941-16

DATE: **12/12/2017** sent to ehsansherkat@dal.ca at **16:12:41**

## ACM Copyright Form and Audio/Video Release

**Title of the Work:** A Visual Analytics Approach for Interactive Document Clustering

**Author/Presenter(s):** Ehsan Sherkat(Dalhousie University); Evangelos Milios(Dalhousie University); Rosane Minghim(University of São Paulo)

**Type of material:**Special Issue on IUI 2018

**Publication:** ACM Transactions on Interactive Intelligent Systems

### I. Copyright Transfer, Reserved Rights and Permitted Uses

\* Your Copyright Transfer is conditional upon you agreeing to the terms set out below.

Copyright to the Work and to any supplemental files integral to the Work which are submitted with it for review and publication such as an extended proof, a PowerPoint outline, or appendices that may exceed a printed page limit, (including without limitation, the right to publish the Work in whole or in part in any and all forms of media, now or hereafter known) is hereby transferred to the ACM (for Government work, to the extent transferable) effective as of the date of this agreement, on the understanding that the Work has been accepted for publication by ACM.

#### Reserved Rights and Permitted Uses

(a) All rights and permissions the author has not granted to ACM are reserved to the Owner, including all other proprietary rights such as patent or trademark rights.

(b) Furthermore, notwithstanding the exclusive rights the Owner has granted to ACM, Owner shall have the right to do the following:

(i) Reuse any portion of the Work, without fee, in any future works written or edited by the Author, including books, lectures and presentations in any and all media.

(ii) Create a "Major Revision" which is wholly owned by the author

(iii) Post the Accepted Version of the Work on (1) the Author's home page, (2) the Owner's institutional repository, or (3) any repository legally mandated by an agency funding the research on which the Work is based.

(iv) Post an "Author-Izer" link enabling free downloads of the Version of Record in the ACM Digital Library on (1) the Author's home page or (2) the Owner's institutional repository;

(v) Prior to commencement of the ACM peer review process, post the version of the Work as submitted to ACM ("Submitted Version" or any earlier versions) to non-peer reviewed servers;

(vi) Make free distributions of the final published Version of Record internally to the Owner's employees, if applicable;

(vii) Make free distributions of the published Version of Record for Classroom and Personal Use;

(viii) Bundle the Work in any of Owner's software distributions; and

(ix) Use any Auxiliary Material independent from the Work.

When preparing your paper for submission using the ACM templates, you will need to include the rights management and bibstrip text blocks below to the lower left hand portion of the first page. As this text will provide rights information for your paper, please make sure that this text is displayed and positioned correctly when you submit your manuscript for publication.

Authors should understand that consistent with ACM's policy of encouraging dissemination of information, each work published by ACM appears with a copyright and the following notice:

*If you are using Authorized ACM TeX templates, the following code will generate the proper statements based on your rights choices. Please copy and paste it into your TeX file between `\begin{document}` and `\maketitle`, either after or before CCS codes.*

```
\setcopyright{acmcopyright}
\acmJournal{TIIS}
\acmYear{2018} \acmVolume{1} \acmNumber{1} \acmArticle{1}
\acmMonth{1} \acmPrice{15.00}
```

*If you are using Word, copy and paste these words in the space provided at the bottom of your first page:*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

Copyright © ACM 2018 2160-6455/2018/MonthOfPublication -  
ArticleNumber \$15.00

NOTE: DOIs will be registered and become active shortly after publication in the ACM Digital Library

A. Assent to Assignment. I hereby represent and warrant that I am the sole owner (or authorized agent of the copyright owner(s)), with the exception of third party materials detailed in section III below. I have obtained permission for any third-party

material included in the Work.

B. Declaration for Government Work. I am an employee of the National Government of my country and my Government claims rights to this work, or it is not copyrightable (Government work is classified as Public Domain in U.S. only)

Are any of the co-authors, employees or contractors of a National Government?

Yes  No

Country:

## II. PERMISSION FOR CONFERENCE TAPING AND DISTRIBUTION (Check A and, if applicable, B)

### A. Audio /Video Release

I hereby grant permission for ACM to include my name, likeness, presentation and comments in any and all forms, for the Conference and/or Publication.

I further grant permission for ACM to record and/or transcribe and reproduce my presentation as part of the ACM Digital Library, and to distribute the same for sale in complete or partial form as part of an ACM product on CD-ROM, DVD, webcast, USB device, streaming video or any other media format now or hereafter known.

I understand that my presentation will not be sold separately by itself as a stand-alone product without my direct consent. Accordingly, I give ACM the right to use my image, voice, pronouncements, likeness, and my name, and any biographical material submitted by me, in connection with the Conference and/or Publication, whether used in excerpts or in full, for distribution described above and for any associated advertising or exhibition.

Do you agree to the above Audio/Video Release?  Yes  No

### III. Auxiliary Materials, not integral to the Work

\* Your Auxiliary Materials Release is conditional upon you agreeing to the terms set out below.

[Defined as additional files, including software and executables that are not submitted for review and publication as an integral part of the Work but are supplied by the author as useful resources for the reader.]

I hereby grant ACM permission to serve files containing my Auxiliary Material from the ACM Digital Library. I hereby represent and warrant that any of my Auxiliary Materials do not knowingly and surreptitiously incorporate malicious code, virus, trojan horse or other software routines or hardware components designed to permit unauthorized access or to disable, erase or otherwise harm any computer systems or software.

I agree to the above Auxiliary Materials permission statement.

This software is knowingly designed to illustrate technique(s) intended to defeat a system's security. The code has been explicitly documented to state this fact.

### IV. Third Party Materials

In the event that any materials used in my presentation or Auxiliary Materials contain the work of third-party individuals or organizations (including copyrighted

music or movie excerpts or anything not owned by me), I understand that it is my responsibility to secure any necessary permissions and/or licenses for print and/or digital publication, and cite or attach them below.

- We/I have not used third-party material.  
 We/I have used third-party materials and have necessary permissions.

#### **V. Artistic Images**

If your paper includes images that were created for any purpose other than this paper and to which you or your employer claim copyright, you must complete Part IV and be sure to include a notice of copyright with each such image in the paper.

- We/I do not have any artistic images.  
 We/I have have any artistic images.

---

#### **VI. Representations, Warranties and Covenants**

The undersigned hereby represents, warrants and covenants as follows:

- (a) Owner is the sole owner or authorized agent of Owner(s) of the Work;
- (b) The undersigned is authorized to enter into this Agreement and grant the rights included in this license to ACM;
- (c) The Work is original and does not infringe the rights of any third party; all permissions for use of third-party materials consistent in scope and duration with the rights granted to ACM have been obtained, copies of such permissions have been provided to ACM, and the Work as submitted to ACM clearly and accurately indicates the credit to the proprietors of any such third-party materials (including any applicable copyright notice), or will be revised to indicate such credit;
- (d) The Work has not been published except for informal postings on non-peer reviewed servers, and Owner covenants to use best efforts to place ACM DOI pointers on any such prior postings;
- (e) The Auxiliary Materials, if any, contain no malicious code, virus, trojan horse or other software routines or hardware components designed to permit unauthorized access or to disable, erase or otherwise harm any computer systems or software; and
- (f) The Artistic Images, if any, are clearly and accurately noted as such (including any applicable copyright notice) in the Submitted Version.

- I agree to the Representations, Warranties and Covenants
- 

DATE: **07/18/2018** sent to ehsansherkat@dal.ca at **06:07:33**