DEGREE PROJECT IN MATHEMATICS,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2018

An Object-Oriented Data Analysis approach for text population

JOFFREY DUMONT-LE BRAZIDEC

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ENGINEERING SCIENCES

An Object-Oriented Data Analysis approach for text population

JOFFREY DUMONT-LE BRAZIDEC

Degree Projects in Mathematical Statistics (30 ECTS credits)
Degree Programme in Mathematics (120 credits)
KTH Royal Institute of Technology year 2018
Supervisor at Politecnico di Milano: Prof. Simone Vantini
Co-Supervisor at Politecnico di Milano: Dott.ssa Anna Calissano
Supervisor at KTH: Dr. Jimmy Olsson
Examiner at KTH: Dr. Jimmy Olsson

TRITA-SCI-GRU 2018:014
MAT-E 2018:04

Royal Institute of Technology
School of Engineering Sciences
KTH SCI
SE-100 44 Stockholm, Sweden
URL: www.kth.se/sci

Sammanfattning

Med ständigt ökande tillgänglighet av textvärd data ökar behovet att kunna klustra och klassificera denna data. I detta arbete utvecklar vi statistiska verktyg för hypotestestning, klustring och klassificering av textvärd data inom ramen för objektorienterad dataanalys. Projektet inkluderar forskning på semantiska metoder för att representera texter, jämförelser mellan representationer, avstånd för sådana representationer och prestanda hos permutationstest. De viktigaste metoderna som jämförs är vektorrumsmodeller och ämnesmodeller. Mer specifikt tillhandahåller detta arbete en algoritm för permutationstest, på dokument- eller meningsnivå, i syfte att pröva hypotesen att två texter har samma fördelning med avseende på olika representationer och avstånd. Till sist används en trädrepresentation för att beskriva studiet av texter ur en syntaktisk synvinkel.

Abstract

With more and more digital text-valued data available, the need to be able to cluster, classify and study them arises. We develop in this thesis statistical tools to perform null hypothesis testing and clustering or classification on text-valued data in the framework of Object-Oriented Data Analysis.

The project includes research on semantic methods to represent texts, comparisons between representations, distances for such representations and performance of permutation tests. Main methods compared are Vector Space Model and topic model. More precisely, this thesis will provide an algorithm to compute permutation tests at document or sentence level to study the equality in terms of distribution of two texts for different representations and distances.

Lastly, we describe the study of texts regarding a syntactic point of view and its structure with a tree representation.

Acknowledgments

On the very outset of this report, I would like to extend my sincere gratitude towards all the people who have helped me through the writing of this thesis.
Without their active guidance or help, I would not have been able to produce this paper.

I renew unequivocally that I am indebted to you, Anna Calissano and Simone Vantini for their guidance and encouragement to accomplish this assignment.

I am very grateful to the university of Politecnico di Milano for its hosting during this term and its support on the completion of this project, and to all the PhD candidates that have been there throughout my stay in Italy.

I also acknowledge the university of Kungliga Tekniska Högskolan from Stockholm and my examiner Jimmy Olsson for the presentation and especially for the year that I passed in Stockholm as a student in Applied and computational mathematics.

I am of course very grateful towards my parents and my family for their moral and especially economical support during the whole of my studies in France and abroad.

Gratitude goes also to my friends that directly or indirectly helped me to complete this project.

Any omission of a close or a less close relative here does not mean a lack of gratitude.

Thanks.

Contents

List of Figures

List of Tables

Introduction

Text-valued digital data have invaded our everyday lives. Twitter, Facebook are of course very obvious examples but even speeches of presidents, articles from newspapers about economy or international relationships today are present under digital shape and can then be studied by computers. Consequently a higher and higher access to texts such as the ones found on social networks, newspapers, web engine or books brings the need to be able to study, represent, cluster and classify this data for different benefits.

Texts today are already being studied. For example a query on a web search engine will be associated to ranked results after having been classified. However, the aim of this thesis is to bring the point of view of a text as a statistic object as proposed in (Marron and Alonso 2014) in the framework of Object-Oriented Data Analysis for a lot of other topics. Regarding this aim, we wish to create a permutation test able to test the statistical equality of two texts. Sixty years ago, Isaac Asimov wrote a science-fiction novel where he describes a scientist using a machine able to analyze the text of a diplomat or a politician and to extract the true meaning behind their speech. This machine is thus able to remove all the useless words or meaningless parts. The "understandable" transcript then produced reflects the true meaning of the text. It shows the "truth behind the form", or what the person is wiling to express. He will later discover that there is no meaning to the speech the politician had given and that everything is pure form and meaningless words. The real substance of the speech was equal to a text without substance. The semantic was reduced to nothing. Our work could perfectly fit this idea and we are able to give a simple answer at the end of this thesis about how we could process such a text. Testing the equality of two population of speeches in a proper way could lead to such things described above.

To perform such tests, we will use object-oriented data analysis where we will apply statistical tools to non-euclidean data. To do so, we will have to find a representation of texts which fit in the euclidean frame. Only then, we will be able to apply distances and so permutation tests on our data. We will also provide useful methods and especially algorithms for clustering and classification of texts to complete our study. A discussion about the pertinence, the benefits and the possibilities of these methods will be explored.

All this will be applied to speeches data. We have indeed decided to apply our methods to the State of the Union address, given by three different presidents (Clinton, Bush, Obama). We will describe these speeches later on.

The first section of this thesis will be a description of the framework: texts in Object-Oriented Data Analysis, and more specifically in our data. The rest of the paper is organised as follows. Section 2 outlines the state of the art about the multiple existing ways of text representation, where a particular attention will be given to a repartition in different sets of methods. Section 3 introduces distances for text representations with a particular focus on what will be applied to our representations. Section 4 introduces the permutations methods, their frameworks, the statistical tests which will be used and some discussions around the p-value. Section 5 presents the clustering methods and their frameworks, K-means and hierarchical clustering. Section 6 presents the complete statistical methodology that we propose to test data. Finally Section 7 is fully dedicated to the analysis of the presidents' speeches that constitute our data and the analysis or the performance of the proposed methods. A discussion and a conclusion leading to future work will be presented in the final section.

1 Object-Oriented Data Analysis

1.1 OODA and Text-valued Data

object-oriented data analysis (OODA), whose mathematical structure was introduced in (Wang and Marron 2007), is the statistical analysis of data sets of complex objects as explained in the introduction paper (Marron and Alonso 2014). The philosophy described in it is about complex data likely to be processed by statistical methods but which do not belong to the euclidean frame. This kind of complex data can be trees such as in (Wang and Marron 2007), curves in (Marron and Alonso 2014) or image analysis (Marron and Alonso 2014) and (Wei, C. Lee, and Marron 2016), networks, covariances matrix in (Dryden, Koloydenko, and Zhou 2009) or again texts, which we are focusing on.
As it is noted by Marron in his paper, the main problem of big data is not ultimately their more and more widespread use but their complexity that has become more and more important. The complexity of these data and the challenge their study represents is expressed as a task for especially the field of statistics and mathematics. Note that the notion of OODA has already been raised from a computer science point of view in (Rademakers 1997), but the fundamental difference here, the spirit of this framework is the ambition to study the topic from a mathematical point of view, putting the focus on key statistics. Marron's assumption is that mathematics should have a strong role in this field since it has a very great potential for inventing new statistical methodology, new methods for the study of object-oriented data, such as in (Cristianini and Shawe-Taylor 2000) or (Vapnik 1999).
Many examples of where mathematics could be useful to OODA are presented. In (Marron and Alonso 2014) is related the case of the support vector machine as an approach that is based without taking account of underlying probabilities and so lacks of a statistical point of view.
A well-known statistical procedure useful in this context in the Principal Component Analysis (PCA). In OODA usually a common first task is to define a center-point such as a median or a mean as explained in (Wang and Marron 2007) and this point will be expanded latter on in our paper. The second task is about defining variations of these objects in order to explain how the objects relate to each other (Wang and Marron 2007). That is very well done with a Principal Component Analysis (PCA).

Regarding texts i.e documents, sentences, tweets, posts, messages, it is clear that this kind of data is not immediately included within the euclidean frame. Texts are not points, do not belong to any classical geometry and their study can be set in the OODA frame. In other words statistical analysis of complex data needs new methods and can not be satisfied with standard statistical methods. The study of texts, their representations and their comparisons are highly topical issues of major importance. It is more and more explosive since the wake of computer science. This work is of importance for, by example, a web search engine. A web search engine often returns thousands of pages in response to a broad query, making it difficult for users to browse or to identify relevant information. Clustering methods can be used to automatically group the retrieved documents into a list of meaningful categories, as it is achieved by an open source software such as Carrot2 in (Osinski and Weiss n.d.). More broadly and to sum up, the growing amount of text-valued data (social networks, web search engine, web newspapers...) leads to the necessity to process them.

The first work in OODA is to study ways to describe data i.e study different representations of texts. The second work is about being able to describe relations between them i.e be able to apply statistics to them. As a result, we will deal with collections of texts which will be treated as i.i.d realisations of text-valued random variable.
The project of this paper is to give a methodology and tools to study these questions.
This work will thus try to provide discussion around the different main issues in text representation. A big point of this thesis is to develop statistical tools to perform null hypothesis testing and clustering or classification on text-valued data. First in document clustering that is the analysis of clusters of textual documents i.e the development of clustering algorithms in computational text analysis or again the algorithmically grouping of documents into a set of texts

that are called subsets or clusters where the algorithm's goal is to create internally coherent clusters that are distinct from one another such as explained in ("Introduction to Information Retrieval" 2016). Then in document classification that can be content-based (classification in libraries for example) or request-oriented (i.e classified to be found by a query in specific conditions).

1.2 Our Data : Texts as Speeches

Since our choice of study is in OODA, the data we have selected are texts of sufficient length and strong intern relation. The former is necessary for a good application of topic model representation that we will explain later on and the latter is because we want to provide useful exploitation of our data. The sample chosen are texts of the State of the Union Address, an annual message presented by the President of the United States to a joint session of the United States Congress. This choice gives us assurance that the samples are not too biased because the message is general, always destined to the same public at the same location and towards the same goal.
It is constituted of a total of 236 annual speeches. In order to provide accurate and clear results we have decided to work only on the speeches of Bill Clinton, George W. Bush, Barack Obama, so a total of 24 speeches. This basis could have been supported by Weekly Address speeches from the same presidents that are easily accessible. We will use this sample both at document and sentence level for different reasons that will be explained later.

Here we make a brief description of the three presidents whose speeches are our data scope.
Bill Clinton was elected president in 1992 and presided over the longest period of peacetime economic expansion in American history. In 1996, Clinton became the first Democrat to be elected to a second full term. Notable events during his presidency happen in 1993 (Explosion at the World Trade Center, Signature of NAFTA), in 1994 (Republican Party won unified control of the Congress) and in 1998 (Clinton–Lewinsky scandal and attempt of impeachment). His former chief speech writer was David Kusnet (1992-1994) and his latter Michael Waldman (1995-1999).
Georges Bush was elected president in 2000 and has been reelected in 2004. Georges Bush was from the the Republican Party. Notable events during his presidency happen in 2001 (the events of the 9/11 and in Afghanistan), in 2002 (constitution of the axis of the evil and the Iraq war), and in 2005 (Hurricane Katrina). His former chief speech writer was Michael Gerson (2001-2006) and his latter William McGurn (2006-2008).
Barack Obama was elected president in 2008 and reelected in 2012. He is a democrat president. Notable events during his presidency happen in 2009 (Nobel peace prize), in 2010 (Obamacare), in 2011 (Libyan war and end of the Iraq war with also the death of Ossama Bin Laden) and in 2013 (Edward Snowden reveals). Jon Favreau (2009-2013) has been the first chief speech writer of Obama and it was Cody Keenan (2013-2016) for the second part.

2 Text Representation: state of the art

In this part we introduce text representations, mathematical models for representing text documents in the OODA framework.Here we introduce some methods that could be used to present a fast chronological evolution of text representations. Vector Space Models for text representations have been mainly used since the 1900s in distributional semantics. Since then, we have seen the development of count-based models used for estimating continuous representations of words, such as Latent Semantic Analysis (LSA) and topic models such as Latent Dirichlet Allocation (LDA) being two such examples.
However, recent attempts to give improved representations of words and documents have been brought by models based on word embeddings and neural methods such as in (Mikolov and al. 2013) through the creation of Word2Vec.
Finally, the field of work which studies syntax is mainly focused on the creation of grammar and syntax trees on language.

2.1 Vector Space Model

Vector Space Model or term vector model is an algebraic model for representing text documents as vectors. In comparison of texts, the Sparse Bag of Words (sBoW) presented in many texts such as (Scott and Matwin 1999) is the simplest idea and the root of the semantic-based model. A Sparse Bag of Words-valued text is the list of the words in the text disregarding grammar and the order of the sentences but keeping the multiplicity. The usefulness of the Sparse Bag of Words model relates to the assumption that if documents have similar words and similar number of words then they tend to have similar meanings.

More generally, the Vector Space Models (VSM) presented in (Salton, Wong, and Yang 1975) reduces each document d_j in the corpus D to a vector of real numbers, each of which reflects the count of an unordered collection of words.
This is:

$$d_j = (w_{ij})_{1 \leq i \leq m}$$

with w_{ij} is the weight of the word i of the vocabulary of size m in d_j. Thus a co-occurrence term-document matrix **X** that we work on can be built with, in rows, the words and in columns the documents. The word-vector is a high-dimensional vector in which each element corresponds to a unique vocabulary term. The assumption of the representation is that semantically similar words will be mapped to nearby points. In this case the sBoW model is a Vector Space Model where each weight is simply equal to the count of each word in each document.

Since then, several approaches improving the quality of the basic VSM model that is the sBoW have been developed. The Tf-Idf numerical statistic is the most known and is an approach formulated in two times from two statistical interpretations. The first statistical interpretation is made by taking interest in the term's frequency in (Luhn 1957) based on the Luhn Assumption: the weight of a term that occurs in a document is simply proportional to the term's frequency (the TF part). The second assumption is proposed in (Sparck 1972) : the specificity of a term can be quantified as an inverse function of the number of documents in which it occurs. That leads to a statistical interpretation of term's specificity called Inverse Document Frequency (IDF).
The Tf-Idf model was developed, presented and tested for the first time in (Salton, Wong, and Yang 1975). The Term Frequency–Inverse document frequency is presented as :

$$\mathrm{tdf}(t_i, d_j, D) = \mathrm{tf}(t_i, d_j) * \mathrm{idf}(t_i, D) = w_{ij}$$

and represents a weight given to a word t_i in the document d_j of the corpus D. For a term t and a document d, the Tf part is the number of times t occurs in d such as in the sBoW model. The Idf part is used to quantify the number of times t occurs the corpus of documents D. Several functions can be used to measure and quantify this quantity and the influence on classification it should have. The main idea besides this Inverse Document Frequency is about diminishing the influence a very common word should have on classification of texts since a very common word will probably not bring any discrimination and therefore material to classify.

2.2 Distributional Semantic Models: Count-based Model

The methods of Distributional Semantic Models (DSM) are VSM with the particularity that they share the following assumption : words that appear in same contexts share the same meaning. It is called the distributional hypothesis.
The most obvious problem with Tf-Idf is that this method does not deal with synonyms and other related semantic problems. For this reason in (Deerwester, Dumais, and Furnas 1990) is developed the method of the Latent Semantic Analysis (LSA) that applies a Singular Value Decomposition (SVD) to our term-document weighted Tf-Idf matrix (or sBoW) in order to find a so-called latent semantic space that retains most of the variances in the corpus. Each feature in the new space is a linear combination of the original Tf-Idf features, which naturally handles the synonymy problem. This SVD will therefore drastically reduce the size of the co-occurrence matrix.

Other methods belonging to the range of Vector Space Model have been developed such as in (Gabrilovich and Markovitch 2007) who proposed to represent each word or text as a weighted vector of Wikipedia concepts. Let $< k_{ij} >$ be an index measuring the correlation between the term t_i and c_j where ${c_j}_{1 \leq j \leq M}$ is a list of M Wikipedia concepts. Then the semantic interpretation vector V of the document d is the vector :

$$\sum_i w_i \; k_{ij} \qquad (k_{ij})_{1 \leq j \leq M}$$

where w_i is the Tf-Idf weight i.e the vector of size M representing the relevance of the c_j concepts. In this case, we can create a matrix such as in the previous part but this matrix will not carry on term–document similarities but better on word–context similarities such as explained in (Turney and Pantel 2010).

Finally the Random Indexing method presented in (Sahlgren 2005) is presented as having very good properties and performances. This method can be described in two steps.The algorithm first assigns to each context (i.e word or document) a unique vector d-dimensioned called an index vector composed of a small number of -1,1 and the rest of 0. Then it scans through the text and each time a word occurs in a context the index vector is added to the word's context vector. And so words are represented by context vectors. From these context vectors it is thus possible to build an approximation of the term-document co-occurrence matrix $\mathbf{X}$. To perform text classification, the easiest possibility is thus to sum the context-vectors belonging to a document such as in (Sahlgren and Cöster 2004).

2.3 Distributional Semantic Models: Topic Model

Topic Model Methods are probabilistic model methods. These methods also use the distributional hypothesis that is the fact that words which appear in same contexts tend to have similar meanings. Most topic models produce a vector of numbers for every text - the distribution of topics and a similar vector for every word - the affinity of the word to every topic.

One of the first methods in Topic Model is the probabilistic LSI (pLSA method) proposed in (Hofmann 1999). The pLSA approach models each word in a document as a sample from a mixture model, where the mixture components are multinomial random variables that can be viewed as representations of "topics." Thus each word is generated from a single topic, and different words in a document may be generated from different topics. Each document is represented as a list of mixing proportions for these mixture components and thereby reduced to a probability distribution on a fixed set of topics. This distribution is the "reduced description" associated with the document.

Latent Dirichlet Allocation (LDA), today a main method in Topic Model representation of texts, has been developed in (D. M. Blei, Andrew, and Michael 2003) or again in (Pritchard, Stephens, and Donnelly 2000), and is very alike the pLSA method except that in LDA the topic distribution is assumed to have a sparse Dirichlet prior. The sparse Dirichlet priors encodes the intuition that documents cover only a small set of topics and that topics use only a small set of words frequently.

Let us note that in the case of topic model representation as it is explained in the previous papers of this section, any topic is not strictly defined, neither semantically nor epistemologically. The process to identify and create the topic is made by automatic detection of the likelihood of term co-occurrence. Consequently, a lexical word may occur in several topics with a different probability. Nevertheless this word will occur with a different typical set of neighboring words in each topic. Note that in this model each document is assumed to be characterized by a particular set of topics. This is somehow similar to the standard bag of words model assumption i.e two documents that tend to have similar words and number of words tend to have similar meanings, and this makes the individual words exchangeable.

Here we add the algorithm and more explanations about formal procedure to apply Latent Dirichlet Allocation to a set of texts. The following algorithm has been written from the work in (D. M. Blei, Andrew, and Michael 2003).
LDA assumes the following generative process for a corpus D consisting of n documents each of length n_i. Admit the number of topics is equal to k. Then choose for each document $i \in (1, ..., n)$ a (sparse) Dirichlet distribution of topics θ_i as a prior. For each topic then choose a (sparse) Dirichlet distribution of words ϕ_k in this topic.
Now for each word $w_{i,j}$ from the document i and at the position j:
• choose a topic $z_{i,j} \sim$ Multinomial(θ_i)
• choose a word in this topic $\sim$ Multinomial($\phi_{z_{i,j}}$)

Note how is working the LDA : the first multinomial assignment step is the answer to the question : “How prevalent are topics in the document?” while the second multinomial assignment step is the answer to the question : “How prevalent is that word across topics?”. Thus, in the process of assignment of topics : one should choose a topic for a word according to a weight of these two criterion.
Since, other methods like the Pachinko allocation presented in (Li and McCallum 2006) have been proposed to improve the quality of LDA.
Deepest explanations about LDA and other methods of topic model can be found in (D. Blei 2012).

2.4 Word Embeddings (Neural Language Model)

Very dynamic topic, Neural language methods or predictive methods using word embeddings have proven to be very efficient first in (Collobert and Weston 2008). These predictive-based methods are compared to count-based methods in (Baroni, Dinu, and Kruszewski 2014), the paper showing that predictive methods outperform count-based methods and are thus very useful to represent texts.
On the surface, Distributional Semantic Models and word embedding models use varying algorithms to learn word representations : the former counts, the latter predicts. Nevertheless the two types of models are acting on the same underlying statistics of the data, i.e. the co-occurrence counts between words.

Word2Vec is today arguably the most popular method of the word embedding models. (Le Quoc and T. 2014) recommends two architectures to learn word embeddings that are : cBoW and skip-gram. The main idea of Word2Vec is to train words to learn to predict neighbor words. While cBoW trains a window of n words around the target w_t to predict it, skip-gram trains a word to predict the context i.e w_t to predict a window of n words around itself. The use of word embeddings to produce representations of texts can also be useful to describe documents. The process Doc2vec described in (Quoc 2014) adds the document to the algorithm as a feature. The document's meaning in then represented in a space of words that adds in the algorithm the document as a feature and such tries to represent document meaning in a space of words. It is thus trained and represented as a vector.

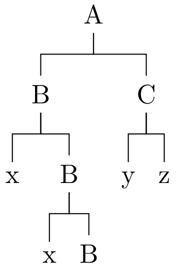
Global Vectors for Word Representation (GloVe) is a method developed and described in (Pennington, Socher, and Manning 2014). Glove is another very important method in words embeddings. It goes from the assumption that the statistics of word's occurrences in a corpus is the primary source of information available to all unsupervised methods for learning word representations. GloVe can be considered as a count-based method: it uses the co-occurrence word-word matrix and reduces it to a co-occurrence word-feature matrix. By this way it constructs word-vectors. A way to then represent documents is to take an average of these word-vectors or an weighted-average of these word-vectors.

2.5 Tree Parsing and Visualization of the Structure

In this part we describe a way to describe texts from a syntactic point of view. Since we have mainly worked from a semantic pov in the previous method, structure will be our main focus here.

The idea of describing texts with tree structures has been partly suggested by the "father of modern linguistics" Noam Chomsky. To describe languages, the linguist sets out a series of formal grammars in (Chomsky 1956).
One formal grammar used to build parse trees is the context-free grammar. In formal language theories, a context-free grammar (CFG) is the grammar that generates a context-free language (CFL). It sets out that any language that can be defined as a context free language has a structure of Symbols, Terminals and Rules between Symbols and Terminals. The words in the language are sequences of terminals only.

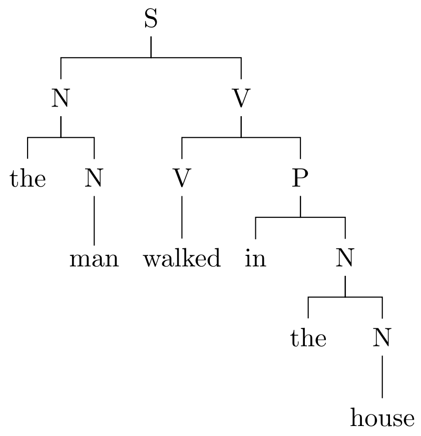
Symbols represent part of speech and Terminals are the words themselves. Hence any sentence in the language is built from context-free rules. The words in the language are sequences of terminals only.
Let us give an example. Let us say that A, B and C are symbols. 'x', 'y' and 'z' are terminals and the rules that we construct are:
A → B C B → x B → xB C → y z
Then that leads to the following tree:



Let us now use part of speech categories as symbols and words in the language itself as terminals. For example let us take a small part of the English dictionary and say we have only Nouns, Verbs, Determiners and Preposition words in the language. Let us attribute to these categories the symbols that will thus be N (Noun), V (Verb), D (determiner), P(Preposition). Now let us define the following terminals: we define in the category N (Noun) the words "man", "the", and "house" in the category V (Verb) "walked" and in the category P (Preposition) the word "in".
A very restricted possibility of the full grammar book with our previous categories and terminals associated is:

S → N V N → “the” N N→ “man” or “dog” or “house” V → V P V → “walked”
P→ P N P→ “in”

This grammar book allows us to build the following tree that will lead to the following sentence (read the bottom of the tree)



The idea behind is that the full English grammar and all the grammars in general are construction of these categories, nodes, terminals linked with these rules that lead the construction of trees that are our sentences. The idea behind tree representation is thus to represent sentences as a tree that is built from the grammar.
The major flaw with this method is the multiplicity of ways to build the tree and so the multiplicity of possible structures for a unique sentence. This multiplicity comes in fact from the multiple ways a sentence can be understood and then the creation of multiple structures.
This is explained in (Bird, Klein, and Loper 2009) with the sentence “I shot an elephant in my pajamas”. This sentence can have two different structures depending on whether I or an elephant is in my pajamas.

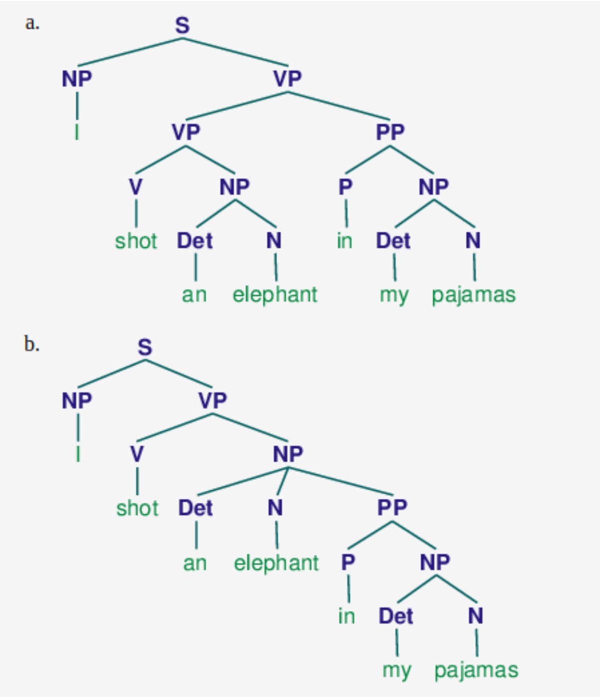


Figure 1: Two different syntactic structures built from the same sentences with tree representation (from (Bird, Klein, and Loper 2009)).

Regarding longer sentences, the number of ambiguities can lead to a large number of structures for exactly the same sentence. Furthermore the structure of the tree can depend on the parser that we use. As explained in (Bird, Klein, and Loper 2009) : a parser processes input sentences according to the production of a grammar, and builds one or more constituent structures that conform to the grammar. It searches through the space of trees licensed by a grammar to find one that has the required sentence along its fringes. Consequently the parser does not guarantee the uniqueness of the found tree.

2.6 Summary of Semantic Methods

Day	Min Temp	Summary
VSM based on the assumption that documents that have similar words and similar number of words are similar	sBoW : Sparse Bag of words	Each term is represented by a vector of 0 and 1, the text is then the sum of these vectors (after stemming, deleting of stop words etc) vector d of N (size vocabulary) components with d[i] = count of term i
	Tf-Idf : Term frequency Inverse Document Frequency	instead of counting the words we will apply a weight Tf-Idf(t,d,D) susceptible to select the most important words vector d of N (size vocabulary) components with d[i] = Tf-Idf(t_i,d,D)
VSM + Count-based model based on the counting of recurrence of a context, of a word in different situations	LSA : Latent Semantic Analysis	apply a Singular Value Decomposition (SVD) to our term-document weighted Tf-Idf matrix (or sBoW) in order to find a so-called latent semantic space that retains most of the variances in the corpus. X = U*T*V (reduction of matrix) with X the term-document matrix
	ESA : Explicit Semantic Analysis	represents each word or text as a weighted vector of Wikipedia concepts vector d of N components with d[i]=sum of concepts of Wikipedia by summation of the words
	Random Indexing	assigns to each context (i.e word or document) a unique vector d-dimensioned called an index vector composed of a small number of -1,1 and the rest of 0. vector d of N components
VSM + Topic Models based on the representation of documents by a distribution in topics	pLSA : Probabilistic Latent Semantic Analysis	models each word in a document as a sample from a mixture model, where the mixture components are multinomial random variables that can be viewed as representations of probability distribution on a fixed set of topics, themselves a distribution on words
	LDA : Latent Dirichlet Allocation	same than pLSA with a Dirichlet prior on the topic distributions
Word-Embedding - Neural method methods using word-embedding to achieve good representation	Word2Vec+Doc2Vec	trains words to learn how to predict neighbouring words: can also train a document in the process
	GloVe : Global vectors for word representation	count-based method: uses the co-occurrence word-word matrix and reduces it to a co-occurrence word-feature matrix. represents each word as a vector (already a large set of words trained)

From all these representations, our choice was to study the data from three points of view. The Vector Space Model (VSM) as reported in (Turney and Pantel 2010) has for aim to represent each document in a collection as a point in a space (a vector in a vector space) and the documents relations in Term–Document, Word–Context, and Pair–Pattern matrices. The Latent Dirichlet Allocation presented in (D. M. Blei, Andrew, and Michael 2003) is presenting documents as a list of topics or multinomial distributions of topics. These two methods work on the same aspect of the text that is the semantic. A last method that we wish to add is the tree representation method since this one, as we decide to compute it, will only be impacted by the syntax of the text and thus could possibly add precious informations.

These two first methods about semantic have non-negligible qualities. The Latent Dirichlet Allocation is shown to be efficient for example in (D. Blei 2012), the VSM methods have presented pros in (Turney and Pantel 2010) and have been presented as the most widely used method for query retrieval in (Singh n.d.). The tree representation method has not yet shown such qualities but has an undeniable interest due to its framework (the syntax).

3 State of the Art. Distances for Text Representations

3.1 Distances for Vector Space Model

Different metrics can be used to compare texts under Vector Space Model representation. The most straight-forward is the Euclidean distance between two vectors x and y which represent two texts :

$$\sqrt{\sum (x_i - y_i)^2}$$

A lot of distances are of interest to work on word similarity. As explained in (Turney and Pantel 2010) there are geometric measures of vector distance such as Euclidean distance and Manhattan distance but also distance measures from information theory including Hellinger, Bhattacharya, and Kullback-Leibler. (Bullinaria and Levy 2007) compared these five distance measures and the cosine similarity measure on four different tasks involving word similarity. Cosine similarity performed better over all others.
Note that there still are other distances interesting for the measure of dissimilarity in texts such as the Pearson Correlation Coefficient or the Averaged Kullback-Leibler Divergence developed in (Huang 2008).

We will finally take interest in two distances that have a special appeal for the exploitation of texts. We decided to focus on the two followings : Jaccard Index and Cosine similarity. The first because it provides good result when it is used such as in (Huang 2008) or in (L. Lee 1999) and the second for its relative popularity in data mining, for example used in (Pennington, Socher, and Manning 2014) or in (Gabrilovich and Markovitch 2007) but also for its results previously stated.

First introduced in (Jaccard 1901), the Jaccard index is a measure of similarity between two sample sets and is exactly the number of common attributes divided by the number of attributes which exist in at least one of the two objects. Its mathematical expression is :

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

for two sample sets A and B.
The measure of dissimilarity $1 - J(A,B)$ is actually a distance and will be the expression used in our work. To prove it can be used the Steinhaus Transform. Given a metric (X,d) and a fixed point a $\in$ X, one can define a new distance D_{bis} as

$$D_{bis}(x,y) = \frac{2D(x,y)}{D(x,a) + D(y,a) + D(x,y)}.$$

This transformation is known to produce a metric from a metric in (Späth 1981). One should then take as the base D the symmetric difference between two sets, and what one ends up with is the Jaccard distance.
If $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ are two vectors with all real $x_i, y_i \geq 0$, then their Jaccard similarity coefficient is defined as :

$$\frac{\sum\limits_{i=1}^{n} min(x_i, y_i)}{\sum\limits_{i=1}^{n} max(x_i, y_i)} \text{ with } x_i, y_i \text{ real superior or equal to 0}$$

This definition is the one useful to our work since it fits the Vector Space Model representations.

Cosine similarity is usually used in the context of text mining to compare documents or emails

In other words, in cosine similarity, the number of common attributes is divided by the total number of possible attributes. Cosine captures the idea that the length of the vectors is irrelevant; the important thing is the angle between the vectors. For $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ two vectors with all real $x_i, y_i \geq 0$, then we define the function similarity Cosine :

$$\frac{x \cdot y}{||x||_2||y||_2} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

and its dissimilarity Cosine

$$1 - \frac{x \cdot y}{||x||_2||y||_2}$$

that is actually not a distance because it can not fill the triangular inequality property. Indeed in $\mathbf{R}^2$ and for angular coordinates, we have :

$$1 - \cos(0, \frac{\pi}{2}) = 1 \ \geq 2 - \cos(0, \frac{\pi}{4}) - (\cos(\frac{\pi}{2}, \frac{\pi}{4}) = 2(1 - \frac{\sqrt{2}}{2})$$

Note that the cosine similarity is related to the Euclidean distance as follows :

$$||x - y||^2 = ||x||^2 + ||y||^2 - 2x \cdot y = 2(1 - \frac{x \cdot y}{||x||_2||y||_2}) \text{ for normalized } x,\ y$$

hence the dissimilarity Cosine is equal to $\frac{1}{2}\ ||x - y||^2$. A distance can also be derived directly from the cosine similarity that is simply the angle distance, its expression for positive vectors is :

$$\frac{2 \cdot \cos^{-1}(\text{Cosine similarity})}{\pi}$$

.

3.2 Topic Models Distances

In this part is discussed the interest of several distances for topic models. As well as the distances for Vector Space Model, there are numerous distances that we decided not to use such as the Bhattacharyya, the J-Divergence or again the Wassertein distance.

In (Aitchison 1992) four specific conditions that should be verified to obtain a good scalar measure on compositional data are proposed. These are scale invariance, permutation invariance, perturbation invariance and subcompositionnal dominance. These conditions are tested against several distances in (Fernandez, Vidal, and Pawlowsky-Glahn n.d.). Since any of our compositions will be scaled to 1, the scale invariance is not essential. Hence only three conditions must be checked. The conclusion is that within a wide range of distances, only the so-called Aitchison distance and Mahalanobis (clr) distance meet every condition and we will use the first one in what follows.
The Aitchison distance is defined from the geometrical mean :

$$g : \left| \begin{array}{ccc} \mathbf{R}^n & \longrightarrow & \mathbf{R} \\ x & \longmapsto & g(x) = (\prod_{i=1}^{n} x_i)^{1/n} \end{array} \right.$$

Then the Aitchison distance for two discrete distributions p and q is :

$$(\sum(log(\frac{p_i}{g(p)}) - log(\frac{q_i}{g(q)}))^2)^{1/2}.$$

A second distance that is of interest because of its popularity is the Jensen-Shannon Divergence. It is defined from the Kullback-Leibler Divergence or relative entropy that is for two discrete probability distributions p and q and “from q to p”

$$D_{KL}(p||q) = -\sum_{i} p(i) log \frac{q(i)}{p(i)}$$

This measure, introduced in (Kullback and Leibler 1951), is about how one probability distribution diverges from a second. A lot of metrics for probability distributions are derived from this expression such as the total variation distance related to it by the Pinsker’s inequality and the Jensen-Shannon Divergence.
The Jensen-Shannon Divergence has some notable (and useful) differences with Kullback-Leibler divergence, including that it is symmetric and it is always a finite value : the square root of the Jensen–Shannon divergence is thus a metric often referred to as Jensen-Shannon distance.
For two discrete probability distributions p and q, the Jensen Shannon Divergence is defined to be :

$$\text{JSD}(p||q) = \frac{1}{2}D_{KL}(p||m) + \frac{1}{2}D_{KL}(q||m) \text{ with } m = \frac{1}{2}(p+q)$$

The distance is notably established as a good distance for the study of texts in (L. Lee 1999).

3.3 Tree Representation Distances

To study tree representations, we decided to use a distance derived from the symmetric difference also known as the disjunctive union. The symmetric difference such as defined in (Orowski and Borwein 1991) of two sets is the set of elements which are in either of the sets and not in their intersection. Its mathematical expression is :

$$|A \cup B| - |A \cap B|$$

for two sample sets A and B.

The distance that we will apply to our trees is the summation of the symmetric difference for all partitions between two trees. The trees that we take in account are actually themselves without their last leaves i.e the words of the sentences. It means that we keep the whole structure but the words. This choice is made since we chose to compare the structures of the texts and not what they contain i.e we wish to study within this distance and for the tree representation the syntax and not the semantic.
This distance is used for the study of phylogenetic trees. We chose it in part for its practicality since it is already used for the study in phylogeny.

4 Permutation Methods for Hypothesis Testing

4.1 Permutation Framework

Statistical tests, given a test statistic, can be designed in either a parametric or a non-parametric way. In the framework of OODA for texts-valued random variables, the underlying probabilistic models of the data we are working on are very complex and thus make the parametric way time-consuming to be used or even impractical such as in (Ginestet et al. 2017). Hence, in this section, we we will define a non-parametric statistical test using in particularly the permutation theory explained for example in (Bóna and Miklós 2004).
Thus, let $(d_{11}, ..., d_{1n1}, d_{21}, ..., d_{2n_2})$ be independent text-representation random variables.
Let the random variables in the first sample $d_1 := d_{11}, ..., d_{1n_1}$ of size n_1 (respectively in the second sample $d_2 := d_{21}, ..., d_{2n_2}$ of size n_2) be identically distributed with a continuous cumulative distribution F_1 (respectively F_2).
Then the null hypothesis H_0 that we want to test is the hypothesis that the two distributions are identically distributed :

$$H_0 \ : \ F_1 = F_2 \text{ against } H_1 \ : \ F_1 \neq F_2$$

A test statistic is a statistic (a quantity derived from the sample) used in statistical hypothesis testing as explained in (Berger and Casella 2001).
In general, it is explained in (Berger and Casella 2001) that a test statistic is selected in order to quantify, within observed data, behaviors that would distinguish the null from the alternative hypothesis (if such alternative actually exists). T the statistic test is thus characterized as a test that can highlight a real difference between two samples i.e that can highlight differences between F_1 and F_2 with our previous notations. Under null hypothesis, the two samples of texts are exchangeable. Hence, it is possible to estimate the null distribution as explained in (Stuart, Ord, and Arnold 1999), i.e the probability distribution of the test statistic when the null hypothesis is true, of T by randomly permuting the group labels of our texts. For each permutation, we get a value t_{perm} of the "permuted" test statistic. The set of all t_{perm} values defines a discrete approximation of the null distribution (under assumption the null hypothesis is true) of the test statistic.
Note the number of all possible and unique permutations is equal to $\dfrac{(n_1 + n_2)!}{n_1!n_2!}$. The term permutation is actually very close of the combination term. By taking all the n_1 elements subsets of a $n_1 + n_2$ sized set and ordering each of them in all possible ways we obtain all the n_1-permutations of the set. This number then corresponds to the binomial coefficient:

$$\binom{n_1 + n_2}{n_1} = \frac{(n_1 + n_2)!}{(n_1 + n_2 - n_1)!n_1!} = \frac{(n_1 + n_2)!}{n_1!n_2!}$$

Note also that in the case the test is two-sided i.e $n_1 = n_2$ therefore the number of possible permutations is further divided by a factor of two (by symmetry of the permutations). In any event, the number of possible permutations grows very fast with the sample sizes. For example, when $n_1 = n_2 = 8$, which are our maximum number of texts from the same president here, we should already run 6435 permutations, which, in fact, makes the exhaustive computation of the permutation distribution highly time-consuming. Hence, in a case of a too big sample size, it will be necessary to sample a subset of permutations with replacement among the possible ones, assuming that each of the possible values of the test statistic after permutation are equally likely to arise. Note that if we decide to work at sentence level, the number of possible permutations will simply be too high to be reached. Indeed we will work with $n_1 = n_2 > 1000$ that leads to non-computable time.

The choice of the statistical test used has to be picked within a large frame and adapted to our data. The first that will be tested is the Test Two-Sample T-Test for Equal Means.
Its mathematical form expressed in ("e-Handbook of Statistical Methods" 2012), by supposing

equal variances :

$$H_0 : \ \mu_1 = \mu_2 \quad H_1 = \mu_1 \neq \mu_2 \quad T = \frac{d(\overline{Y}_1, \overline{Y}_2)}{\sqrt{1/n_1 + 1/n_2}}$$

with n_1 and n_2 the sizes of the two samples, d the distance we wish to use and $\overline{Y}_1$, $\overline{Y}_2$ the sample means. The whole statistic test is then based on the calculation of the means of the two samples after permutation. The straight-forward problem for this calculation is that the mean calculated in an euclidean way does not fit any distances other than the euclidean one. It is thus not coherent when it comes to compute the value of the statistic test that is the distance between the two means.

An other method was considered to compute this mean in a more coherent way within our framework. The Fréchet mean named after Maurice Fréchet and reported for instance in (Marron and Alonso 2014) is a generalization of centroids to metric spaces, giving a single representative point or central tendency for a cluster of points. Let (M, d) be a complete metric space, $(x_1, ...x_n)$ be our n representations of documents and y any point in M thus the Fréchet variance is :

$$\Phi(y) = \sum_{i=1}^{n} d^2(y, x_i)$$

We have therefore the Fréchet mean to be the point that minimises the Fréchet variance:

$$m = \underset{y \in M}{\arg\min} \sum_{i=1}^{n} d^2(y, x_i)$$

This value defines a proper mean appropriate to our distance and then coherent. Nevertheless a problem with this is the high computational cost that it implies and so makes it almost unpractical. We will thus also consider a medoïd version of this mean i.e, when the computational cost is too high, we will consider $i \in (1, ..., n)$ such that:

$$m = \underset{j \in (1,...,n)}{\arg\min} \sum_{i=1}^{n} d^2(x_j, x_i)$$

We will also consider an other statistic test that may best fit our data. The test consists in summing all the distances between each point of the two samples. Let (M, d) be a complete metric space, $(x_1, ...x_n)$ be our n representations of documents thus the statistic test is defined as:

$$T = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} d(x_i, x_j)$$

In the following development of this paper, we will call this statistic the Total distances.

4.2 P-value

As defined first in (Pearson 1900), the p-value is the probability for a given statistical model that, when the null hypothesis is true, the statistical summary would be the same as or of greater magnitude than the actual observed results.

We now aim to use this null hypothesis tail probability $p_\infty = P_{H_0}(T \geq t_{obs})$ as an indicator for deciding whether to reject H_0. Note that we are here using t_{obs} as the result of the test statistic T with no permutation done. However, we do not know the actual distribution of T under the null hypothesis. In order to estimate T, we use the permutation framework described previously. The distribution is approximated through the discrete permutation distribution using m random permutations sampled with replacement as explained for example in (Stuart, Ord, and Arnold 1999). Let B the integer-valued random variable that count the number of permutations out of m that led to values t_{perm} of the test statistic at least as extreme as the observed value t_{obs} i.e that led to $t_{perm} \geq t_{obs}$. Let then b_{obs} be the number of permutations out of m that led to $t_{perm} \geq t_{obs}$ in a specific run of the test.

There are different ways of estimating the p-value out of the mechanics of permutations. The common approach as defined before counts the number of times the value t_{perm} is equal or exceed the observed value t_{obs} out of the m sampled permutations (Pesarin and Salmaso 2010). This approach is providing an unbiased estimation of the p-value but fails to provide exact testing procedures in the usual sense of the term because it does not account for the variability introduced by sampling the permutations as explained in (Phipson and Smyth 2010). In this work, we instead rely on the definition proposed by still (Phipson and Smyth 2010), which is based on randomization tests and proposes to read $p = P(B \leq b_{obs})$ instead of $p_\infty = P_{H_0}(T \geq t_{obs})$. This definition provides an exact test – P_{H_0} (p $\leq \alpha) = \alpha$ – regardless of the sample sizes and number m of sampled permutations. Hence, the choice of m only impacts the power of the test, as expected. In practice, this p-value is computed in (Phipson and Smyth 2010):

$$p(T) = \frac{1}{m_t + 1} \sum_{b_t=0}^{m_t} F(b(T); m, \frac{b_t + 1}{m_t + 1})$$

with F the cumulative probability function of the binomial distribution.

Once we have defined our p-value, it is necessary to be able to analyze results and usually the next thresholds are taken as reference, (Nuzzo 2014):

- $p \leqslant 0{,}01$: very strong presumption against the null hypothesis
- $0{,}01 < p \leqslant 0{,}05$: strong presumption against the null hypothesis
- $0{,}05 < p \leqslant 0{,}1$: weak presumption against the null hypothesis
- $p > 0{,}1$: no presumption against the null hypothesis

5 Clustering Methods

The aim of clustering is to find structures in data and is therefore exploratory in nature. Clustering has a long and rich history in a variety of scientific fields. Indeed in clustering analysis one does not use category labels that tag objects with prior identifiers, i.e. class labels : data are by nature unknown and there are no labels to classify them in. This absence of category information distinguishes data clustering (unsupervised learning) from classification or discriminant analysis (supervised learning).

5.1 K-means

One of the most popular and simple clustering algorithms, K-means, was first published in 1955. The K-means algorithm for classification is an idea that goes back to Hugo Steinhaus that refers of it in (Steinhaus 1957). A detailed overview and discussion about the K-means algorithm can be found in (Jain 2010).

K-means clustering aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster.
For $(x_1,\ ..., x_n)$ a set of observation with each observation x_i being a d-dimensional real vector, K-means aims to classify this set in $S = (S_1,\ ..., S_k)$ in order to minimize the within-clusters sum of squares or variance, called later within-ss. K-means clustering for a set of observations aims to find:

$$\arg\min_S \sum_{i=1}^{k} \sum_{x \in S_i} d(x, \mu_i)^2$$

with μ_i the mean of the cluster S_i and d our chosen distance.

The algorithm we will use is very straight-forward except that the centroids will not be the means of the observations belonging to a cluster. We will choose the centroid of S_i to be one of the observations of the cluster S_i i.e the centroid will be computed in a medoïd way. Outside of this particularity the K-means algorithm chosen is the classical way. It is based on two steps : first the assignment step (assignment of the observations into centroids) and then the update step (computation of the new centroids).

To identify the "most correct" number of clusters for our data, we will use the elbow method. This method looks at the percentage of variance explained as a function of the number of clusters. The main idea is to set k to the number of clusters m so that adding another cluster doesn't give much better modeling of the data i.e such as the variance explained associated to the number of clusters m+1 is not much more than the previous one.

Examples of application of the K-means algorithm are numerous. Let us note that cluster analysis is a common tool for market segmentation such as in (Kuo, Ho, and Hu 2002). With the help of a K-means clustering, it is possible to use the implemented clusters to determine which factors group members relate (for customers, these would be their buying preferences). It is also used in various other topics, including geostatistics in (Honarkhah 2010) and agriculture such as in (Burrougha, Van Gaansa, and MacMillanb 2000) or (Al Blesh, Braik, and Bani-Ahmad 2011).

5.2 Hierarchical Clustering

Hierarchical clustering is a method of clustering producing a set of nested clusters organized as a hierarchical tree. Two types of hierarchical clustering exist: the agglomerative that takes in input each point as an individual cluster and outputs one single cluster; and the divisive part builds the hierarchy from the individual elements by progressively merging them into clusters as described in (Lior and Maimon 2005).

To achieve so, the different steps are to determine which clusters to merge in one single cluster. Usually, we want to take the two closest clusters, according to the chosen distance. There are different ways to compute the distance between two clusters (each cluster being composed of one or more elements).
The Single linkage method that consists to define the distance between two clusters as the minimum distance between any single data point in the first cluster and any single data point in the second cluster while the Complete linkage consists in defining the distance between two clusters to be the maximum distance between any single data point in the first cluster and any single data point in the second cluster.
The Average linkage that consists to define the distance between two clusters to be the average distance between data points in the first cluster and data points in the second cluster.
The Centroid method where we use the distance between the means of the two clusters. We will not use this method since the mean would be defined in an euclidean way and then would not fit the distances we are using.
The Ward's method: This method does not directly define a measure of distance between two points or clusters and is based on an ANOVA approach. It is best described in (Ward 1963).

Hierarchical clustering does not assume a specific number of clusters since the output of the algorithm is a tree that can be cut at any point (for, so, any number of clusters desired). Since it is a clustering method, hierarchical clustering shares mainly application topics with K-means clustering.

6 Proposed Methodology for Testing our Data

Let us recall that the project of this thesis is to propose a methodology that aims to develop statistical tools to perform null hypothesis testing and clustering or classification on texts. The goal is to be able to provide a procedure to test data, the algorithms that go with it and a methodology. We describe here the methodology we will use for the testing of our data and we will also describe how we are able to provide the best representation and distance associated related to our data.

We first draw a tree to recapitulate every method we want to apply on our tests and to test afterwards.

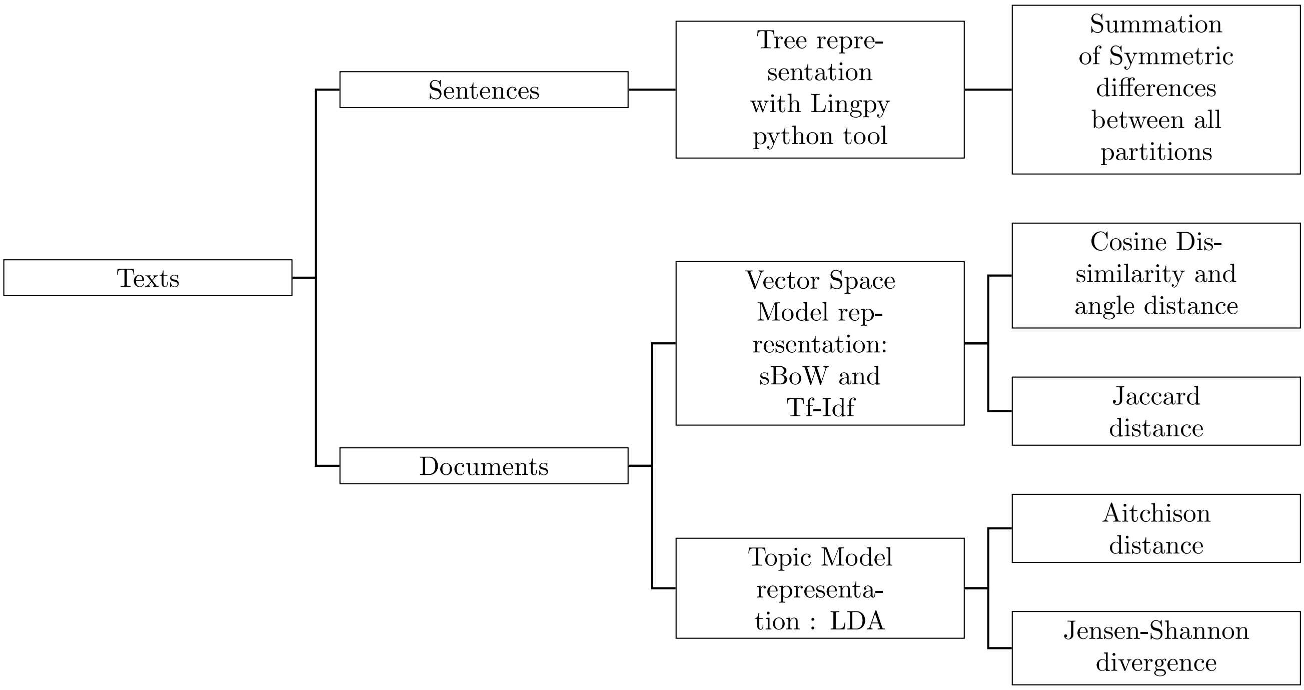

Texts
Sentences
Documents
Tree representation with Lingpy python tool
Summation of Symmetric differences between all partitions
Vector Space Model representation: sBoW and Tf-Idf
Cosine Dissimilarity and angle distance
Jaccard distance
Topic Model representation : LDA
Aitchison distance
Jensen-Shannon divergence

Table 1: Resume of the full methodology, representation methods and distances for representations.

Each data at the beginning is a full text, speech, document, tweet. The first step is to chose whether we will work at document or sentence level.
The next step is to make the data susceptible to be exploited. This requires cleaning and tokenization of the samples. The tokenization is the transformation of each document-sentence to a bag of words. For example the first sentence of the first speech of Bill Clinton during his first term will be turned from:
Mr. President, Mr. Speaker, Members of the House and the Senate, distinguished Americans here as visitors in this Chamber, as am I.
to:
mr presid mr speaker member hous senat distinguish american visitor chamber
This step requires previously cleaning of the data. The cleaning is the erasure of all non-useful words in the clustering of texts such as “of”, “the” but also “I” and “am”. It implies as well the replacement of other words by a common root they would share with other words such as “distinguished” that becomes “distinguish” and would share this root with “distinguish”, “distinguishing” or again “distinguishable”. Note the cleaning is only performed at document level since at sentence level we wish to keep the structure of the phrase.

Once the data are usable, the representation should be chosen.
If we chose to work at document level, our choice was to present two already popular types of

representations. In the table presented in the State of the Art section, we see that data such as documents can be represented in either Vector Space Model, (distributional semantics including extended count-based model) or Topic Model or finally Predictive Model (Tree being a special case). Since predictive methods are basically extensions of previous methods, we decided to present more basic but very popular and used methods.

Consequently we decided to present data firstly using the Vector Space Model and the so-called sparse Bag of Words. In some cases we will apply a Tf-Idf rule to this bag of word in order to get more accuracy on our clusterings. The second very popular representation is the topic model and thus we will use the Latent Dirichlet Allocation. Each data will consequently be a compositional data, a distribution among topics (note that topic model is useful only for modeling of data at document level: indeed, it requires a large number of words to provide a sense-full result).

If we decide to work at sentence level, we will consider a tree representation model that is derived as well from the Bag of Words. As said previously we will take this time only interest in the structure of the texts and not in the words. Note that we will use as a parser the one used in (Moran 2013) i.e the module python Lingpy. This is not completely adapted to our work since it had been firstly done for historical linguistic. Hence one that would have a particular interest in parsing trees, would prefer other ways for parsing trees. One could rather prefer to draw on (Bird, Klein, and Loper 2009) for extended research. But this module by having already implemented distances for trees is interesting for the introduction we want to do.

Once the representation is chosen, a distance should be used and applied to compare new formed representation-texts between them. As said before, we have chosen to apply two distances for the Vector Space Model representations, respectively the Jaccard distance and the Cosine Dissimilarity or its derived form the Angle Distance. We have also chosen to apply two distances for the topic model i.e the Aitchison distance and the Jensen Shannon Divergence. One distance finally will be used for the study of the tree representation pertinence: the symmetric difference.

Note now that once our representations and their distances are available, the main work is about defining the different methods for testing data that we have exposed. We will apply K-means clustering, hierarchical clustering and permutation tests on our data.

For the case of permutation tests, we must apply a statistic test to these representations. We will use the two described above, respectively the T-test with a non-coherent euclidean mean or with the Fréchet mean and the total distances statistic test.

In this study we will provide comparisons at several levels. First at representations level because we provide three of them including two based on semantic (Bag of Words and LDA that has the advantage to take the meaning of a text and express it in topics) and one based on the structure of the text only (tree representation).

A second level of comparison will be proposed between the different distances for the case of Bag of Words and LDA representations : at first the Cosine Dissimilarity against the Jaccard distance, and then the Aitchison distance against the Jensen-Shannon divergence respectively. We will also take interest of what might be a statistic test suitable for our data and how to apply it.

7 Case Study

In the case study, we will present five cases:
· in the Vector Space Model framework : the study of the Tf-Idf model with the Cosine Dissimilarity and of the Sparse Bag of Words model with the Jaccard distance.
· in the topic model framework : the study of the Latent Dirichlet Allocation with the Aitchison distance and of the Latent Dirichlet Allocation with the Jensen Shannon Divergence.
· in the tree representation framework, we will use the tool Lingpy with the symmetric difference distance defined above.

A lesser goal of this study is to provide interesting analyses on presidents of the United States of this 24 four last years (without including the current term). We will then try to provide useful comparisons at this level as well. It is important to keep in mind that we will work at document level during almost the whole case study. The sentence level will be used only for tree representation.
Before starting the study, we note here some of our expectations. We are studying three presidents including two democrats (Clinton and Obama) and one republican (Bush). That leads us to think classification or p-permutation tests would allow each president to recognize itself, and allow Clinton and Obama to be on the same page regarding Bush. We also expect an especially high value of similarity into each term of each president (for example between two consecutive years of the first term of a president) in comparison with the similarity between the two terms of a president since it happens very often that a politician renews for his reelection. An other expected behavior of our data is the chronology. Indeed we expect documents that are closer in time to treat about close topics and so to be closer in general. Then it may be that speeches of the last years of Clinton have more similarities with the Bush's first speeches than with the Obama's speeches, simply because of their proximity in time. A last point is about big events and radical change of policy such as the attacks on the twin towers in New York in 2001 that we expect to change radically the policy and so the nature of certain speeches.

Let us make a note about the permutation tests : the four tables described in the case of cosine dissimilarity are the results of the permutation tests applied with : the two first the T-statistic test with an euclidean mean and the two last with the total distances statistic test. For the rest of the methods we will only use Total distances statistic test. Furthermore each test will be applied for different time separations. In the first table each variable tested is a complete term of a president. (for example, we can test Clinton first term with Clinton second term).
In the second table, each variable tested is either the outside (the two first and the two last years of each president) either the inside (the four year consecutive in the center of the complete period of the president). For example in the second table we can compare the two first years and two last years of Clinton (i.e 93-94 and 99-00) with the four years of the center of his period (i.e 95-98).

7.1 Tf-Idf study with Cosine Dissimilarity and Derivatives

In this subsection we are working on the Cosine Dissimilarity for the clustering part and its derivative the angle distance for permutation tests. A distance was indeed required to practice coherent statistic permutation tests. Nevertheless it was not necessarily sought for the clustering part.

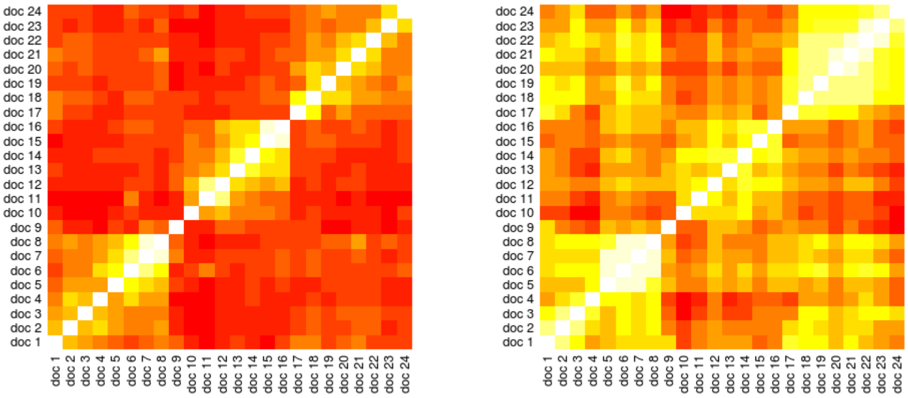


Figure 2: Matrices of Cosine similarities for Tf-Idf (on the left) and sBoW (on the right) representations at document level (24 speeches).

Here is computed, in order to get a first idea of the repartition about the texts according to their similarities towards each other, the Cosine similarities. We used different Vector Space Models to represent the data. The left graphic, based on Tf-Idf representation, is far more discriminative than the right one i.e the value

$$\frac{\text{similarities between different speech of the same president}}{\text{similarities between two different presidents}}$$

is more important for the left graphic than for the right one. In a more general way we can see that Tf-Idf tends to increase differences between graphs and particularly to increase differences when they are high. Indeed it tends to be more discriminative on speeches between different presidents but also sometimes between speeches of the same president such as the ones of Obama or the speeches between the beginning of the terms of Bush and its end. Still we consider that for Cosine similarity, Tf-Idf is more well-suited comparing to the simple Bag of Words because it decreases importance of a lot of non-useful words. Let us recall that in cosine similarity, the number of common attributes is divided by the total number of possible attributes. In the comparison of two texts it seems that Tf-Idf, from this graphic, permits to reduce the numerator in case of documents are not alike : because of the term of inverse document frequency, only the real common attributes will be really discriminative.

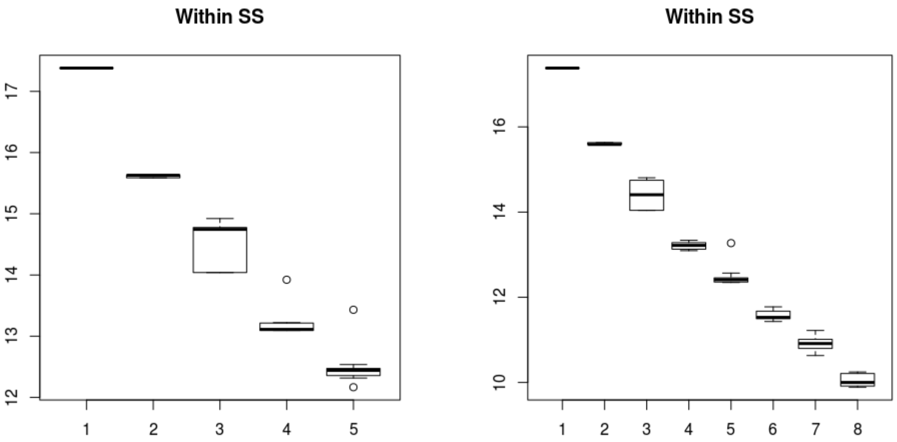


Figure 3: Boxplots of the within-ss for each k of K-means clustering - Tf-Idf-Cosine similarity at document level.

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
K-means for 2 clusters	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	1	2	1	1	1	1	1	1
K-means for 3 clusters	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	3

Figure 4: K-means clustering for two and three clusters at document level for Tf-Idf-Cosine similarity.

The variances or withinss are of (10.126374, 5.461446) for two clusters and of (4.664077, 3.877149, 5.498448) for three clusters. We decide to put two boxplots for different maximum number of k in k-clustering, having judge that the first graphic with k=5 did not give us enough information. It can be seen on the second one that the "break" in the explained variance can be situated whether in k=2 or in k=3 according to the point of view of the reader. We will then test the two possibilities.

For the 2-clusters case, the result is very conform to our expectations since it is discriminative regarding the political party. Then Clinton and Obama are classified together and Bush is put apart. We can note nevertheless a very strange affectation of the second year speech of Obama closer to the Bush speeches.

For the 3-clusters case the speeches have been well divided between each president except for the first speech of Clinton that is classified with the ones of Obama. That is in part a confirmation of the matrices of Cosine similarity previously shown : the first speech of Clinton is very few similar (with Tf-Idf) to his other speeches. Nevertheless it is not more similar to the speeches of Obama so this classification is also quite strange.

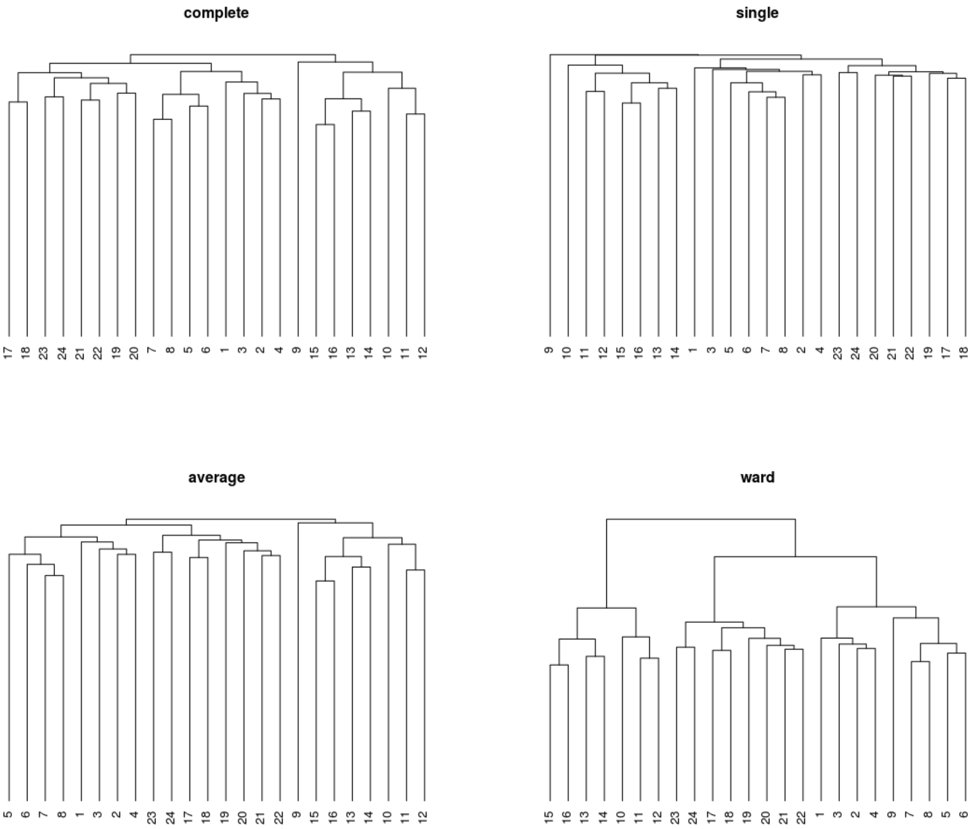


Figure 5: Trees at document level - Hierarchical clustering (four methods) for Tf-Idf-Cosine similarity.

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
complete	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1
single	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1
average	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1
ward	1	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2

Figure 6: Clustering for three clusters at document level - Hierarchical clustering for Tf-Idf-Cosine similarity.

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
complete	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4
single	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4
average	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4
ward	1	1	1	1	2	2	2	2	2	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4

Figure 7: Clustering for four clusters at document level - Hierarchical clustering for Tf-Idf-Cosine similarity.

We provide two tables of results and the different trees accorded to the method of hierarchical clustering used. The tables are just extracted from the trees and are repetitive in their information. They however classify in the number of clusters that we desire and the first table

is about a number of clusters equal to 3. We can see that for the tree first methods (complete, single, average) the classification done is very surprising since it estimates the first speech of Bush (in 2000) to be so different from every other speech that it constitutes a category apart. This speech has actually the property to be the only republican-Bush speech before the 2001 events and thus seems to be unique. This leads to set the rest of the Bush's speeches as a single cluster (the post-2001) and the democrat speeches (Clinton, Obama) as a last cluster. This is important because it estimates the first Bush's speech to be more different of the rest of his speeches than the Clinton's speeches are from the ones of Obama. It means that a major event can be more discriminative than the belonging to a political party.
If we use four clusters, the results are still showing the particularity of the ninth speech (the first of the Bush period) but the speeches of Obama logically form one single cluster. Note that with the Ward's method the speeches of the second term of Clinton are classified with the first speech of Bush rather than with the speeches of the first term of Clinton. This surprising result express two different things : firstly there is a certain temporal continuity, even between a democrat and a republican such as Clinton and Bush. And the semantic of a president can change before and after his second term. Note also that consecutive speeches are particularly close : the temporal continuity is certainly present.

In our aim to provide a coherent T-statistic test with so a coherent mean adapted to our distance, we want to compute the Fréchet mean of our data. The Fréchet mean being not defined for our distances, we decided to use the medoïd of it. At document level, that yields an immediate problem because it would mean that for n documents, one of the documents is representative of the n documents. At n=4 such as we will work in the following, it is clearly too ambitious to pretend that four documents can be well summarized in one.
We decide then to apply our Fréchet medoïd at sentence level. That makes the Fréchet mean to be one sentence from our sample of sentences that will be the representative of all the rest. We were expecting the problem that several texts (typically for $n = 4$) could be hardly summarized in one sentence and the following tables presents some tests of it:

Docs	american	back	countri	give	loan	make	pay	program	servic	time
123	2	3	4	3	2	3	2	2	2	2

Table 2: Fréchet mean for Cosine dissimilarity and full Clinton sentences.

Docs	abl	countri	end	help	must	nation	new	proud	plan	secur	will
4415	1	1	1	1	1	1	1	1	1	1	1

Table 3: Fréchet mean for Cosine dissimilarity and full Bush sentences.

Docs	comprehens	congress	deliv	later	mani	need	one	peopl	plan	will
85	1	1	1	1	1	1	1	1	1	4

Table 4: Fréchet mean for Cosine dissimilarity and first 100 - 300 - 600 - 1000 Clinton sentences.

Docs	can	everi	make	must	now	one	peopl	respons	sure	work
1576	1	1	1	1	2	1	1	1	1	2

Table 5: Fréchet mean for Cosine dissimilarity and first 2000 Clinton sentences.

These tables are an attempt of explanation for the bad results that we can find in the appendix. The Fréchet mean is not able to provide good representation of a sample of sentences. The means are bags of less than 15 words in some cases that is too low to represent the complexity of thousands of sentences. We can also see that the mean will not change by taking it for the first 100 - 300 - 600 - 1000 Clinton sentences. This yields that no difference can be made between the 100 first sentences of Clinton and the 1000 first sentences of Clinton if we apply a distance between the two (same) means and that will inevitably give poor results. The Fréchet mean will not be used in the future and we will provide results with not-coherent euclidean and geometrical mean instead.

The four tables described are the results of the different permutation tests. First we can compare the results between the two means : we can only find meaningless differences between the two performances. Results are practically the same, there is nothing to pretend that one is performing better.

If we look at the first table we see that radically all of the p-value for all the tests tend to indicate a presumption against the null hypothesis i.e a presumption against the fact that both of the speeches come from the same distribution. Such presumption tend to be whether strong or "absolute" but never weak i.e there are no reason to think after seeing such results that any four speeches term of any president is distributively alike any other four speeches term. This result is quite surprising but shows a real independence between each term of each president.

The results of the second table are more reassuring in the sens that it brings results we are expecting. Indeed we can notice that it has no presumption against the null hypothesis in the case where we test Clinton outside and Clinton inside, and the same for Bush. For the case of Obama, it seems that there is a weak presumption against the null hypothesis ($0.05 < 0.0568 < 0.1$) i.e the equality in distribution for the case of the statistic total distances and a stronger presumption in the case of the euclidean mean (0.0331). If we take for hypothesis that the speeches from Obama come from the same distribution then we have a argument against the quality of the permutation tests realized by T-statistic test with non-coherent euclidean mean. It can finally be noted that the test seems to be consistent (in most cases, the p-value is lesser than 0.05).

	Clinton first term	Clinton second term	Bush first term	Bush second term	Obama first term	Obama second term
Clinton first term		0.0084	0.0227	0.0222	0.0047	0.0244
Clinton second term	0.0084		0.0028	0.0035	0.0061	0.0142
Bush first term	0.0227	0.0028		0.0066	0.0072	0.0142
Bush second term	0.0222	0.0035	0.0066		0.0083	0.0156
Obama first term	0.0047	0.0061	0.0072	0.0083		0.0083
Obama second term	0.0244	0.0142	0.0142	0.0156	0.0083	

Table 6: Permutation tests (terms of each president) using Tf-Idf-Cosine similarity and T-statistic test with a euclidean mean at document level.

	Clinton 93-94—99-00	Clinton 95-98	Bush 01-02—07-08	Bush 03-06	Obama 09-10—15-16	Obama 11-14
Clinton 93-94—99-00		**0.2247**	0.0051	0.0255	0.0200	0.0149
Clinton 95-98	**0.2247**		0.0135	0.0135	0.0025	0.0156
Bush 01-02—07-08	0.0051	0.0135		**0.1370**	0.0162	0.0188
Bush 03-06	0.0255	0.0135	**0.1370**		0.0236	5.2326e-05
Obama 09-10—15-16	0.0200	0.0025	0.0162	0.0236		0.0331
Obama 11-14	0.0149	0.0156	0.0188	5.2326e-05	0.0331	

Table 7: Permutation tests (inside and outside terms) using Tf-Idf-Cosine similarity and T-statistic test with a euclidean mean at document level.

	Clinton first term	Clinton second term	Bush first term	Bush second term	Obama first term	Obama second term
Clinton first term		0.0.0039	0.0285	0.0051	0.0244	0.0115
Clinton second term	0.0.0039		0.0267	0.01628	0.0217	0.0056
Bush first term	0.0285	0.0267		0.0227	0.0018	0.0232
Bush second term	0.0051	0.01628	0.0227		0.0025	0.0149
Obama first term	0.0244	0.0217	0.0018	0.0025		0.0025
Obama second term	0.0115	0.0056	0.0232	0.0149	0.0025	

Table 8: Permutation tests (terms of each president) using Tf-Idf-Cosine similarity and statistic test total distances at document level.

	Clinton 93-94—99-00	Clinton 95-98	Bush 01-02—07-08	Bush 03-06	Obama 09-10—15-16	Obama 11-14
Clinton 93-94—99-00		**0.2717**	0.0004	0.0043	0.0002	0.0061
Clinton 95-98	**0.2717**		0.0002	0.0018	0.0188	0.0182
Bush 01-02—07-08	0.0004	0.0002		**0.1697**	0.0267	0.0149
Bush 03-06	0.0043	0.0018	**0.1697**		0.0035	0.0095
Obama 09-10—15-16	0.0002	0.0188	0.0267	0.0035		0.0568
Obama 11-14	0.0061	0.0182	0.0149	0.0095	0.0568	

Table 9: Permutation tests (inside and outside terms) using Tf-Idf-Cosine similarity and statistic test total distances at document level.

7.2 Sparse Bag of Words study with Jaccard Distance

We will now study our data in the Vector Space Model with the Jaccard distance. This distance is counting the number of common attributes divided by the number of attributes that exist in at least one of the two objects. Note that the Jaccard index only takes into account the words included in one of the two sets of texts. Thus it reduces a lot the number of words to take into account in the calculation and Tf-Idf loses its usefulness. In order to avoid complexity we will then simply use the sBoW model on our data with the Jaccard Index. The result of our

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
K-means for 3 clusters	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3

Figure 8: K-means clustering for three clusters at document level for sBoW-Jaccard distance.

K-means clustering are coherent with our immediate thoughts : for three clusters we will get one for each of the president. This is quite similar to the result we get with the cosine similarity but the first speech of Clinton is considered to be closer of his other speeches than from the speeches of Obama. That is more coherent with our expectations.

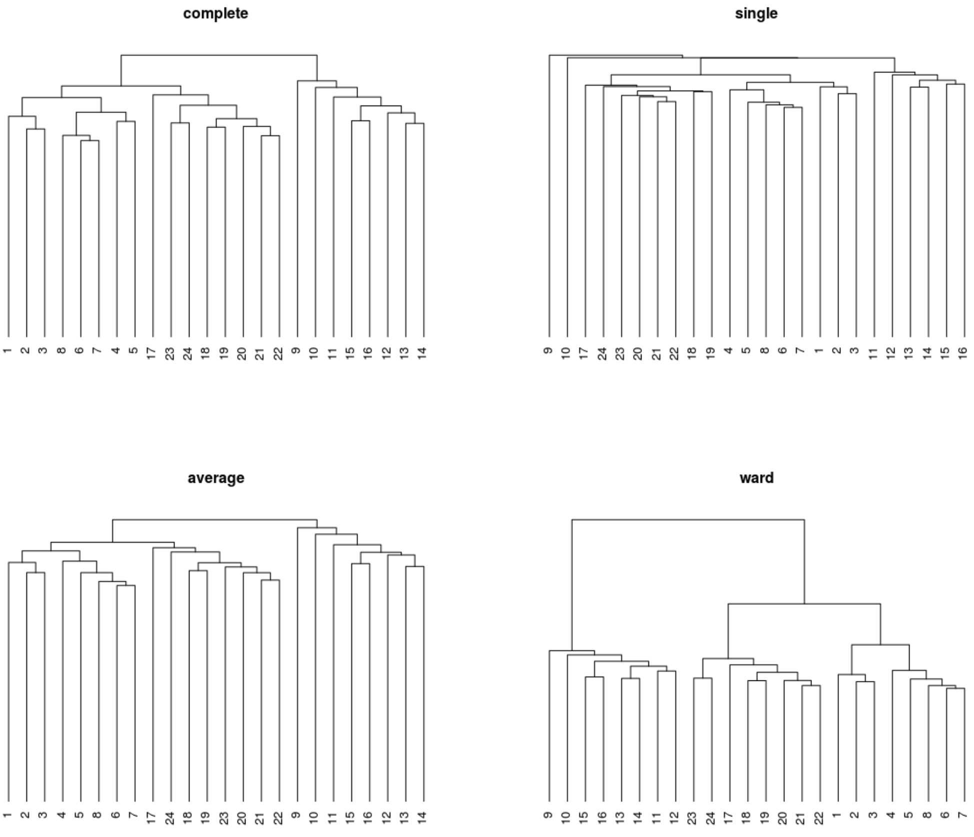


Figure 9: Trees - Hierarchical clustering for sBoW-Jaccard distance.

About hierarchical clustering, the results are very chaotic but also very interesting. Each method does not provide the same classification. The complete method will provide a similar result to the Cosine clustering since it gives an exclusive cluster to the first speech of Bush. The single method will classify any speech that is not the first speech of Bush (just before the attacks) and the second speech of Bush (just after the attacks) as belonging to the same cluster.

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
complete	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1
single	1	1	1	1	1	1	1	1	2	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1
average	1	1	1	1	1	1	1	1	2	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1
ward	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3

Figure 10: Clustering for three clusters at document level - Hierarchical clustering for sBoW-Jaccard distance.

It gives then a very exclusive place to these two speeches that belong to no cluster as we can see on the graphic. The average method is similar to the complete one while the ward method classify our data as the most intuitive : one cluster for each president.

Permutation tests with Jaccard distance are consistent with the previous tests using angle distance, except for one. When we test Obama outside (09-10 and 15-16) against Obama inside (11-14) the T-statistic test gives a p-value of 0.1474 and with the statistic test total distances a p-value of 0.1273. This means that there is no presumption against equality in distribution of these two samples and this is more coherent with our intuition. We can suppose that Jaccard index performs better on this case since it fits to our expectations.

	Clinton first term	Clinton second term	Bush first term	Bush second term	Obama first term	Obama second term
Clinton first term		0.0082	0.0166	0.0222	0.0048	0.0236
Clinton second term	0.0082		0.0048	0.0166	0.0312	0.0102
Bush first term	0.0166	0.0048		0.0186	0.0145	0.0236
Bush second term	0.0222	0.0166	0.0186		0.0035	0.0102
Obama first term	0.0048	0.0312	0.0145	0.0035		0.0205
Obama second term	0.0236	0.0102	0.0236	0.0102	0.0205	

Table 10: Permutation tests (terms of each president) using sBoW-Jaccard distance and statistic test total distances at document level.

	Clinton 93-94—99-00	Clinton 95-98	Bush 01-02—07-08	Bush 03-06	Obama 09-10—15-16	Obama 11-14
Clinton 93-94—99-00		**0.2015**	0.0048	0.0064	0.0248	0.0222
Clinton 95-98	**0.2015**		0.0081	0.0281	0.0277	0.0222
Bush 01-02—07-08	0.0048	0.0081		**0.2445**	0.0166	0.0205
Bush 03-06	0.0064	0.0281	**0.2445**		0.0300	0.0258
Obama 09-10—15-16	0.0248	0.0277	0.0166	0.0300		**0.1273**
Obama 11-14	0.0222	0.0222	0.0205	0.0258	**0.1273**	

Table 11: Permutation tests (inside and outside terms) using sBoW-Jaccard distance and statistic test total distances at document level.

7.3 LDA study with Aitchison Distance

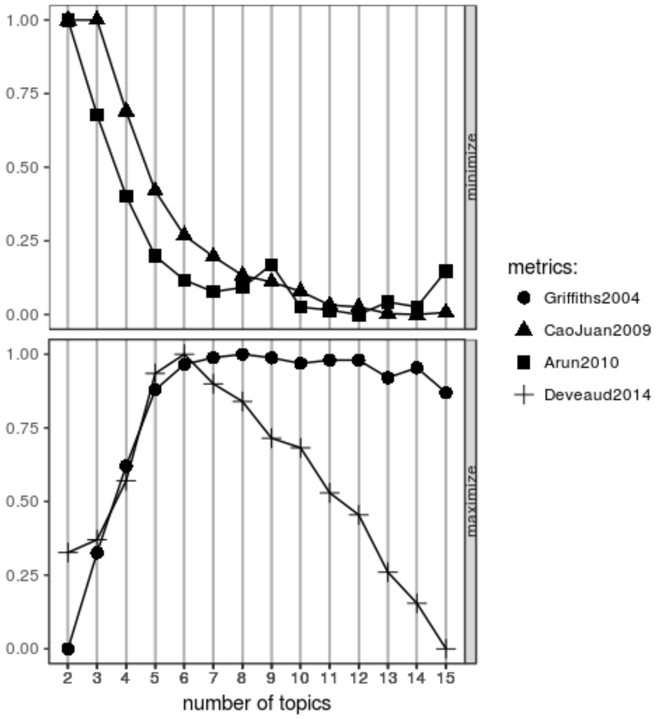


Figure 11: Decision graphs to choose the number of topics for applying LDA at document level

In order to apply our Latent Dirichlet Allocation to our data, we need to find the proper number of topic models that will suit to them. We use different metrics to estimate the most preferable number of topics for LDA model. There is different approaches by cross-validation that has been proposed to determine the good number of topics, different metrics are suggested based for example on assessing maximizing likelihood, minimizing Kullback-Leibler divergence or similar. String or vector of possible metrics are developed in (Arun, Suresh, and Madhavan 2010) or (Juan, Tian, and L. 2009) that both follow the same idea of computing similarities between pairs of topics of the model, while varying the number of topics. The presumed optimal amount of topics is reached when the overall dissimilarity between the topics achieve its maximum value. In (Deveaud, SanJuan, and Bellot 2014) is proposed a simple heuristic which estimates the number of latent concepts of a user query by maximizing the Kullback-Leibler Divergence between all pairs of LDA's topics. The last method is in (Griffiths and Steyvers 2004) where the strategy for discovering topics is in particularly based on using Gibbs sampling.
From the graphic, the minimization on the top graphic and the maximization on the bottom graphic we can assume the number of topics to be superior to six. As we decided to satisfy approximately all the criterion, the number of topics will be fitted to seven.

.

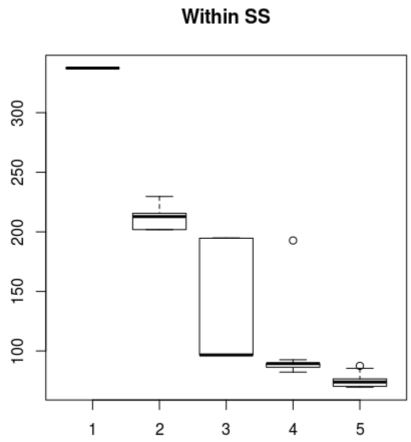


Figure 12: Boxplots of the within-ss for each k of K-means clustering - LDA -Aitchison distance

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
K-means for 3 clusters	3	2	2	2	2	2	2	2	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	3

Figure 13: K-means clustering at document level for three clusters for LDA -Aitchison distance.

The boxplots of the K-means clustering here show that the good number of clusters, i.e the one that explain the most of the variance, is three. We can somehow have a doubt since the third boxplot is not centered. The 3-means clustering looks alike the one made for the cosine similarity. The first speech of Clinton is again attributed to the speeches of Obama instead of the other speeches of Clinton. The first speech of a president seems to not necessarily be representative of his presidency.

The results of the hierarchical clustering are very coherent between them compared to the ones of the Vector Space Model. Indeed the separation in three clusters is very clear. The only difference can be found between the complete and single methods, and between the average and ward methods. The two first present the first speech of Bush as corresponding to the rest of his presidency while the two last methods consider this speech to be closer of the speeches of Clinton in terms of topics. The two results are very coherent with the rest of our analysis until now.

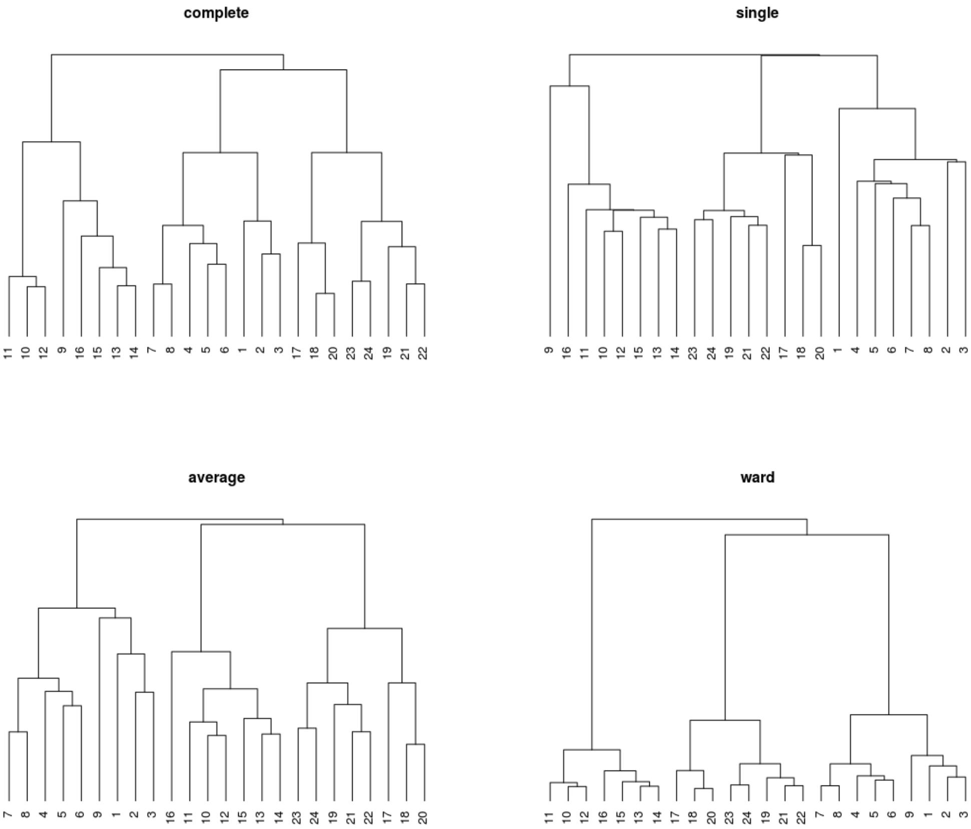


Figure 14: Trees - Hierarchical clustering for LDA -Aitchison distance.

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
complete	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2
single	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2
average	1	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2
ward	1	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2

Figure 15: Clustering for three clusters at document level - Hierarchical clustering for LDA - Aitchison distance.

Concerning permutation tests, the results are very precise and are similar to those with the Jaccard Index and sBoW. Both the statistic total distance test and the T-statistic test with euclidean mean present a strong presumption against any other hypothesis except the outside president against inside presidents. Indeed we will have for example between Bush outside (01-02 and 07-08) against Bush inside (03-06) a p-value of 0.3590 for the T-statistic test and 0.1927 for the total distances statistic test. This corresponds to what we were expecting. Furthermore the results tend to be more discriminative. With Vector Space Models, p-values tend to be bigger and thus the presumption tend to be “only” strong. With this topic model associated to this Aitchison distance, p-values tend to be less than 0.01 i.e to present a “very strong” presumption against null hypothesis in most of the cases.

	Clinton first term	Clinton second term	Bush first term	Bush second term	Obama first term	Obama second term
Clinton first term		0.0260	0.0263	0.0232	0.0280	0.0025
Clinton second term	0.0260		0.0251	0.0169	0.0031	0.0227
Bush first term	0.0263	0.0251		**0.1090**	0.0227	0.0016
Bush second term	0.0232	0.0169	**0.1090**		0.0212	0.0095
Obama first term	0.0280	0.0031	0.0227	0.0212		0.0206
Obama second term	0.0025	0.0227	0.0016	0.0095	0.0206	

Table 12: Permutation tests (terms of each president) using LDA - Aitchison distance and statistic test total distances at document level.

	Clinton 93-94—99-00	Clinton 95-98	Bush 01-02—07-08	Bush 03-06	Obama 09-10—15-16	Obama 11-14
Clinton 93-94—99-00		**0.2912**	0.0014	0.0009	0.0258	0.0271
Clinton 95-98	**0.2912**		0.0227	0.0232	0.0176	0.0149
Bush 01-02—07-08	0.0014	0.0227		**0.1927**	0.0212	0.0149
Bush 03-06	0.0009	0.0232	**0.1927**		0.0255	0.0047
Obama 09-10—15-16	0.0258	0.0176	0.0212	0.0255		**0.7108**
Obama 11-14	0.0271	0.0149	0.0149	0.0047	**0.7108**	

Table 13: Permutation tests (inside and outside terms) using LDA - Aitchison distance and statistic test total distances at document level.

7.4 LDA study with Jensen-Shannon Divergence

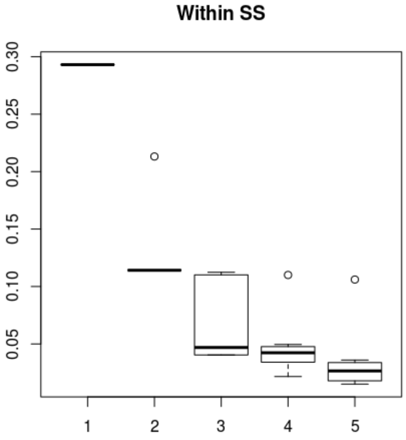


Figure 16: Boxplots of the within-ss for each k of K-means clustering for LDA-JSD

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
K-means for 3 clusters	2	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2

Figure 17: K-means clustering at document level for three clusters for LDA-JSD.

The results for the LDA representation (topic model) with this time the Jensen Shannon Divergence as a distance are very similar to those obtained with the Aitchison distance. On the boxplots for the K-means clustering we can see very clear boxplots which clearly show that most of the variance is explained by two clusters and almost everything is explained by three. The classification is the same than the one found with the Aitchison distance. The hierarchical clustering and the permutation test results are also very alike the one performed with the Aitchison distance. In this case the distance seems to not be very discriminative.

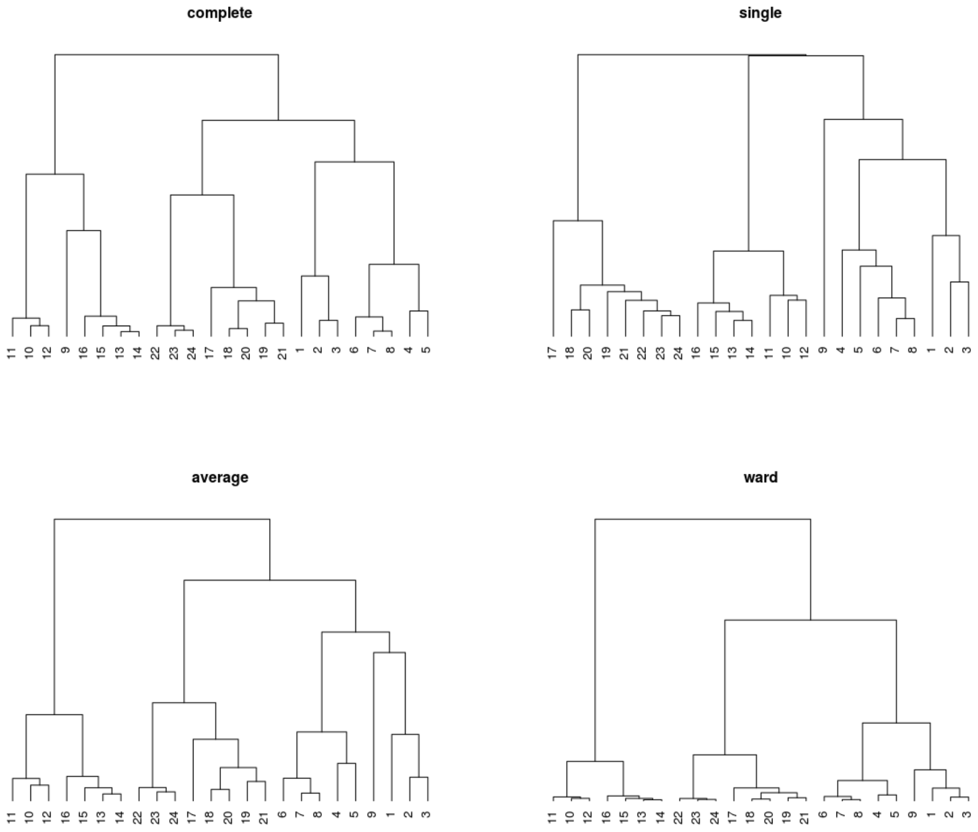


Figure 18: Trees - Hierarchical clustering for LDA-JSD at document level.

	Clinton								Bush								Obama							
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
complete	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2
single	1	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2
average	1	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2
ward	1	1	1	1	1	1	1	1	1	3	3	3	3	3	3	3	2	2	2	2	2	2	2	2

Figure 19: Clustering for three clusters at document level - Hierarchical clustering for LDA-JSD.

	Clinton first term	Clinton second term	Bush first term	Bush second term	Obama first term	Obama second term
Clinton first term		0.0176	0.0066	0.0212	0.0156	0.0206
Clinton second term	0.0176		0.0251	0.0227	0.0176	0.0072
Bush first term	0.0066	0.0251		0.0273	0.0283	0.0200
Bush second term	0.0212	0.0227	0.0273		0.0169	0.0227
Obama first term	0.0156	0.0176	0.0283	0.0169		0.0200
Obama second term	0.0206	0.0072	0.0200	0.0227	0.0200	

Table 14: Permutation tests (terms of each president) using LDA-JSD and statistic test total distances at document level.

	Clinton 93-94—99-00	Clinton 95-98	Bush 01-02—07-08	Bush 03-06	Obama 09-10—15-16	Obama 11-14
Clinton 93-94—99-00		**0.1653**	0.0061	0.0142	0.0248	0.0248
Clinton 95-98	**0.1653**		0.0047	0.0051	0.0156	0.0066
Bush 01-02—07-08	0.0061	0.0047		**0.5792**	0.0010	0.0280
Bush 03-06	0.0142	0.0051	**0.5792**		0.01561	0.0162
Obama 09-10—15-16	0.0248	0.0156	0.0010	0.01561		**0.8481**
Obama 11-14	0.0248	0.0066	0.0280	0.0162	**0.8481**	

Table 15: Permutation tests (inside and outside terms) using LDA-JSD and statistic test total distances at document level.

7.5 Tree Representation and Symmetric Difference

First note that the tests on trees are performed at sentence level since each tree is derived from a sentence. The tests will be done between very large samples of sentences : we will compare groups of 1500-sentences between each other. As presented above, each sentence forms a tree which represents its structure.
Since the high number of trees that we will compare, the computation time will be also very high and the permutation tests performed will be less accurate than the rest.
Here is presented an example to see what exactly our parser is doing and why the results presented are more of an introduction to the possibilities of tree parsing than a conclusion to it.
In our example we will take the four following sentences :
“The cat eats a fish”
“He eats a fish”
“He eats a fish with a knife”
“The dog takes a stick”
“The cat eats a fish”
These four sentences have a lot in common. The first and the fifth are totally equal while the first and the fourth should be equal in terms of structure. The first and the second should be very close since the only difference is inside the noun part. (”the cat” or “he” that is different in terms of structure). Finally the first and the third differ a bit since the third has one more part (”with a knife”).
The first step to apply on these sentences is the tokenization. For the first sentence, it leads to “[’The’, ’cat’, ’eats’, ’a’, ’fish’]” for example. Then we “tag” our tokens i.e we begin the parsing. It leads to “[(’The’, ’DT’), (’cat’, ’NN’), (’eats’, ’VBZ’), (’a’, ’DT’), (’fish’, ’NN’)]”. And here we had to make a choice that will decrease the quality of our results. The tree constitution that we use, which is very useful for its practicality, is not able to build a tree from both the words and the tags. It means that it builds the tree only from [’DT’, ’NN’, ’VBZ’, ’DT’, ’NN’,] without taking account of the words. The parsing tree is only made from the tags and the order of these tags. But it does not provide enough information about the sentence, especially about how these tags are related. We recall, as explained in the part on the trees, that from the same sentence it is possible to build several trees. The result of it is that for only one sentence a large frame of trees will be possible to build. Our example will then look at the consequences of this large frame of possibilities.
For example note that for the phrase “The cat eats a fish” can be produced :
Tree(“(NN,(DT,(DT.2,(VBZ,NN.2))))”) or again Tree(“(NN,(((DT,DT.2),NN.2),VBZ))”)

Now let us give the matrix of the symmetric distances for this 5 sentences:

	1	2	3	4	5
1 - “The cat eats a fish”	0	14	20	4	4
2 - “He eats a fish”	14	0	18	14	14
3 - “He eats a fish with a knife”	20	18	0	20	20
4 - “The dog takes a stick”	4	14	20	0	4
5 - “The cat eats a fish”	4	14	20	4	0

Figure 20: Matrix of distances for symmetric difference.

Of this matrix of distances we can describe several things about the behavior of the symmetric difference on our parses. First of all between 1 - “The cat eats a fish” and 5 - “The cat eats a fish” there is a distance of 4 units, which leads to think that the parsing did not produce the same tree in both cases. We just saw that this was a possibility. Between 1 - “The cat eats a fish” and 4 - “The dog takes a stick” that is the same in terms of structure we have also a difference of 4 units and actually on our different tests on these same sentences we have seen that this distance will be contained between 0 and 4 most of the time. (0 in the case the parsing has been made the same way in both cases). This is reassuring since it means that even if we do not have the same result of parsing for the same sentences (1, 4, 5 are supposed to produce the same tree since they all have the same structure or tag), the distance between them is very few. Indeed, 4 is very little compared to the rest of the distances in the table.

In the rest of the table we can have some informations about the way our symmetric difference will perform. The difference of 14 units between 1 and 2 or 20 units between 1 and 3 is also quite corresponding to our attempts for a good distance since it gives more difference in adding a new part to the sentence (“with a knife”) than in changing one part (in the noun part changing “the cat” by “he”).

We can also see some reasonable coherence : the second 2nd and the 3rd are both at equal distance from the 1th, 4th, 5th that are supposed to be the same in terms of structure. Furthermore the 2nd and the 3rd are closer than the 1th is from the 3rd (since they have “he” in common)

We will now present the results of the permutation tests. The number of permutations has been reduced to facilitate the computation.

One of the first thing we can note is that the average and more frequent p-value (also almost the lesser) is around 0.05. Regarding the previous assumptions assessed, we were not supposed to have anything more than a weak presumption against the null hypothesis. It means that approximately all the speeches come from the same distribution. But there are significant differences between the p-values and we think that there is a possible exploitation of these results. Instead of being focused on each p-value, we will next focus on the results that are significantly higher that this lesser value of 0.05. This treatment corresponds to the assumption that a weak presumption in this case still affords a notable presumption.

In the first table we have basically almost all of the values that are included between 0 and 0.1 i.e all of our values present a weak presumption against the null hypothesis. In this case it corresponds to a significant presumption that we have to take into account. Hence we can see that we do not have any presumption against the two terms of Obama coming from the same distribution. It seems that our test is able to recognize the same structure in the speeches of Obama through his two terms. That is very interesting since the other methods found a presumption against the null hypothesis in this case (in general previous methods found that speeches between the first and the second term were not coming from the same distribution for each president). Since we are dealing with structures and not with semantics, this result is coherent and is a real addition to our previous results. Note that in the first table, there is also no weak presumption against the null hypothesis between the second term of Bush and the first term of Clinton. We have no explanation about this. It could be a noise coming from the weakness of our computations. The p-value is only equal to 0.1181 which is only lightly superior to 0.1 (the limit set for the weak presumption).

The second table is providing very interesting results as well. We can see that there are no presumption against the null hypothesis in two of the three cases where we compare presidents against themselves. Indeed Clinton “inside” against Clinton “outside” gives a p-value of 0.1528, and we get 0.0490 and 0.2790 for Bush and Obama respectively. This is a very important result because it means that the structure of each president (except Bush) keep its coherence and can be recognized through the whole period of eight years. We saw in the previous methods that our methods were barely able to recognize presidents between their first and second term but were very good at recognizing presidents between their first and last years and their central term. We see the same kind of result here. Other notes can be made such as the p-value of 0.1047 that is barely higher that our limit for weak presumption. This p-value is reached for the testing of the null hypothesis between Clinton central term and Obama central term. This could be very interesting and could be assimilated to a similarity in “style of structure” between the two presidents after the beginning of their term. Note also the very high value of 0.1809 between Bush “outside” and Clinton “inside”. Their structures appear to be connected and we do not have any presumption against them coming from the same distribution.

	Clinton first term	Clinton second term	Bush first term	Bush second term	Obama first term	Obama second term
Clinton first term		0.0847	0.0547	**0.1181**	0.0747	0.0490
Clinton second term	0.0847		0.0480	0.0480	0.0590	0.0485
Bush first term	0.0547	0.0480		0.0523	0.0490	0.0495
Bush second term	**0.1181**	0.0480	0.0523		0.0542	0.0480
Obama first term	0.0747	0.0590	0.0490	0.0542		**0.1661**
Obama second term	0.0490	0.0485	0.0495	0.0480	**0.1661**	

Table 16: Permutation tests (terms of each president) using Tree representation, symmetric difference and statistic test total distances at sentence level.

	Clinton 93-94—99-00	Clinton 95-98	Bush 01-02—07-08	Bush 03-06	Obama 09-10—15-16	Obama 11-14
Clinton 93-94—99-00		**0.1528**	0.0485	0.0476	0.05	0.0552
Clinton 95-98	**0.1528**		**0.1809**	0.0490	0.0523	0.1047
Bush 01-02—07-08	0.0485	**0.1809**		0.0490	0.0519	0.0766
Bush 03-06	0.0476	0.0490	0.0490		0.0480	0.0476
Obama 09-10—15-16	0.05	0.0523	0.0519	0.0480		**0.2790**
Obama 11-14	0.0552	0.1047	0.0766	0.0476	**0.2790**	

Table 17: Permutation tests (inside and outside terms) using Tree representation, symmetric difference and statistic test total distances at sentence level.

Here the p-values of the full speeches (i.e all the sentences of each speech) against each other : 0.08571429 (Clinton-Bush) , 0.09952381 (Clinton - Obama) , 0.06190476 (Obama - Bush). These last values are also interesting to analyze : if we assimilate the p-value between the speeches of Clinton and Obama that is 0.099 to 0.1 we are exactly at the limit of the weak presumption. It means that we have more or less (since the low quality of our accuracy with these tests) no presumption against the structures of the speeches of Obama and Clinton coming from the same distribution. The two other p-values (0.06 and 0.08) show weak presumptions in both other cases.

8 Conclusions

8.1 Discussions

We have seen various methods and we are able to provide different answers regarding the questions asked i.e the comparisons that we wanted to make proposed in the methodology.

In the following section, we underline the pros and cons about the different text representations.
We have seen three different representations along this text. First the Vector Space Model that was represented by either the sparse Bag of Words alone or added to a Tf-Idf method. We want to compare it to the topic model with the use of a Latent Dirichlet Allocation. These two representations try to provide a good semantic "meaning" of a text and thus a comparison for which modeling is more pertinent for representing the semantics is pertinent. Let us compare results for the permutation tests where we will base for the total distances statistic test and the Aitchison distance for Topic model against the Jaccard distance for Vector Space Model.
For the case of topic model and Vector Space Model we note that the results have some accuracy since most of the p-values are between 0 and 0.05. That means that no presumption is "only" weak : when there is a presumption against the null hypothesis, this is strong. Now for the case of the first table where we compare each term of each president, we can see that there is a severe difference between the two. Indeed the Aitchison distance will note that there is no hypothesis against the speeches of the two terms of Bush coming from the same distribution while the Jaccard distance states that there is a presumption. Since all the other p-values stated by the topic model with Aitchison distance are very low, and since we could expect the two terms of Bush coming from the same distribution, we tend to think that the topic model has here a real advantage over its counterpart the Vector Space Model since it is "able" to recognize the semantic of Bush through its two terms (if we admit that there are really from the same distribution). The rest of the values from the first table don't give very significant differences between the two methods.
In the second table we again do not see significant differences, only that the low p-value of 0.1273 between the Obama "inside" and the Obama "outside" for the Vector Space Model is leading to a very low doubt about the equality while the p-value of 0.7108 for the topic model case do not let place to doubt. Since we expect these two samples to come from the same distribution, we give again a preference to the topic model to provide more sure p-values.
Also few words can be said about the clusterings. We have seen that K-means and hierarchical clustering give very different results themselves. What we can say is that the topic model gives very consistent results between K-means, hierarchical clustering and between the different methods of hierarchical clustering. This consistence let us think that there is more pertinence to prefer clustering with Topic Model.
Finally we have also introduced an other representation that is the tree representation for the syntax of a text. Note here that even if this is not very pertinent to compare it to the other methods (they are arguing about semantic) we can look about the quality of the results it provides and even if we can have some doubts about pertinence of the test since our lack of accuracy and the lack of surety in the presumptions against the null hypothesis, the permutation tests seem to recognize the link between the two terms of Obama for example, when this is not the case for the semantic tests.

In this part, we will focus on the advantages and disadvantages about the different distances for text representations.
First, for the case of Vector Space Model, we would like to provide a real comparison of Cosine Dissimilarity against the Jaccard distance. The result seems without appeal since there is significant differences between the two. Indeed we can see that the p-value for the test of the null hypothesis for Obama "outside" against Obama "inside" is 0.0568 only for the cosine part when there is no presumption for the Jaccard distance. Since we assume there are coming from the same distributions (since it is also the cases for the topic models) then the Cosine Dissimilarity has a weak presumption probably not suited. The hierarchical clustering is also in favor of the

Jaccard index since for the cosine part, it assimilates Obama and Clinton to gives a very special role to the first speech of Bush. This is an interesting part but regarding the results also for the topic model, the results for the Jaccard index (except for the single method) seem to be more trustworthy : it gives to each president one cluster (except some details such as the first speech of Bush)
Secondly for the case of the distances of topic model, we see that the main difference is about the fact the Aitchison distance is the only one able to provide no presumption against the null hypothesis between the two terms of Bush. The rest of the results are very alike but this significant difference leads us to think that the Aitchison distance might be more interesting.

We will here speak about the statistic tests for permutations. First of all we saw different comparisons of the T-statistic test against the total distances test. Let us note that since the Fréchet mean has not proven to be easily computable or useful in the case it is, the T-statistic test had to be performed with an euclidean mean or a geometrical mean (in the case of topic models). Consequently the T-statistic test has the problem of its coherence as explained above. Furthermore this T-statistic test seems to provide significant lesser results. Indeed if we look at the permutations tests for cosine, we have a score between Obama "inside" and Obama "outside" of 0.0568 for the total distances test and of 0.0331 for the T test. That means that we pass from a weak presumption against the null hypothesis in this case (hypothesis that we expect from our intuition to be true) to a strong presumption. In the same idea, a lot of other values seem to be under evaluated with the T-test comparing to the total distances test.

Finally we will speak about the data and their exploitation.
With our results we can make some observations and also conclusions confronting our expectations at the beginning of this study.
The two democrat presidents of this study are closer from each other than they are from the republican president respectively. That can be seen on the matrices of similarity computed with the cosine similarity but also with for example the K-means of the Cosine Dissimilarity for two clusters where we will see that Obama and Clinton are clustered together while Bush is in a specific cluster. That fits our expectations since semantic of the speeches of presidents should be influenced by their respective political part.
Secondly we can note a real change between each term. What that means is the fact that almost every permutation test between the two terms of the presidents were showing clear presumption against the hypothesis that they come from the same distribution. That was not particularly expected since we would have better trusted a kind of coherence through the terms. But the evidences in the matrices for example but particularly in the tables of permutations are very coherent. Must of tables show (in average regarding our results) that no one of the three presidents has his speeches from the first term and the second term coming from the same distribution.
Nevertheless note that there is a real coherence through the speeches of the presidents even if there is the change between the two terms. Indeed we see on the second table of permutation tests that between the inside speeches of the presidents and the outside (the end and the beginning) there are very often no presumptions against them coming from the same distribution. This comes to give credit to the fact there is a real convenience to use statistics on politicians since their speeches are coherent between them, or at least have enough coherence to be studied.
An other note that we can make is the importance given to the first speech. For example the first speech of Clinton looks more like the speeches of Obama according to several distances and tests : the K-means for cosine similarity and three clusters, the K-means for the Aitchison distance and the Jensen-Shannon-Divergence so the whole topic model. But the first speech of Bush, for some obvious reasons that we will develop afterwards, has also a very specific place since it is classified as a Clinton speech for example according to the hierarchical clustering for Aitchison distance and Ward's or Average's method or the hierarchical clustering for the Jensen Shannon Divergence for all the methods except the single one.
This particular place of the first speech of Bush holds its place very probably regarding the particular event of 2001. We can note it indeed from the hierarchical clustering with the Cosine Dissimilarity that it is classified as a single cluster itself. The hierarchical clustering done with the Jaccard distance and single method puts as well Obama speeches and Clinton speeches

together.
We let here some factual notes about specific presidents. First Obama seems to be more consistent than the two others along its two terms. Indeed its matrix of dissimilarity as well as the tree representation tend to present the two terms of Obama to be not so different from each other. Indeed there seems to be a syntactic relation than there is not in the two other presidents (at least not one we found with our method) and the cosine matrix shows clear relations.
Note also that on the permutation tests with tree representations, the speeches of Bush between the outside and the inside seem to be the only one that do seem to be presumed not coming from the same distribution. Indeed we get a score of 0.0490 for Bush “outside” against Bush “inside” with tree representation while we get 0.1528 for Clinton and 0.2790 for Obama. Since the accuracy of our test is not solid enough to provide a conclusion out of criticisms, we will only limit to the observation that Bush may be less coherent in a syntactic way than its two counterparts.

8.2 Conclusions and Future Work

In the previous discussions we have been able to gather observations and to provide discussions. We can thus provide conclusions here, some recommendations extracted from our results. First of all for the case of representations, we have seen that topic model seems to outperform classic Vector Space Model. This was expected since Latent Dirichlet Allocation is more recent and has been in other paper sometimes outperformed more classic models. Secondly the Aitchison distance seems to outperform the Jensen-Shannon-Divergence as well as the Jaccard distance gives more trustworthy results than the Cosine Dissimilarity. Finally the total distances test has two advantages on its counterpart. The first one is its coherence and the second one is the results that simply seem better.

From this thesis we can say that we have reached our goals. We have provided algorithms to perform permutation tests, clustering and classification and we are able to propose appropriate representations and distance to perform this at its best. We were also able to present the advantages of several of these methods. We have provided methods to answer to a query-demand, to a classification demand or again and this one the most important a test of equality in the sense of a distribution. Finally we have answered to the main aim : we have presented a methodology for testing text-valued data in the frame of Object-Oriented Data Analysis.

This paper has voluntary omitted very new methods such as predictive-word embeddings that are already very popular but the tests on them to be tested against the classic methods that we have presented here may prove they surpass them.
Furthermore we have discussed a few about exploitation of syntax structure and what results it could bring. Tree structures that have been discussed here are very few developed in literature compared to the previous methods we have seen. Note that a method such as the Word2Vec is also exploiting the syntax of a text to give conclusions. But more precisely, the exploitation of the grammar of a text could maybe lead to good conclusions.

Note that a good test to answer to our introduction regarding the Isaac Asimov’s machine would be to perform a permutation test between the text of the politician we would like to extract the substance and a large frame of texts from many politicians. We make the hypothesis that indeed from many texts of politicians, the average text would be close to no substance and would be close of the most used words of politicians i.e the less new or the less “full” or real ideas and real substance. So we expect that the production of the mixing of many politicians texts gives a non-sense politician text, one without substance. Then a permutation test as presented here should be able to detect if our politician speech that we want to test is alike this mixing and so if it is alike this “non-sense”.

Appendices

	Clinton first term	Clinton second term	Bush first term	Bush second term	Obama first term	Obama second term
Clinton first term		0.0156	0.0004	0.0268	0.0249	0.0102
Clinton second term	0.0156		0.0129	0.0090	0.0201	0.0002
Bush first term	0.0004	0.0129		0.0258	0.0122	0.0005
Bush second term	0.0268	0.0090	0.0258		0.0245	0.0255
Obama first term	0.0249	0.0201	0.0122	0.0245		0.0004
Obama second term	0.0102	0.0002	0.0005	0.0255	0.0004	

Table 18: Permutation tests (terms of each president) using sBoW-Jaccard distance and T-statistic test with a euclidean mean at document level.

	Clinton	Clinton	Bush	Bush	Obama	Obama
	93-94—99-00	95-98	01-02—07-08	03-06	09-10—15-16	11-14
Clinton 93-94—99-00		**0.3966**	0.0072	0.0276	0.0163	0.0160
Clinton 95-98	**0.3966**		0.0237	0.0183	0.0189	0.0109
Bush 01-02—07-08	0.0072	0.0237		**0.3755**	0.0232	0.0009
Bush 03-06	0.0276	0.0183	**0.3755**		0.0182	0.0270
Obama 09-10—15-16	0.0163	0.0189	0.0232	0.0182		**0.1474**
Obama 11-14	0.0156	0.0109	0.0009	0.0270	**0.1474**	

Table 19: Permutation tests (inside and outside terms) using sBoW-Jaccard distance and T-statistic test with a euclidean mean at document level.

	Clinton	Clinton	Bush	Bush	Obama	Obama
	first term	second term	first term	second term	first term	second term
Clinton first term		0.0195	0.0156	0.0051	0.0014	0
Clinton second term	0.0195		0.0162	0.0122	0	0.0142
Bush first term	0.0156	0.0162		0.0547	0.0003	0
Bush second term	0.0051	0.0122	0.0547		0.0217	0.004
Obama first term	0.0014	0	0.0003	0.0217		0.0056
Obama second term	0	0.0142	0	0.004	0.0056	

Table 20: Permutation tests (terms of each president) using LDA - Aitchison distance and T-statistic test with a euclidean mean at document level.

	Clinton 93-94—99-00	Clinton 95-98	Bush 01-02—07-08	Bush 03-06	Obama 09-10—15-16	Obama 11-14
Clinton 93-94—99-00		**0.3790**	0.0227	0	0.0061	0
Clinton 95-98	**0.3790**		0.0182	0.0020	0.0135	0.0240
Bush 01-02—07-08	0.0227	0.0182		**0.3590**	0	0.0035
Bush 03-06	0	0.0020	**0.3590**		0.0122	0.0122
Obama 09-10—15-16	0.0061	0.0135	0	0.0122		**0.5472**
Obama 11-14	0	0.0240	0.0035	0.0122	**0.5472**	

Table 21: Permutation tests (inside and outside terms) using LDA - Aitchison distance and T-statistic test with a euclidean mean at document level.

	Clinton first term	Clinton second term	Bush first term	Bush second term	Obama first term	Obama second term
Clinton first term		0.0083	0.0212	0.0282	0.0056	0.0095
Clinton second term	0.0083		0.0182	0.0285	0.0047	0.0122
Bush first term	0.0212	0.0182		0.0156	0.0056	0.0035
Bush second term	0.0282	0.0285	0.0156		0.0115	0.0061
Obama first term	0.0056	0.0047	0.0056	0.0115		0.0122
Obama second term	0.0095	0.0122	0.0035	0.0061	0.0122	

Table 22: Permutation tests (terms of each president) using LDA-JSD and T-statistic test with a euclidean mean at document level.

	Clinton 93-94—99-00	Clinton 95-98	Bush 01-02—07-08	Bush 03-06	Obama 09-10—15-16	Obama 11-14
Clinton 93-94—99-00		**0.3364**	0.0227	0.0009	0.0003	0.0222
Clinton 95-98	**0.3364**		0.0162	0.0279	0.0188	0.0142
Bush 01-02—07-08	0.0227	0.0162		**0.3497**	0.0222	0.0176
Bush 03-06	0.0009	0.0279	**0.3497**		0.0217	0.0077
Obama 09-10—15-16	0.0003	0.0188	0.0222	0.0217		**0.4836**
Obama 11-14	0.0222	0.0142	0.0176	0.0077	**0.4836**	

Table 23: Permutation tests (inside and outside terms) using LDA-JSD and T-statistic test with a euclidean mean at document level.

References

[1] J. Marron and A. Alonso. “Overview of object oriented data analysis.” In: *Biometrical Journal* 56 (2014), pp. 732–753.

[2] H. Wang and J. Marron. “Object oriented data : set of trees”. In: *The Annals of Statistics* 35 (2007), pp. 1849–1873.

[3] S. Wei, C. Lee, and J. Marron. “Direction-projection-permutation for high-dimensional hy- pothesis tests.” In: *Journal of Computational and Graphical Statistics* 25 (2016), pp. 549–569.

[4] I. Dryden, A. Koloydenko, and D. Zhou. “Non-euclidean statistics for covariance matrices, with application to diffusion tensor imaging.” In: *Annals of Applied Statistics* 3 (2009).

[5] B. Rademakers. “Nuclear Inst. and Methods in Physics Research, A,” in: *Genetics* 389 (1997), pp. 81–86.

[6] N. Cristianini and J. Shawe-Taylor. “An Introduction to Support Vector Machines and Other Kernel-based Learning Methods.” In: *Cambridge university press, Cambridge, UK.* (2000).

[7] V. N. Vapnik. “The Nature of Statistical Learning Theory.” In: *Springer* (1999).

[8] S. Osinski and D. Weiss. “Carrot2 Project”. In: *Open Source Search Results Clustering Engine* ().

[9] “Introduction to Information Retrieval”. In: *nlp.stanford.edu.* (2016), p. 349.

[10] T. Mikolov and et al. “Distributed Representations of Words and Phrases and their Compositionality”. In: *Advances in neural information processing systems* (2013), pp. 3111–3119.

[11] S. Scott and S. Matwin. “Feature Engineering for Text Classification”. In: *Proceedings of ICML-99, 16th International Conference on Machine Learning* (1999).

[12] G. Salton, A. Wong, and C. Yang. “A Vector Space Model for Automatic Indexing”. In: *ACM* 18 (1975).

[13] H. Luhn. “A Statistical Approach to Mechanized Encoding and Searching of Literary Information”. In: *IBM Journal of research and development* (1957).

[14] K. Sparck. “A statistical interpretation of term specificity and its application in retrieval”. In: *Journal of Documentation* 28 (1972), pp. 11–21.

[15] S. Deerwester, S. Dumais, and G. Furnas. “Indexing by Latent Semantic Analysis”. In: *Journal of the association for information science and technology* (1990).

[16] E. Gabrilovich and S. Markovitch. “Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis”. In: *Bibliometrics Data Bibliometrics* (2007).

[17] P. Turney and P. Pantel. “From Frequency to Meaning: Vector Space Models of Semantics”. In: *Journal of Artificial Intelligence Research* 37 (2010), pp. 141–188.

[18] M. Sahlgren. “An Introduction to Random Indexing”. In: *Proceedings of the Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering* (2005).

[19] M. Sahlgren and R. Cöster. “Using Bag-of-Concepts to Improve the Performance of Support Vector Machines in Text Categorization”. In: *Proceedings of the 20th international conference on Computational Linguistics* 487 (2004).

[20] T. Hofmann. “Probabilistic Latent Semantic Analysis”. In: *Uncertainity in Articial Intelligence* (1999).

[21] D. M. Blei, Y. N. Andrew, and I. J. Michael. “Latent dirichlet allocation”. In: *Journal of Machine Learning Research* 3 (2003), pp. 993–1022.

[22] J. K. Pritchard, M. Stephens, and P. Donnelly. “Inference of population structure using multilocus genotype data”. In: *Genetics* 155 (2000), pp. 945–959.

[23] W. Li and A. McCallum. “Pachinko Allocation: DAG-Structured Mixture Models of Topic Correlations”. In: *International Conference on Machine Learning* (2006), pp. 577–584.

[24] D. Blei. “Probabilistic topic models”. In: *Proceedings of Machine Learning Research* (2012).

[25] R. Collobert and J. Weston. “A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning”. In: *Proceedings of the 25th international conference on Machine learning* 8 (2008), pp. 160–167.

[26] M. Baroni, G. Dinu, and G. Kruszewski. “Don’t count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors”. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics* 1 (2014).

[27] V. Le Quoc and M. T. “Distributed Representations of Sentences and Documents”. In: *31st International Conference on Machine Learning, ICML* 84 (2014).

[28] T. Quoc V. Leand Mikolov. “Distributed Representations of Sentences and Documents”. In: *Proceedings of Machine Learning Research* 32 (2014).

[29] J. Pennington, R. Socher, and C. Manning. “GloVe: Global Vectors for Word Representation”. In: *Stanford NLP group* (2014).

[30] N. Chomsky. “Three models for the description of language”. In: *IRE Transactions on Information Theory* 2 (1956), pp. 113–124.

[31] S. Bird, E. Klein, and E. Loper. “Natural Language Processing with Python, Analyzing Text with the Natural Language Toolkit”. In: *O’Reilly Media* (2009).

[32] J. N. Singh. "A Comparative Study on Approaches of Vector Space Model in Information Retrieval". In: *Annals of Applied Statistics* (), pp. 37–40.

[33] J. Bullinaria and J. Levy. "Extracting semantic representations from word cooccurrence statistics: A computational study". In: *Behavior Research Methods* 39 (2007), pp. 510–526.

[34] A. Huang. "Similarity measures for text document clustering". In: *Proceedings of the Sixth New Zealand Computer Science Research Student Conference, Christchurch, New Zealand : Computer Science Research Student Conference* (2008), pp. 49–56.

[35] L. Lee. "Measures of distributional similarity". In: *In Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics* (1999), pp. 25–32.

[36] P. Jaccard. "Étude comparative de la distribution florale dans une portion des Alpes et des Jura". In: *Bulletin de la Société Vaudoise des Sciences Naturelles* 37 (1901), pp. 547–579.

[37] H. Späth. "The minisum location problem for the Jaccard metric". In: *Operations-Research-Spektrum* 3 (1981), pp. 91–94.

[38] J. Aitchison. "On criteria for measures of compositional differences". In: *Math. Geology* 24 (1992), pp. 365–380.

[39] J. M. Fernandez, C. B. Vidal, and V. Pawlowsky-Glahn. "Measures of difference for compositional data and hierarchical clustering methods". In: ().

[40] S. Kullback and R. Leibler. "On information and sufficiency". In: *Annals of Mathematical Statistics.* 22 (1951), pp. 79–86.

[41] E. J. Orowski and J. M. Borwein. "The HarperCollins Dictionary of Mathematics". In: *New York: HarperCollins* (1991).

[42] C. E. Ginestet, J. Li, P. Balanchandran, S. Rosenberg, and E. D. Kolaczyk. "Hypothesis testing for network data in functional neuroimaging". In: *Annals of Applied Statistics* (2017).

[43] Bóna and Miklós. "Combinatorics of Permutations". In: *Chapman Hall-CRC* (2004).

[44] R. L. Berger and G. Casella. "Statistical Inference". In: *Duxbury Press, Second Edition* (2001).

[45] A. Stuart, K. Ord, and S. Arnold. "Classical Inference and the Linear Model". In: *Kendall's Advanced Theory of Statistics* (1999).

[46] "e-Handbook of Statistical Methods". In: *NIST/SEMATECH http://www.itl.nist.gov/div898/handbook/* (2012).

[47] K. Pearson. "On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling". In: *Philosophical Magazine* 5 (1900), pp. 157–175.

[48] F. Pesarin and L. Salmaso. "Permutation Tests for Complex Data". In: *Wiley* (2010).

[49] B. Phipson and G. K. Smyth. "Permutation P-values Should Never Be Zero: Calculating Exact P-values When Permutations Are Randomly Drawn". In: *Statistical Applications in Genetics and Molecular Biology* 9 (2010), pp. 1–12.

[50] R. Nuzzo. "Scientific method: Statistical errors". In: *Nature* 506 (2014), pp. 150–152.

[51] H. Steinhaus. "Sur la division des corps matériels en parties". In: *Bull. Acad. Polon. Sci. (in French)* 4 (1957), pp. 801–804.

[52] A. K. Jain. "Data clustering: 50 years beyond K-means". In: *Pattern Recognition Letters* 31 (2010), pp. 651–666.

[53] R. Kuo, L. Ho, and C. Hu. “Integration of self-organizing feature map and K-means algorithm for market segmentation”. In: *Computers and Operations Research* 29 (2002), pp. 1475–1493.

[54] J. Honarkhah Mand Caers. “Stochastic Simulation of Patterns Using Distance-Based Pattern Modeling”. In: *Mathematical Geosciences* (2010).

[55] P. Burrougha, P. Van Gaansa, and R. MacMillanb. “High-resolution landform classication using fuzzy k-means”. In: *Fuzzy Sets and Systems* 113 (2000), pp. 37–52.

[56] D. Al Blesh, M. Braik, and S. Bani-Ahmad. “Detection and Classification of leaf diseases using K-means based segmentation and Neural-networks based Classification”. In: *Information Technology Journal* (2011), pp. 265–267.

[57] R. Lior and O. Maimon. “Clustering methods”. In: *Data mining and knowledge discovery handbook* (2005), pp. 321–352.

[58] J. Ward. “Hierarchical Grouping to Optimize an Objective Function”. In: *Journal of the American Statistical Association* 58 (1963), pp. 236–244.

[59] S. J.-M. Moran. “An open source toolkit for quantitative historical linguistics.” In: *In Proceedings of the 51st Conference of the Association for Computational Linguistics* (2013).

[60] R. Arun, V. Suresh, and C. E. Madhavan. “On finding the natural number of topics with latent dirichlet allocation: Some observations”. In: *In Advances in knowledge discovery and data mining, Springer Berlin Heidelberg* (2010), pp. 391–402.

[61] C. Juan, X. Tian, and J. L. “A density-based method for adaptive lDA model selection”. In: *Neurocomputing — 16th European Symposium on Artificial Neural Networks 2008* 72 (2009), pp. 1775–1781.

[62] R. Deveaud, E. SanJuan, and P. Bellot. “Accurate and effective latent concept modeling for ad hoc information retrieval”. In: *Document numérique 17* (2014), pp. 61–84.

[63] T. Griffiths and M. Steyvers. “Finding scientific topics”. In: *Proceedings of the National Academy of Sciences* 101 (2004), pp. 5228–5235.

TRITA TRITA-SCI-GRU 2018:014

www.kth.se