Modeling and learning dependencies with copulas in latent topic models

Hesam Amoualian

Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : Informatique

Arrêté ministériel : 25 mai 2016

Présentée par

Hesam AMOUALIAN

Thèse dirigée par **Eric GAUSSIER**, UGA
et codirigée par **Massih-Reza AMINI**, Professeur, Université Grenoble Alpes

préparée au sein du **Laboratoire Laboratoire d'Informatique de Grenoble**
dans **l'École Doctorale Mathématiques, Sciences et technologies de l'information, Informatique**

Mise à l'échelle des modèles latente Sujet / Classe de grandes collections de données et les flux

Scaling Latent Topic/Class Models to Big Data Collections and Streams

Thèse soutenue publiquement le **12 décembre 2017**,
devant le jury composé de :

Monsieur ERIC GAUSSIER
PROFESSEUR, UNIVERSITE GRENOBLE ALPES, Directeur de thèse
Monsieur MASSIH-REZA AMINI
PROFESSEUR, UNIVERSITE GRENOBLE ALPES, Co-directeur de thèse
Madame MARIANNE CLAUSEL
PROFESSEUR ASSOCIE, UNIVERSITE GRENOBLE ALPES, Examinateur
Madame MARIE-FRANCINE MOENS
PROFESSEUR, KU LEUVEN - BELGIQUE, Président
Monsieur JULIEN VELCIN
MAITRE DE CONFERENCES, UNIVERSITE LYON 2, Rapporteur
Monsieur WEI LU
PROFESSEUR ASSISTANT, UNIV. DE TECH. ET DE DESIGN DE SINGAPOUR, Examinateur

This thesis is dedicated to my parents for their love,
my friends for their support and encouragement
without whom it wouldn't be possible

Abstract

This thesis focuses on scaling latent topic models for big data collections, especially when document streams. Although the main goal of probabilistic modeling is to find word topics, an equally interesting objective is to examine topic evolutions and transitions. To accomplish this task, we propose in Chapter 3, three new models for modeling topic and word-topic dependencies between consecutive documents in document streams. The first model is a direct extension of Latent Dirichlet Allocation model (LDA) and makes use of a Dirichlet distribution to balance the influence of the LDA prior parameters with respect to topic and word-topic distributions of the previous document. The second extension makes use of copulas, which constitute a generic tool to model dependencies between random variables. We rely here on Archimedean copulas, and more precisely on Franck copula, as they are symmetric and associative and are thus appropriate for exchangeable random variables. Lastly, the third model is a non-parametric extension of the second one through the integration of copulas in the stick-breaking construction of Hierarchical Dirichlet Processes (HDP). Our experiments, conducted on five standard collections that have been used in several studies on topic modeling, show that our proposals outperform previous ones, as dynamic topic models, temporal LDA and the Evolving Hierarchical Processes, both in terms of perplexity and for tracking similar topics in document streams. Compared to previous proposals, our models have extra flexibility and can adapt to situations where there are no dependencies between the documents.

On the other hand, the "Exchangeability" assumption in topic models like LDA often results in inferring inconsistent topics for the words of text spans like noun-phrases, which are usually expected to be topically coherent. In Chapter 4, we propose copulaLDA (copLDA), that extends LDA by integrating part of the text structure to the model and relaxes the conditional independence assumption between the word-specific latent topics given the per-document topic distributions. To this end, we assume that the words of text spans like noun-phrases are topically bound and we model this dependence with copulas. We demonstrate empirically the effectiveness of copLDA on both intrinsic and extrinsic evaluation tasks on several publicly available corpora.

To complete the previous model (copLDA), Chapter 5 presents an LDA-based model that generates topically coherent segments within documents by jointly segmenting documents and assigning topics to their words. The coherence between topics is ensured through a copula, binding the topics associated to the words of a segment. In addition, this model relies on both document and segment specific topic distributions so as to capture fine-grained

differences in topic assignments. We show that the proposed model naturally encompasses other state-of-the-art LDA-based models designed for similar tasks. Furthermore, our experiments, conducted on six different publicly available datasets, show the effectiveness of our model in terms of perplexity, Normalized Pointwise Mutual Information, which captures the coherence between the generated topics, and the Micro F1 measure for text classification.

Résumé

Ce travail de thése a pour objectif de s'intéresser à une classe de modèles hiérarchiques bayesiens, appelés topic models, servant à modéliser de grands corpus de documents et ceci en particulier dans le cas où ces documents arrivent séquentiellement. Pour cela, nous introduisons au Chapitre 3, trois nouveaux modèles prenant en compte les dépendances entre les thèmes relatifs à chaque document pour deux documents successifs. Le premier modèle s'avère être une généralisation directe du modèle `LDA` (Latent Dirichlet Allocation). On utilise une loi de Dirichlet pour prendre en compte l'influence sur un document des paramètres relatifs aux thèmes sous jacents du document précédent. Le deuxième modèle utilise les copules, outil générique servant à modéliser les dépendances entre variables aléatoires. La famille de copules utilisée est la famille des copules Archimédiens et plus précisément la famille des copules de Franck qui vérifient de bonnes propriétés (symétrie, associativité) et qui sont donc adaptés à la modélisation de variables échangeables. Enfin le dernier modèle est une extension non paramétrique du deuxième. On intègre cette fois ci les copules dans la construction stick-breaking des Processus de Dirichlet Hiérarchique (`HDP`). Nos expériences numériques, réalisées sur cinq collections standard, mettent en évidence les performances de notre approche, par rapport aux approches existantes dans la littérature comme les dynamic topic models, le temporal `LDA` et les Evolving Hierarchical Processes, et ceci à la fois sur le plan de la perplexité et en terme de performances lorsqu'on cherche à détecter des thèmes similaires dans des flux de documents. Notre approche, comparée aux autres, se révèle être capable de modéliser un plus grand nombre de situations allant d'une dépendance forte entre les documents à une totale indépendance.

Par ailleurs, l'hypothèse d'échangeabilité sous jacente à tous les topics models du type du `LDA` amène souvent à estimer des thèmes différents pour des mots relevant pourtant du même segment de phrase ce qui n'est pas cohérent. Dans le Chapitre 4, nous introduisons le copulaLDA (`copLDA`), qui généralise le `LDA` en intégrant la structure du texte dans le modèle of the text et de relaxer l'hypothèse d'indépendance conditionnelle. Pour cela, nous supposons que les groupes de mots dans un texte sont reliés thématiquement entre eux. Nous modélisons cette dépendance avec les copules. Nous montrons de manière empirique l'efficacité du modèle `copLDA` pour effectuer à la fois des tâches de nature intrinsèque et extrinsèque sur différents corpus accessibles publiquement. Pour compléter le modèle précédent (`copLDA`), le chapitre 5 présente un modèle de type `LDA` qui gén'ere des segments dont les thèmes sont cohérents à l'intérieur de chaque document en faisant de manière simultanée la segmentation des documents et l'affectation des thèmes à chaque

mot. La cohérence entre les différents thèmes internes à chaque groupe de mots est assurée grâce aux copules qui relient les thèmes entre eux. De plus ce modèle s'appuie tout à la fois sur des distributions spécifiques pour les thèmes reliés à chaque document et à chaque groupe de mots, ceci permettant de capturer les différents degrés de granularité. Nous montrons que le modèle proposé généralise naturellement plusieurs modèles de type LDA qui ont été introduits pour des tâches similaires. Par ailleurs nos expériences, effectuées sur six bases de données différentes mettent en évidence les performances de notre modèle mesurée de différentes manières : à l'aide de la perplexité, de la Pointwise Mutual Information Normalisée, qui capture la cohérence entre les thèmes et la mesure Micro F1 measure utilisée en classification de texte.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Numerous pieces of content are currently exchanged in social media, making them an important source of information. For example, people share, per month, 30 billion pieces of content on Facebook and over 5 billion tweets (see for example the site mashable.com). This importance is also reflected in the fact that, when searching for information online, 18% of the users directly search on social media sites (as Twitter, Facebook or blog sites), a proportion constantly growing. Searching, filtering, enriching and organizing this information, as well as being able to rapidly identify important new events, are major challenges faced by researchers from different communities, as information retrieval, data mining and machine learning.

Several approaches have been developed in the past to address these challenges, even though not at the scale and speed required by current data collections and streams. Among these different approaches, the ones based on latent topic/class analysis (as Latent Dirichlet Allocation proposed by [Blei et al., 2003]) or their hierarchical extensions are particularly interesting as they yield state-of-the-art results and allow one to categorize/annotate documents with existing taxonomies (filtering and enriching), to infer new taxonomies or complement existing ones (organizing) and to detect outliers and new events (event detection). However, current latent topic models have major drawbacks that prevent their use on large-scale collections and high-speed streams, like they are mainly static and do not take into account the dynamics of the data. The goal of this thesis is precisely to address these problems, by constructing new latent topic models able to handle dynamic data, and by designing new learning and inference methods able to provide good estimates of the parameters of the new models. In following, we state an introduction on language

models, generative methods, latent topic models and finally copula as a generic tool to capture dependencies between random variables.

A language model is a way to assign probability distribution over a sequence of words which are sampled from a big collection of data like vocabulary [Rosenfeld, 2000]. knowing a way to estimate the relative likelihood for different phrases and sentences is always useful in many language processing applications, especially when one generates text as output. The simplest type of language model may be equal to a probabilistic finite automaton with a single probability distribution for producing different words. This model generates a term and then decides whether to stop or keep searching for producing another term, so this model also desires a probability for making a decision on stopping or looping in the finishing state. This kind of model applies a probability distribution over any sequence of words. Using this structure, it can also be a model to generate long sentences or text according to its distribution.

We now try to explain some types of language models. To apply a probability distribution over sequences of words, it is always helpful to apply the chain rule to break the probability of a sequence of words down into the probability of each successive sampled word conditioned on previous words. For simplicity, we assume four words and the model can be as follows:

$$P(w_1w_2w_3w_4) = P(w_1)P(w_2|w_1)P(w_3|w_1w_2)P(w_4|w_1w_2w_3) \tag{1.1}$$

Here w_i shows the words which are based in each document of the collection. The simplest type of language model can be interpreted by unchaining all conditions in the context and estimates each word's probability independently. This kind of language model is called unigram language model and it is illustrated in Figure 1.1(a).

$$P_{uni}(w_1w_2w_3w_4) = P(w_1)P(w_2)P(w_3)P(w_4) \tag{1.2}$$

There are several complicated kinds of language model, as an example bigram language model, which keeps condition on the previous word for estimating the probabilities:

$$P_{bigram}(w_1w_2w_3w_4) = P(w_1)P(w_2|w_1)P(w_3|w_2)P(w_4|w_3) \tag{1.3}$$

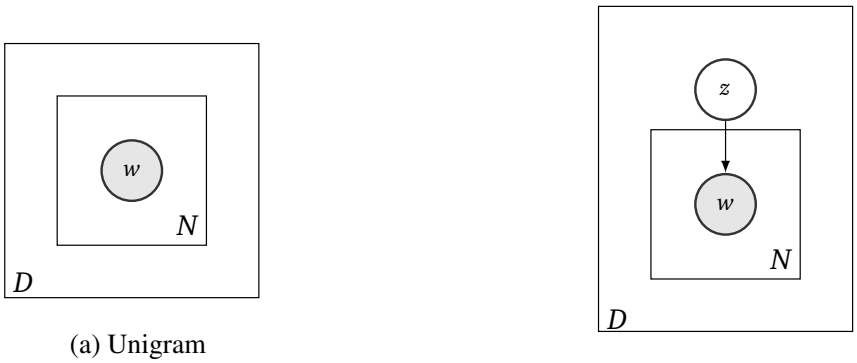
In the unigram language model structure, the order of words is meaningless. Even though there is no condition for generating the text, this model can still provide the

probability of a particular order of words. So, we can conclude a multinomial distribution between the words and infer this model as a multinomial model. Using these assumptions, this model refers to:

$$P(d) = \frac{(\sum wf_i)!}{\prod wf_i!} \prod P(w_i)^{wf_i} : \tag{1.4}$$

wf stands for the word occurrence frequency inside document d. If we incorporate a discrete random vector of topic variable z into the unigram model, we attain a mixture of unigrams model [Nigam et al., 2000]. Generating procedure for this mixture model that is illustrated in Figure 1.1(b), is as follows: each document is generated by firstly choosing a topic z then generating independently N words from the multinomial conditional distribution of $P(w|z)$. Then the probability of a document consisting of W words [Blei et al., 2003]:

$$P(W) = \sum_z P(z) \prod_{n=1}^{N} P(w_n|z). \tag{1.5}$$



(a) Unigram

(b) Mixture

Figure 1.1: Graphical models for Unigram and Mixture of Unigrams

Using this model, the word distributions can be interpreted as a representation of topics with the assumption that the model assigns only one topic for each document. Typically, this assumption is too restricting to have a precise model for a large corpus of words. The experimental results in [Blei et al., 2003] have proved this conclusion. As a way to avoid this problem, Latent Dirichlet Allocation (LDA) model introduced by [Blei et al., 2003] allows documents to obtain multiple topics with different probabilities. This problem

is fixed in LDA with integrating one additional parameter; in particular, the mixture of unigrams model has $k-1$ parameters associated with $P(z)$ as the probability of topics, where in LDA model there are k parameters associated with $P(\theta|\alpha)$ which is the probability distribution over topics and will be explained in sequel.

1.1 Topic models

Topic models are based on the concept that documents of a collection of words are mixtures of topics, where topics are vectors of probability distribution over words. In fact, a topic model is a generative model for the document and the words that belong to them. It makes a specific probabilistic procedure to generate words and consecutively documents. The procedure is as follows: for generating a new document, it first chooses a distribution over topics. Then, for each word in that document, one randomly chooses a topic according to this distribution and finally selects a word based on the topic which has been selected. Different statistical techniques and inferences can be used to reverse the whole process, presenting the matrix of topics that were assigned for generating a collection of documents. A generative model for documents is formed by a simple probabilistic sampling procedure that rules the way of generating words in documents based on the latent and hidden variables distributions. Observing the words of documents, the goal of topic model (fitting a generative model) is to find the most precise set of latent variables that can describe this observed data. Using this model, various set of documents can be produced by choosing words from a topic-word distribution depending on the weight of the topic in document-topic distribution. This generative process does not make any assumptions about the order of words and the way that they appear in documents. The only important information related to the model, is the number of times words occurred and chosen in the generative process. This is a well-known assumption, `bag-of-words` assumption, and is common to statistical language models like Latent Semantic Analysis (`LSA`, [Deerwester et al., 1990]) or the other topic models like Latent Dirichlet Allocation (`LDA`, [Blei et al., 2003] or Hierarchical Dirichlet Process (`HDP`, [Teh et al., 2006]). Of course, this is not a correct assumption when words-order contains important information regarding the content of a document or the relation between them. Later, we are going to consider this problem and the solutions.

As one of the leading statical topic models, the Latent Semantic Analysis (LSA) [Deerwester et al., 1990] or Latent Semantic Indexing (LSI) [Landauer and Dumais, 1997]) posits a linear topic model that refers to a matrix factorization way over the matrix of document-word in corpus C consists of c_{dw} as the count of occurrences of word w in document d. This model aims to find a low-rank approximation of the matrix C by factorizing it into two separate matrices. One of these matrices represents the relation between documents and topics, and the other shows the relation between topics and words. According to Eckart–Young–Mirsky theorem [Eckart and Young, 1936], having an $M \times N$ matrix of C and a positive integer k, a low-rank approximation of C with rank k will be a matrix of C_k with rank at most k which minimizes the Frobenius norm of the $C - C_k$. Applying Singular Value Decomposition (SVD) on $X = U\Sigma V^T$ we can conclude C_k. SVD chooses the K largest singular values of the matrix Σ and the corresponding values in the matrix U and V^T, then best rank K approximation of matrix C will be obtained. This low-rank approximation of C brings in a new representation regards to each document.
Although there are some advantages in usual vector space representation for a document like: homogeneous behaviors of queries and documents in terms of vectors, taking benefits from the induced computation score according to cosine similarity between vectors, the ability to put different weights to different words, and its application beyond the document retrieval to accomplish the tasks like clustering and classification, it is inadequate to cope with two fundamental problems which should be solved in natural languages. First, the Synonymy when two different words have the same meaning and second, Polysemy when the same word have different meanings. Latent semantic indexing or analysis deploys the SVD to compose a low-rank approximation for the word-document matrix, for a rank of k that is way smaller than the original rank of matrix C. Indeed, it maps each row and column of this matrix which is word occurrence in the corpus to a k-dimensional space. Then, one can apply cosine similarity between the vectors over this new representation to carry a clustering task out. LSI can be inferred as a soft clustering by interpreting each dimension of the new reduced space as a cluster, then the fractional membership of the cluster will be the value that a document owed on this dimension. These clusters can be recognized as ground topics that can explain the structure and the meaning of the collection. In this model, the SVD helps to obtain rows of U as a representation of documents, and rows of V^T the representation of topics. Then each document can be exhibited as a linear combination of topics. As a conclusion, the Latent Semantic Analysis gains three characteristics in topic models: the semantic information can be stemmed from a co-occurrence matrix of word-document, the dimension of the model is reduced to very

small value, and also the words and documents now can be showed as points in Euclidean space.

Different probabilistic topic models have also been used to analyze the content of documents and the relation between the words. All models share the same fundamental belief that a document contains of a mixture of topics but with slight difference in terms of statistical assumptions. Probabilistic Latent Semantic Indexing (pLSI [Hofmann, 1999]) is one of probabilistic topic models which widely deployed for document summarization as an application in the topic model. The pLSI model, represented in Figure 1.2, claims that a document d and a word w in the whole collection are conditionally independent given an unobserved latent topic z:

$$P(d, w_n) = P(d) \sum_z P(w_n|z)p(z|d). \tag{1.6}$$

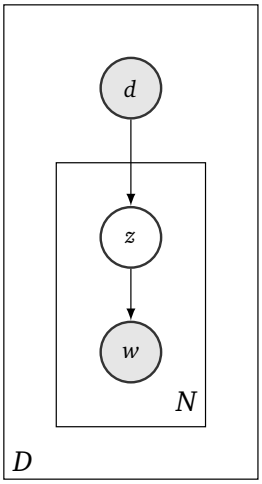


Figure 1.2: Graphical models for pLSI

The pLSI model endeavors to relax the simplifying assumption made in the mixture of unigrams model. In the mixture models, each document is generated only from one topic, where pLSI is able to assign multiple topics to a document as $P(z|d)$ contains a mixture of weights of the topics z for a particular document d. It is important to note that d is supposed to be chosen from the list of documents in the training set. Thus, d is a multinomial random variable with possible values of total number of training documents. Thus, the model can learn the topic proportions $P(z|d)$ only for the documents that are in the training set. Assuming this problem, pLSI is not a very well-suited generative

model for documents in topic modeling where there is no intrinsic way to use this model to assign probabilities to an unseen document. Another disadvantage of pLSI is is that the number of parameters which must be estimated grows linearly with the number of training documents which stems from applying a distribution indexed by training documents. The whole number of parameters that should be used in a k-topic pLSI model is $kV + kD$. These are k multinomial distributions of size V (unique words in vocabulary) and D (number of documents in the collection) mixtures over the k hidden topics. This results in a linear growth in D. As [Blei et al., 2003] illustrated in their results, the linear growth in parameters makes the model prone to overfitting and this problem prevents the topic model to estimate the content precisely. As a solution, a tempering heuristic has been used to smooth the parameters in the model for an acceptable accurate prediction. However, it has been shown, that overfitting can happen even when the tempering method is applied ([Popescul et al., 2001]).

LDA overcomes both the linear growth and unseen prediction problems. It uses the topic mixture weights as a k hidden random variables rather than a huge set of individual parameters that are linked explicitly to the training documents. Also, in LDA, each word in the observed or unseen documents is generated by a topic which randomly has been chosen from a distribution with a randomly chosen parameter. This parameter is also drawn from a smooth distribution once per document with a dimension k. Thus, the $k + kV$ parameters in LDA are not increased with D.

Latent Dirichlet Allocation (LDA, [Blei et al., 2003]) is a probabilistic Bayesian model used to describe a corpus of D documents, associated with a vocabulary of size V. LDA is based on the idea that documents in collection represented using random mixtures over hidden variables (topics) and each topic is identified by a distribution over words of the vocabulary associated with corpus. In the model illustrated in Figure 1.3, latent variables, indexed in $\{1, \cdots, K\}$, are used to represent the *hidden* (in the sense non-observed) topics underlying each document. It should be noted that referring to the latent multinomial variables for topics in LDA is for captureing text-oriented information, as [Blei et al., 2003] has mentioned there is no epistemological claim regards to these latent variables more than their benefits to represent the probability distributions over the words. The challenge for LDA is that the topics are not known previously and the goal would be learning them from the collection of words. Hidden variable models like LDA are structured distributions where observed data like words interact with hidden random variables like topics. In these models, the user puts a hidden structure over the observed data and then learns the structure using posterior inference. Hidden variable models are common in the machine learning domain; they can be Hidden Markov Models [Rabiner, 1990] or Kalman Filters [Kalman and Others, 1960] or Mixture Models [McLachlan and Peel, 2000]. In LDA, the observed data are the words from documents and the hidden variables show the latent topical format of each document. LDA is associated to the following generative model[1]:

- Generate, for each topic $k, 1 \leq k \leq K$, a distribution over the words: $\phi_k \sim Dir(\beta)$, where ϕ_k and β are V dimensional vectors;
- For each document d:
  - Choose a distribution over the topics: $\theta^d \sim Dir(\alpha)$, where θ^d and α are K dimensional vectors;
  - For each position (indexed by n, $1 \leq n \leq N$) in d: (a) Choose a topic assignment: $z_n^d \sim mult(1, \theta^d)$; (b) Choose the word w_n^d from the topic z_n^d with probability $P(w_n^d = v | z_n^d = k) = \phi_{k,v}$;

where N is the length of each document and $\phi_{k,v}$ is the v^{th} coordinate of ϕ_k. α and β correspond to the priors of the model.

There are assumptions that are made in LDA. First, the dimension of the number of latent

[1] For simplification and following standard practice, we do not model here the length of each document, assumed to be fixed and equal to N.

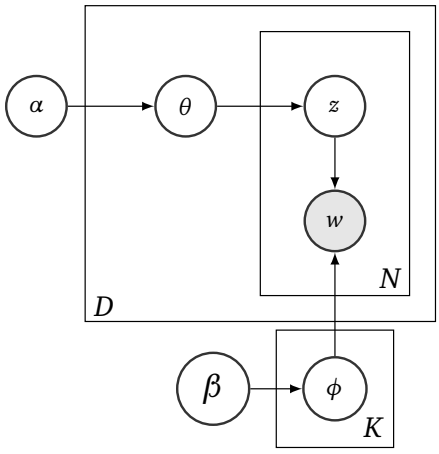


Figure 1.3: Graphical models for LDA

topics K, which is also the dimension of Dirichlet distribution over the topics, is assumed fixed and known. Second, the word probabilities ϕ is a $K \times V$ matrix that should be estimated after running the model. Third, α and β are usually fixed, following [Blei et al., 2003]. Furthermore, in almost all previous studies on LDA, the priors are considered to be symmetric, each coordinate of the vector being equal: $\alpha_1 = \cdots = \alpha_K$. If one assumes a broad Gamma prior for both α and β, then their value can be easily learned from data by *maximum a posteriori* [Asuncion et al., 2009] or *Markov Chain Monte Carlo* [Neal, 2003] methods. One can also envisage learning asymmetric Dirichlet priors [M. Wallach et al., 2009], which raises no particular difficulties for the models we are considering. For clarity sake, we however assume here fixed, symmetric priors; the extension to their learning through Gamma priors or through asymmetric priors is purely technical. In the remainder, we will denote by α and β the priors for the Dirichlet distributions as well the constant value taken by each coordinate of these priors, the context being sufficient to determine which element is referred to.

There is still a question that why this model deploys the Dirichlet Distribution. The Dirichlet Distribution is a convenient distribution over the fundamental elements. It is also positioned in the exponential family and relies on finite dimensional statistics. The most important characteristics of this distribution is conjugation with multinomial distribution which makes the model easy for the development of inference and parameter estimation. The K-dimensional Dirichlet distribution over θ given a vector of hyper-parameters α is as

follows:

$$P(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \dots \theta_K^{\alpha_K - 1} \tag{1.7}$$

where Γ stands for the Gamma function.

The generative process explained above has led to the following joint distribution:

$$P(w, z, \theta, \phi|\alpha, \beta) = P(\phi|\beta)P(\theta|\alpha)P(z|\theta)P(w|\phi_z) \tag{1.8}$$

The hyper-parameters α, β and the random variable ϕ are in the corpus level and are assumed to be drawn once during the generating a corpus. The random variables θ are in the document level and are drawn once per document. Finally, the hidden variables z_d^n and w_d^n are in the word level and are sampled once for each word of a document.

The prominent problem in topic models is the posterior inference. Posterior inference is reversing the derived generative process and learning the distributions of the latent variables and parameters in the model using the observed words. This inference for LDA is defined as follows:

$$P(z, \theta, \phi|w, \alpha, \beta) = \frac{P(w, z, \theta, \phi|\alpha, \beta)}{P(w|\alpha, \beta)} \tag{1.9}$$

The problem with the computation of $P(w|\alpha, \beta)$, makes this posterior intractable. However, there are a number of approximation techniques for the inference including Variational Bayes and Gibbs Sampling methods.

An important characteristic of LDA is that each document is generated independently from the previous ones. This is not a realistic assumption in different settings, as document streams and also an interesting objective in topic model can be to examine topic evolution and transitions, that in this case, LDA is not capable of capturing this evolution. Also in LDA, the word-order is not relevant and words are generated independently. This assumption called Exchangeability and has a direct influence on the LDA to facilitate the inference development. Nonetheless, this is not again a realistic assumption as we may miss important information with various orders. Also, words can be divided into different semantically coherent units such as Segments, Chunks, Sentences and Phrases that are not captured in LDA.

Regarding these two problems, we introduce our models respectively for the former in Chapters 3 and for the later in Chapter 4 and 5. These models are based on the integration

of Copula into `LDA` as a tool to capture dependencies between random variables. In the next section, we describe more about this tool and its features.

1.2 An introduction to copulas

The study of copula and its applications is a quite contemporary section in mathematics and specially statistics. Until recently, it was very difficult even to find the word of copula in the statistical articles. The very first referring of term copula in the Encyclopedia of Statistical Sciences is in year 1981 by [Schweizer and Wolff, 1981]. Although, in the first eighteen volumes of the indexes to statistics (1975-1992) there are only eleven papers mentioning copulas, however, there are 71 referring in the next ten volumes (1993-2002) which evidences the growth of interest in copulas and their applications to statistics and probability. Recently, there have been several venues devoted to or invoked somehow this concept, for example, the conference related to Distributions with Fixed Marginals, Doubly Stochastic Measures, and Markov Operators; the conference on Distributions with Given Marginals and Moment Problems; the conference on Distributions with Given Marginals and Statistical Modeling; the conference on Computational and Methodological Statistics. Then, there are conferences on the application of copulas into finance: conference on Dependence Modeling: Statistical Theory and Applications in Finance and Insurance; the conference on Statistics and Econometrics. As most of the titles indicate, copulas are mostly supposed to be part of study upon to marginal distributions.

To define a good description for this concept, from [Nelsen, 2007], copulas are functions that join or couple multivariate distribution functions to their one-dimensional marginal distribution functions. In other words, copulas can be seen as multivariate distribution functions whose one-dimensional margins are uniform on the interval $(0, 1)$. But what is important about copula is to know how it is of interest to statisticians and mathematicians. [Fisher, 1997] has responded to this question in his article: "Copulas [are] of interest to statisticians for two main reasons: Firstly, as a way of studying scale-free measures of dependence; and secondly, as a starting point for constructing families of bivariate distributions, sometimes with a view to simulation."

The term copula was first engaged in a mathematical and statistical view by [Sklar, 1959] in the theorem, now known as Sklar theorem, described the functions that "join together" one-dimensional distribution functions to form multivariate distribution functions. This word is a Latino term that means "a link, tie, bond" (Latin Dictionary of Cassell) and grammatically it can be used to explain "that part of a proposition which connects the subject and predicate" (Dictionary of Oxford English). At the moment that Sklar wrote his 1959 paper with the term "copula," he was working with Berthold Schweizer on the

development of the probabilistic metric spaces (PM) theory. During years 1958 to 1976, most of the results which were related to copulas were obtained in the terms of PM studies. To illustrate the relation between copulas and PM spaces , we assume a metric space consists of a set like S and a metric value like d that measures the distances between points of p and q in the set of S. In a probabilistic version of metric space, we can replace the distance metric $d(p,q)$ by a distribution function of F_{pq}. The value of $F_{pq}(x)$ for any real amount of x is the probability that the distance between points of p and q is less than x. The first difficulty in this structure happens when one attempts to estimate a probabilistic analog of the triangle inequality $d(p,r) \leq d(p,q)+d(q,r)$ which is the corresponding relationship between the distribution functions of F_{pr} , F_{pq} , and F_{qr} for all points like p, q, and r in set S. [Menger, 1942] has proposed an inequality of the $F_{pr}(x+y) \geq T(F_{pq}(x), F_{qr}(y))$; where T is a triangle norm or t-norm. Like a copula, t-norms map $[0,1]^2$ to $[0,1]$, and join distribution functions. Accordingly, some of t-norms are copulas and contrarily some of copulas are t-norms. So, as it makes sense, copulas had to proceed in PM spaces studies. One of the most important results in PM spaces was Archimedean t-norms, those t-norms that satisfy $T(u,u) < u$ for all u in $(0,1)$. Archimedean t-norms are also called Archimedean copulas. For some reasons, Archimedean copulas frequently have been applied in multivariate distributions applications like measuring dependencies. The reasons would be the simplicity of their forms, the convenience of constructing this family, and their properties. This is the main topic discussed in [Nelsen, 2007] and we are going to discuss more on this type of copula later, as we choose them as a solution for our problem.

We now focus more on copulas and dependency measurement. The earliest paper which explicitly showed the role of copulas in the study of dependency between random variables is titled "On nonparametric measures of dependence for random variables" by [Schweizer and Wolff, 1981]. In this paper, [Schweizer and Wolff, 1981] discussed the [Rényi, 1959] criteria and modified it to measure the dependency between pairs of random variables. They have expressed the basic invariance properties of copulas under strictly monotone transformations of random variables and introduced the metric of dependency measuring which is now known as Schweizer and Wolff's σ.

In conclusion, copulas are the tools for formalizing dependency structures of random variables. Although copulas have been known about forty years, they have been just recently more applied into sciences like biostatistics, biology, reliability, finance and etc. In finance, they have turned to be a standard tool with several applications like multiasset pricing, risk management, credit portfolio modeling and etc.

Although the concept of copulas is well defined, however, they are recognized as a very

difficult tool for empirical estimation. The problem with the estimation of copulas is that usually every marginal distribution of the fundamental random variables must be estimated and boosted into an estimated multivariate distribution. This procedure makes lots of unexpected effects regarding the usual statistical methods like noisy estimations, non-standard limiting behaviors etc.

Considering the property of copula to capture dependencies among random variables and the flexibility that it provides us in terms of learning the model parameters, we decided to leverage copula for solving the problems of LDA mentioned before. Copula can be accommodated into LDA to secure topic model regarding streams of documents and words dependencies within a document.

The outline of this work is as follows. In the next Chapter, we present the related works with respect to the limitations of LDA. In Chapter 3, we introduce efficient ways to capture topic dependencies when documents stream in topic models like LDA and infinite version of LDA(called iLDA) that topic model is supposed to estimate the number of topics as well. Consecutively, we expose the results obtained with our approaches on distinct datasets. We then describe in Chapter 4 the model that integrates text structure into LDA using copulas to relax the Exchangeability assumption of LDA and make use of words information in topic model. There is also the results achieved by this approach and the comparison with the other well-known methods. In Chapter 5, we position our joint latent model for topics and segments as a complete solution for compensating the independence assumption among words of a document in LDA. There is again the results applying this model to different datasets. In Chapter 6, we summarize this work regarding the methods that we developed and the results that we concluded, we will also describe the future plans for the new direction of investigation on LDA. Finally, the last Chapter is devoted to the mathematical computations for each model named Appendices.

Chapter 2

Related works

2.1 Streams of documents in topic models

Some studies have considered the possibility of modeling different streams of documents. Regarding the properties of the topic model (in terms of estimating the number of topics), streaming can be incorporated into the parametric topic models, such as LDA or non-parametric versions such as HDP.

2.1.1 Parametric topic models

In [Hong et al., 2011], authors tried to leverage standard models (as LDA) by considering topics common to the different streams. In this work, they first extended the standard topic models by integrating each text stream with both the local and share topic distributions, and then for the case of streams, they proposed to associate each topic with a time-dependent function that defines its popularity over time. By adding these two methods, they have tried to capture the dynamics of text streams in a united model. In this paper, they have also evaluated their model using a large dataset that includes text streams from Twitter and Yahoo News. In such studies, the evolution of topics over time is not considered.
The study presented in [Wang and McCallum, 2006], known as TOT, aims at modeling, through an extension of LDAwhere the timestamp associated with each token in a document. This topic model not only captures the low-dimensional structure of data, but also can show how the structure of data changes over the time. This work, unlike the others that commit on Markov assumptions, assumes topics are associated with a continuous

distribution over timestamps, and the mixture distribution of topics for each topic is effected by both the word co-occurrences and the timestamp of document. In this model, the occurrence and correlations of topics evolve significantly by time. The authors have presented their results using nine months personal email and 17 years of NIPS papers in research and 2 centuries of presidential state-of-the-union addresses. Nevertheless, if dependencies between topics are not explicitly modeled, topics tend to specialize over different time periods through the joint dependence of each word and timestamp on the topic variable (z in LDA).

Other studies have addressed the problem of topic evolution and dependencies within a single document, as the recent *sequential* LDA model described in [Du et al., 2010b]. This model aspires to uncover the underlying sequential structure. As an example, a document consists of multiple segments like chapters or paragraphs, each of them is correlated to its antecedent and the subsequent segments. In this model, this type of progressive sequential dependency is supposed to be captured by applying a hierarchical two-parameter Poisson Dirichlet process. The difference between this model and the previous one is, instead of modeling topic evolution in documents based on their timestamps, they model topic progress within each document by taking advantage of the correlations between its segments. They have shown that their model outperforms LDA in terms of perplexity metric over 1000 patent documents that are randomly selected from 8000 U.S patents. In the field of information theory, perplexity proposed by [Shannon, 1948] is a measurement to show how well a probability distribution or model can predict a sample. It can be used to compare different probability models. A lower perplexity for a probability model shows that this model is well trained to predict a sample.

We rather focus in this study on explicitly modeling topic dependencies across documents, for both topic and word-topic distributions. Several studies have addressed a similar problem. One of the first proposals corresponds to the Dynamic Topic Model (DTM), introduced in [Blei and Lafferty, 2006] and illustrated in Figure 3.1. This approach chains the natural parameters of each topic (called ϕ_k in LDA) in a state space model that changes with Gaussian noise. In this model, instead of using Dirichlet distribution for document-specific topic proportions, they have leveraged a logistic normal distribution with mean α (like α in LDA) to express the uncertainty over topic proportions. The sequential dependency between this new variable is again captured with a similar Gaussian distribution. They have also mapped the multinomial distributions to mean parameters for sampling topics and generating words of each document. A variational Bayes approximation based on kalman filters and non parametric wavelet regression over the hidden topics is deployed to

approximate the posterior inference for this model. An interesting feature of DTM is its use of time slices; we have not considered time slices in our study, but the models that we propose in Chapter 3 (as most dynamic models) can be extended to deal with them. They have analyzed this model by considering the task of predicting the next year of Science on the OCR archives of the journal science from years 1880 to 2000. They have also shown the perplexity results by comparing their method and the simple LDA. As it is mentioned before, DTM captures dependencies for both topic and word-topic distributions. These dependencies are however captured through Gaussian distributions, the expectation of which corresponds to the previous parameters. This entails that new parameter values are constrained to be distributed around the values observed previously. In contrast, even in our model ST−LDA−D (a direct extension of LDA using Dirichlet distribution that we propose in Chapter 3) the expectations of the new topic and word-topic distributions (Eqs. 3.2 and 3.4) can be uncorrelated to the previous distributions in the absence of dependencies. Our models will thus offer additional flexibility over the presence or absence of dependencies between consecutive documents in a stream.

The Dynamic Mixture Model (DMM, see Fig.3.1) introduced in [Wei et al., 2007] is similar to DTM except that topic dependencies are directly considered at the topic level (as similar as in our models but not for DTM which operates at the prior level) and that word-topic dependencies are dropped. In comparison with TOT model, DTM relies on discrete time stamps and defines dependencies between two consecutive documents as snapshots. Although TOT model captures both short-term and long-term topics evolution by having time stamps as an observed random variable, DMM is able to capture more details in terms of evolution. Also DMM is capable of modeling the dependency between any sequential shots, which is applicable to any streaming data. In comparison with DTM, DMM tracks the evolution between consecutive documents instead of between grouped slices of documents. It should be also noted that in both DTM and LDA, documents in a corpus and words within a document are completely exchangeable. In DMM, multiple time series which are related to documents, have very strong time order and exchanges of documents can result in a very different model. By this perspective, DMM can be recognized as a real online method for topic modeling. As for DTM, the expectation of a new topic distribution is given by the values obtained in the previous document but instead of Gaussian distribution, it enjoys Dirichlet distribution. This again contrasts with our proposal that introduces additional flexibility, as mentioned before. Results have shown that DMM has outperformed LDA using Chlorine dataset. This dataset is generated by EPA-NET that simulates the hydraulic and chemical phenomena in a drinking water distribution system and light intensity mea-

surements that gathered by Berkeley Mote sensors.

The Topic Tracking Model (TTM, see Fig.3.1) introduced in [Iwata et al., 2009] is similar to our models in the sense that both topic and word-topic (more precisely interest-topic) dependencies are considered. However, as for DTM and DMM, the means of the current topics and interests are the same as the ones of the previous topics and interests. The model is thus again limited in its ability to model the presence or absence of dependencies between consecutive documents. This model showed better results than LDA and an online version of LDA using two real purchase log datasets for movie and cartoon downloading service.

A more recent proposal, called Temporal LDA (TM−LDA, see Fig.3.1), was introduced in [Wang et al., 2012]. TM−LDA attempts to learn the transition parameters among topics by minimizing the prediction error on topic distributions for sequential documents. By training TM−LDA, this model is capable of predicting the expected topic distribution for the future document. For being more accurate in terms of predictions in a realistic online setting, they have developed an updating algorithm to adjust transition parameters when a new document streams in. They have presented their results over a corpus of 30 million Tweets, showed that TM−LDA can outperform the simple version of LDA model for estimating the topic distribution of a new document. TM−LDA differs from the previous models as it also aims at predicting future topics even in the situation where future documents are not seen. It thus assumes a strong dependency between consecutive documents, which is not always realistic, even on such collections as Tweets. Furthermore, TM−LDA does not consider dependencies for the word-topic distributions.

2.1.2 Non parametric topic models

The Hierarchical Dirichlet Process (HDP, [Teh et al., 2006]) is a Bayesian non-parametric model that can be used to model collection of documents with a possibly infinite number of topics as components. It has been widely used in probabilistic topic models, where by giving a collection of documents to model, a posterior inference can estimate the number of topics that potentially needed and describe their distributions. One drawback of HDP model is that standard posterior inference algorithm that defined for it, needs to pass multiple times through all the dataset which makes it intractable for many large-scale datasets. In [Wang et al., 2011], they proposed an online variational inference algorithm for the HDP that is easily applicable for massive data. Their model is much faster than the traditional inference and lets the user analyze larger datasets. They applied coordinate-ascent variational Bayes

without numerical approximation as an inference into the stick-breaking representation of HDP model. Their method was inspired by the online variational bayes algorithm which was proposed by [Hoffman et al., 2010] for LDA. The idea behind this model is to optimize the variational objective function using stochastic optimization. In this model, optimization is carried out by constantly taking a random subdivision of data, and updating the variational parameters regards to them. They have used a log-likelihood metric for evaluating two datasets that consist of Nature (the articles from years 1869 to 2008) and PNAS (the Proceedings of the National Academy of Sciences from years 1914 to 2004). They have finally concluded that this model outperforms the online extension of LDA. Although, this algorithm is applicable to large-scale streaming data, the authors didn't really integrate the streaming assumption as timestamps into the model.

[Wang et al., 2008] have developed the continuous time dynamic topic model (cTDM). As an extension of DTM, it is based on a dynamic topic model that uses Brownian motion to model the latent topics (only ϕ_k) through a sequential collection of documents. They assume each topic as a pattern of the word that evolves over the course of the collection. A limitation of DTM is that the time is discretized into many periods. In DTM, if the resolution is chosen roughly, then the assumption that a group of documents within a time slot is exchangeable will not be a correct one. If the resolution is chosen too fine, then the variational parameters will grow when more timestamps added to the collection. Acknowledging this limitation, the discretization's resolution should be based on the features of data and the computational complexity for the topic model. cTDM, in contrast with DTM, is a model based on continues sequential time-series with arbitrary granularity. In this way, cTDM can be assumed as a normal limit of DTM with the finest resolution. They have shown their results for per-word perplexity and timestamps prediction on two different datasets. The first one is AP collection which is a subset of the TREC AP corpus consists of the news from 05/01/1988 to 06/30/1988 and they are time-stamped by hour. The second one is the *Election 2008* that are the top articles from Digg.com about the 2008 presidential election.

As another extension of DTM, [Ahmed and Xing, 2010] introduced an infinite dynamic topic model (iDTM). In this paper, the authors have considered that documents in the collection are organized into epochs and documents within each epoch are exchangeable in terms of order. Also, the order between the documents is still kept over epochs. In this work, they have accommodated the evolution of document-specific topic and topic-words distributions into normal Chinese Restaurant Franchise (CRF) representation. In iDTM, an infinite number of topics can be activated and deactivated at any epoch, the topic-words

distributions evolve according to a first-order state space model, and the document-specific topics distribution evolve using the idea that rich gets richer with a Δ-order process. The iDTM constructed over the recurrent Chinese Restaurant Franchise (RCRF) process which captures dependencies between the topics and popularity of each epoch. RCRF is also constructed on top of RCRP (Recurrent Chinese Restaurant Process) which is introduced in [Ahmed and Xing, 2008]. For iDTM, an efficient Gibbs sampling inference has been developed. It relies on a dynamic way of maintenance of sufficient statistics to make the sampler faster. The iDTM has been evaluated for the birth and evolution of topics on the NIPS collection[1] and showed better performance than HDP and DTM models in the small number of topics (less than 60 topics). However, results have shown that if the number of topics increases, there is an improvement for the performance of DTM and it may outperform iDTM.

Furthermore, a simple non-parametric dynamic topic model is mentioned as an example for Temporal Dirichlet Mixtures model (TDPM) that they have introduced in [Ahmed and Xing, 2008]. In their framework, they applied the same technique of collection dividing into the epochs and exchangeability within each epoch for documents, they also used a recurrent process in the model by adding the effect of previous document's topic assignment into the current document's topics assignment However, in this model, each document is generated from a single topic instead of a mixture of topics due to inference difficulty of the model but it is still a big assumption for a topic model.

The Dynamic Hierarchical Dirichlet Process (dHDP) [Ren et al., 2008] is one of the direct extensions of HDP where document streams. The authors of this model have applied a bayesian dynamic structure from [Dunson, 2006] to extend HDP and integrate time dependence. They incorporate a linear mixture of weighted topic distribution of the previous document and shared topic distribution within collection for estimating current document's topic distribution. They have used a modified version of block Gibbs sampler proposed in [Ishwaran and James, 2001] for dHDP inference. Nevertheless, they have not evaluated their method for topic modeling tasks nor compared with the other state of art topic models. They analyzed their method in the case of music segmentation to infer relationships between various parts of a sample music, and also time-evolving features of the gene-expression collection.

In the same direction, [Wang et al., 2017] have introduced Evolving Dirichlet Processes (EDP) and Evolving Hierarchical Dirichlet Processes (EHDP) models to track nonlinear

[1] It is available at http://www.datalab.uci.edu/author-topic/NIPs.htm

evolutionary in temporal data. They have used a combination of Dirichlet processes (accordingly for EHDP, Dirichlet base distributions) of previous document and the current one to conclude topic distribution of the current document, they have also applied the same trick to capture the topic-word distributions dependency between consecutive documents. These models are built on the top of the Chinese Restaurant Process representation and a Gibbs sampling method has been developed for them as posterior inference. They have evaluated their methods using 4 different real-world datasets consists of NIPS articles, DBLP abstract of articles, NSF awards[2] and Douban comments about the movies[3], and especially a synthetic dataset to show whether their method can correctly follow the evolutionary evidence of temporal data. They have shown that EHDP can outperform the methods like DTM, HDP, RCRF and TOT in terms of perplexity. This fact has been the reason why we use this method for comparing with our non parametric streaming model (CopHDP which will be proposed in Chapter 3).

[2]It is available in https://victorfang.wordpress.com/2011/09/01/860/

[3]http://movie.douban.com/

2.2 Word dependencies in topic models

Despite the success that vector-space models [Salton et al., 1975] have enjoyed, they come with a number of limitations. We mention, for instance, their inability to model synonymy and polysemy and the sparse, high-dimensional induced representations. Many research studies have researched these problems, and Probabilistic Latent Semantic Indexing [Hofmann, 1999] was among the first attempts to model textual corpora using latent topics. pLSI was the first probabilistic model that explained the generation of co-occurrence data using latent random topics and, the EM algorithm for parameter estimation. The model was found more flexible and scalable than the Latent Semantic Analysis [Deerwester et al., 1990], which is based on the singular value decomposition of the document-term matrix, however, pLSI is not a generative model as parameter estimation should be performed with each addition of new documents. To overcome this drawback, [Blei et al., 2003] have proposed the Latent Dirichlet Allocation (LDA) by assuming that the latent topics are random variables sampled from a Dirichlet distribution and that the generated words, occurring in a document, are exchangeable. In this context, the corpus is associated with a set of latent topics, and each document is associated with a random mixture of those topics. The words are assumed exchangeable, that is their joint probability is invariant to their permutation. Previous works have proposed a variety of extensions to LDA in order to incorporate additional information such as class labels [Mcauliffe and Blei, 2008] and temporal dependencies between stream documents [Wang et al., 2012]. The interdependence assumption allows the parameter estimation and the inference of the LDA model to be carried out efficiently, it is not realistic in the sense that topics assigned to similar words of a text span are generally incoherent.

Different studies, presented in the following sections, attempted to remedy this problem and they can be grouped into two broad families depending on whether they make use of external knowledge-based tools or not in order to exhibit text structure for word-topic assignment.

2.2.1 Knowledge-based topic assignments

The main assumption behind these models is that text-spans such as sentences, phrases or segments are related in their content. Therefore, the integration of these dependent structures can help to discover coherent latent topics for words. Different attempts to combine LDA-based models with statistical tools to discover document structures have

been successfully proposed. In [Griffiths et al., 2005], the authors have investigated a combination of a topic model with a Hidden Markov Model (HMM). They have assumed that the HMM generates the words that handle the long range dependencies (semantic dependencies) and the topic model that generates the words that handle the short range dependencies (syntactic dependencies). Syntactic conditions that bring in short range dependencies, cover many words but not going further than the boundary of a sentence. Semantic conditions that bring in long range dependencies, make various sentences within a document are more likely to have identical content, and consequently, have similar words. In this paper, they have proposed an algorithm that captures the interacts between the short and long range dependencies, base on a generative model where a HMM model decides when a word can be emitted from a topic model. The different abilities of the two elements of this model lead to factorizing a sentence into function words as syntactic classes which controlled by HMM, and content words as semantic classes which controlled by the topic model. They have evaluated their model in different quantitative tasks, like document classification and part-of-speech tagging and concluded better results than simple HMM and LDA.

[Boyd-Graber and Blei, 2009] have proposed the Syntactic topic model whose goal is to integrate the text semantics and the syntax in a non-parametric topic model. In contrast with the previous model that generate the words either from the syntactic or semantic context, this syntactic topic model generates the words that are constrained to be dependent to the both. In this work, they attempt to model a document in a collection as an exchangeable sets of sentences, each of which should be associated with a tree structure like a parse tree. The words within a sentence are supposed to be sampled from a distribution that affected by both of their observed role in mentioned tree and the latent dominant topics in the document. Having the tree, the semantic consistency of each document is given by a distribution over latent topics, as in topic models, and the syntactic consistency by the fact that each element in the tree has also a distribution over the topics of its children. They have used perplexity metric to compare their model with HDP and obtained better results over a Penn Treebank [Marcus et al., 1993] corpus dataset.

In another effort, [Zhu et al., 2006] have proposed TagLDA, where they replace the unigram word distributions by a factored representation that is conditioned on the topic and the part-of-speech tag of a term. In this model, they have assumed a group of tags are Known and pre-defined, they have also assumed that each word in the collection has its own tag given. By this way, tags construct the domain knowledge. In this paper, topics and tags are assumed orthogonal to each other and the same topic can have different

word distribution under different tags. A variational inference has been developed for this model, and it has been analyzed by a group of synthesized and real-world datasets consist of AP news articles, WebKB corpus, and the NIPS one. In the experiments part, TagLDA showed a better result than LDA in terms of perplexity, but there is no significant improvement when TagLDA is applied for the classification task.

Recently, [Balikas et al., 2016a] have introduced senLDA, that assumes that the terms occurring within a sentence are generated by the same topic. They have claimed that the latent topics of short text spans like sentences should be consistent across the words of those spans. In this method, these text spans can include the paragraphs or sentences or even phrases. They have Also showed that in the extreme case of this model where words are the coherent units of text segments, LDA becomes a special case of this approach. senLDA and LDA differ in the case that LDA assumes complete independence between the words of a document in general where senLDA assumes a very strong dependence between the topics assigned to the words of sentences. In the experiments, they have obtained better results than LDA in terms of classification. LDA has shown better performance in terms of perplexity, while senLDA has been still faster convergence in comparison to LDA. In a part of our study in Chapter 4, we integrate part of the text structure in LDA by relying only on the boundaries of contiguous text spans like sentences, which can be obtained without deep linguistic analysis like the one required in the Syntactic Topic Model. Also, differently from senLDA, we do not restrict the words of the spans to be generated by the same topic. Instead, using copulas we pose correlations between those topics, which is more flexible. In this model, contrary to identifying such spans like segments, we assume them to be topically coherent *a priori*, and we investigate how to leverage and incorporate this information to LDA.

In the same line, [Du et al., 2013] following [Du et al., 2010a] have presented a hierarchical Bayesian model for unsupervised topic segmentation. This model integrates a boundary sampling method used in a Bayesian segmentation model introduced by [Purver et al., 2006] to the topic model. For inference, a non-parametric Markov Chain inference is used that splits and merges the segments while a Pitman-Yor process [Teh, 2006] binds the topics. Although, this model has a novel way of binding segmentation with topic models, it is only applied into segmentation tasks and has not been compared with the other stat-of-arts topic models. The authors have used Choi's dataset [Choi, 2000] which is commonly used for topic segmentation evaluation. They have also utilized two annotated meeting transcripts [Kazantseva and Szpakowicz, 2011, Eisenstein and Barzilay, 2008] to show the ability of this method to outperform other models such as Bayesian

segmentation [Purver et al., 2006] and Segmented Topic Model [Du et al., 2010a]. For the evaluation propose, they used P_k (introduced by [Beeferman et al., 1999]) and WindowDiff (WD, introduced by [Pevzner and Hearst, 2002]) which are two widespread metrics used in topic segmentation.
Recently, [Tamura and Sumita, 2016] have extended this idea to the bilingual setting. They have assumed that documents consist of segments and the topic distribution of each segment is generated using a Pitman-Yor process. They have built their model on top of Bilingual Latent Dirichlet Allocation model (`BiLDA`) [Mimno et al., 2009] which considers only cross-lingual alignments between the whole documents, and proposed to also considers the cross-lingual alignments between segments in addition and assigns the same topic distribution to the aligned segments. They have incorporated unsupervised topic segmentation method [Du et al., 2013] mentioned before into this model. Experimental results of this paper have shown that the proposed model outperforms `BiLDA` in terms of perplexity and illustrated an improvement for the translation pair extraction task.

Though, the topic assignments follow the structure of the text; these models suffer from the bias of statistical or linguistic tools they rely on. To overcome this limitation, other systems integrate automatically the extraction of text structure, in the form of phrases, in their process.

2.2.2 Knowledge-free topic assignments

This type of models extracts text-spans using n-gram counts and word collections and use bigrams to integrate the order of words as well as to capture the topical content of a phrase [Lau et al., 2013]. In [Shafiei and Milios, 2006], the authors have proposed a four-level hierarchical structure where the latent topics of paragraphs are decided after performing a nested word-based `LDA` operation. This work contains a four-level Bayesian model, in which each document is a random mixture of document topics, and each topic is a distribution over some segments, then each of these segments within a document can be a mixture of word-topics where each topic is a distribution over words. They have also presented an efficient inference based on a combination of Markov Chain Monte Carlo method and Moment-Matching algorithm. They have reported their results for tasks such as document modeling, document and term clustering and showed a better outcome than `LDA` using two real-world datasets, NIPS collection mentioned previously and Wikipedia XML collection[4].

[4]It is available at `http://www-connex.lip6.fr/~denoyer/wikipediaXML`

[Wang et al., 2007] have studied how the word order in the form of n-grams can be leveraged to better capture a document's topical content. Their topical n-gram model extends LDA by determining unigram words and phrases based on context and assigning mixture of topics to both individual words and n-gram phrases. This model generates words with their textual order where for each word, model first samples a topic then samples its status as a unigram or bigram, and then samples the word from a topic-specific unigram or bigram distribution. As an example, this model can capture *white house* as a special phrase in the *politics* category and not in the *real estate* category. The authors have showed an improvement in the retrieval performance and topic assignment in the experiments run over NIPS and TREC collections.

Further, [Wang et al., 2009] have merged topic models with a unigram model over sentences that assigns topics to the sentences instead of the words. In this paper, they have proposed a new Bayesian topic model for summarization by using both the term-document and term-sentence associations, they also explicitly modeled the probability distributions of selected sentences given over topics and made a prominent way for the summarization task. To evaluate this model, they have presented results using the DUC2002 and DUC2004 datasets, which are the benchmark datasets from Document Understanding Conference for generic automatic summarization. They have shown better performance than models such as non-negative matrix factorization (NMF) or Latent Semantic Analysis (LSA).

The approach that we propose in Chapter 5 also does not make use of external statistical tools to find text segments. The main difference with the previous knowledge-free topic model approaches is that the proposed approach assigns topics to words based on two, segment-specific and document-specific distributions selected from a Bernoulli law. Topics within segments are then constrained using copulas that bind their distributions. In this way, segmentation is embedded in the model and it naturally comes along with the topic assignment.

2.3 Copula applications

Lately, there is an increasing interest in the integration of copulas in machine learning applications. Gal Elidan in [Elidan, 2013] has argued the context of information estimation and multivariate modeling, the strengths and flaws of machine learning domain and showed how copulas offer opportunities for cooperative constructions. This work proposes several structures in machine learning which are based on copula such as multivariate copula-based construction, tree-structured copulas, Bayesian mixtures of copula trees and finally Copula Bayesian Networks (CBN). Network-based classifiers like naive Bayes models are appealing since they are easy to interpret and quite effective most of the time. They can also naturally manage the missing data and some other problems in classification. But for complex datasets with continuous interpretive variables, they have a sub-optimal performance. To overcome this issue, [Elidan, 2012] has presented a Copula Network Classifiers (CNCs) that combine the flexibility of a graph-based construction with the modeling ability of copulas. He has shown that CNCs offer better performance than linear and nonlinear generative models, and also discriminative models such as Radial Basis Functions(RBF, [Powell, 1987]) or Suppor Vector Machines (SVM, [Cortes and Vapnik, 1995]) with polynomial kernel.

[Liu et al., 2009] have introduced a nonparanormal model which is a type of Gaussian copula with nonparametric marginals that is applicable for estimating high dimensional graphs. The nonparanormal model can be assumed as a sparse additive extension for the setting of graphical models. This paper has presented an estimator for the component functions that is built on the tails of the empirical distribution function with relevant levels. Experimentally, the authors showed that fitting a high dimensional nonparanormal model is not computationally more difficult than estimating a multivariate Gaussian model.

Interestingly in the same direction, [Wilson and Ghahramani, 2010] have shown how to incorporate copulas in Gaussian processes in order to model the dependency between random variables with arbitrary marginals with a practical application on predicting the standard deviation of variables in the financial sector (volatility estimation).

In another generic framework, [Tran et al., 2015] have shown the benefits of using copulas to model complex dependencies between latent variables in the general variational inference setting. In this thesis, we present the idea of integrating copulas into topic models which is recently presented in our articles [Amoualian et al., 2016, Balikas et al., 2016b, Amoualian et al., 2017] partially.

Chapter 3

Copula-based parametric and non-parametric LDA models for document streams

The recent proliferation of temporal textual data on the Internet such as Tweets or comments on Youtube has brought new challenges for learning with interdependent data. Though important progress has been made in some directions [Gaber et al., 2005], popular approaches for most of these tasks are designed to deal with static collections of documents. This is specially the case for latent topic modeling, albeit analyzes of social content have gained much attention in recent years for different aspects of daily life, such as latent health-related topic analysis [Paul and Dredze, 2011] or buzz detection [Sakaki et al., 2010].

Although the main goal of probabilistic modeling is to find word topics, an equally interesting objective is to examine topic evolutions and transitions. In this chapter, we propose three extensions of `LDA` for modelling the dependency between two consecutive documents in a stream and examine their topic evolutions and transitions. The seminal work of [Blei and Lafferty, 2006] proposed to model the dynamic evolution of topics by first grouping documents into time slices and then by chaining the evolution of both the word-topic and topic mixture distributions via a Gaussian process. In some cases, the Gaussian distribution was not found to be the appropriate distribution in modelling the topic shifts and some studies considered other probability distributions for capturing the evolution of topics over time, e.g. [Wang and McCallum, 2006]. However, the idea of

grouping documents into epochs for modelling topic evolution was echoed in a number of studies. For example, [Wang et al., 2012] estimated a transition matrix over topic vectors between two predefined epochs and they showed that the LDA model [Blei et al., 2003] can be enhanced by considering directly the evolution of the topics over time.

In this study, we propose three models to capture topic and word-topic dependencies in document streams. In the first model, we suppose that the dependency between topic distributions of two consecutive documents follows a Dirichlet distribution controlled by an hyperparameter. This model is similar to the one of [Blei and Lafferty, 2006] with time slices equal to 1, but it offers a more precise mechanism for controlling the dependencies and is based on a framework encompassing all the situations (from complete independence to plain equality). This first study paves the way for a more general topic model in which the dependencies between the topics of two consecutive documents are captured by copulas which constitute generic tools to model dependencies between random variables [Derrode and Pieczynski, 2013]. Among the several families of copulas that have been defined in the literature, our choice fell on Archimedean copulas [McNeil, 2008, McNeil and Nešlehovà, 2009] as they are symmetric and associative, necessary conditions when dealing with exchangeable random variables [Ostap et al., 2013]. More particularly, we use Franck copulas, a special case of Archimedean copulas that rely on a single parameter, easier to estimate and more robust to sparse data. Lastly, the third model is a non-parametric extension of the second one through the integration of copulas in the stick- breaking construction of Hierarchical Dirichlet Processes.

This study is an extension of the one we presented in [Amoualian et al., 2016] in which the parametric models, already proposed in [Amoualian et al., 2016], are further detailed and in which a new, non-parametric version of the copula-based model is proposed. In addition, the experiments have been extended to cover new datasets, as well as new results, so as to better illustrate the behaviour of the proposed models.

Using five collections with different characteristics, we show that our approaches are faster and outperform state-of-the-art topic models both in terms of perplexity and for tracking similar topics in document streams.

The outline of this chapter is as follows: In the next section, we present the first model, a direct extension of LDA to capture topic dependency. Section 3.2 includes a copula-based extension of LDA to track the dependency when documents stream. Section 3.3 presents a Non parametric version of copula-based approach uses stick-breaking to represent the infinite extension of LDA. In Section 3.4, we introduce an efficient procedure to estimate the most important, in terms of size, parameters. We then describe in Section 3.5 the results

obtained with our approaches on five distinct datasets. Finally, Section 3.6 concludes our chapter by summarizing its main results and by giving some pointers to future research.

3.1 Dirichlet-based dependencies for LDA

We introduce here a first extension of LDA, that we refer to as ST−LDA−D.

3.1.1 Presentation of ST−LDA−D model

In this first model, we rely on a direct extension of the LDA model to take into account dependencies between the document-specific topic distributions of two sequential documents, denoted $(d-1)$ and d $(2 \leq d \leq D)$. This extension uses, as the standard LDA model, Dirichlet distributions for the document-specific topic distributions, the parameters of which are linear combination of the standard prior α and the topic distribution estimated in the previous document:

$$\theta^d | \theta^{d-1} \sim Dir(\alpha + \lambda_d \theta^{d-1}) \tag{3.1}$$

where λ_d is a uniformly distributed parameter that controls the influence of the topics of document $(d-1)$ on the topics of document d (see Figure 3.1). The expectation of each component of θ^d is given by:

$$\mathbb{E}[\theta_i^d | \theta_i^{d-1}] = \frac{\alpha + \lambda_d \theta_i^{d-1}}{K\alpha + \lambda_d} \tag{3.2}$$

Hence, if λ_d is high, i.e. if document d covers the same topics as document $(d-1)$, then $\mathbb{E}[\theta_i^d | \theta_i^{d-1}] \approx \theta_i^{d-1}$.

We furthermore assume that the previous document, $(d-1)$, can influence the word-topic distributions of the current document d. This assumption, also made in dynamic topic models [Blei and Lafferty, 2006] and topic tracking models [Iwata et al., 2009], is motivated by the fact that, within a given topic, if word distributions evolve over time, they tend to do so in a smooth way. As before, one can use a direct extension of the LDA model to account for dependencies between word-topic distributions in sequential documents:

$$\forall k, 1 \leq k \leq K, \phi_k^d | \phi_k^{d-1} \sim Dir(\beta + \mu_d \phi_k^{d-1}) \tag{3.3}$$

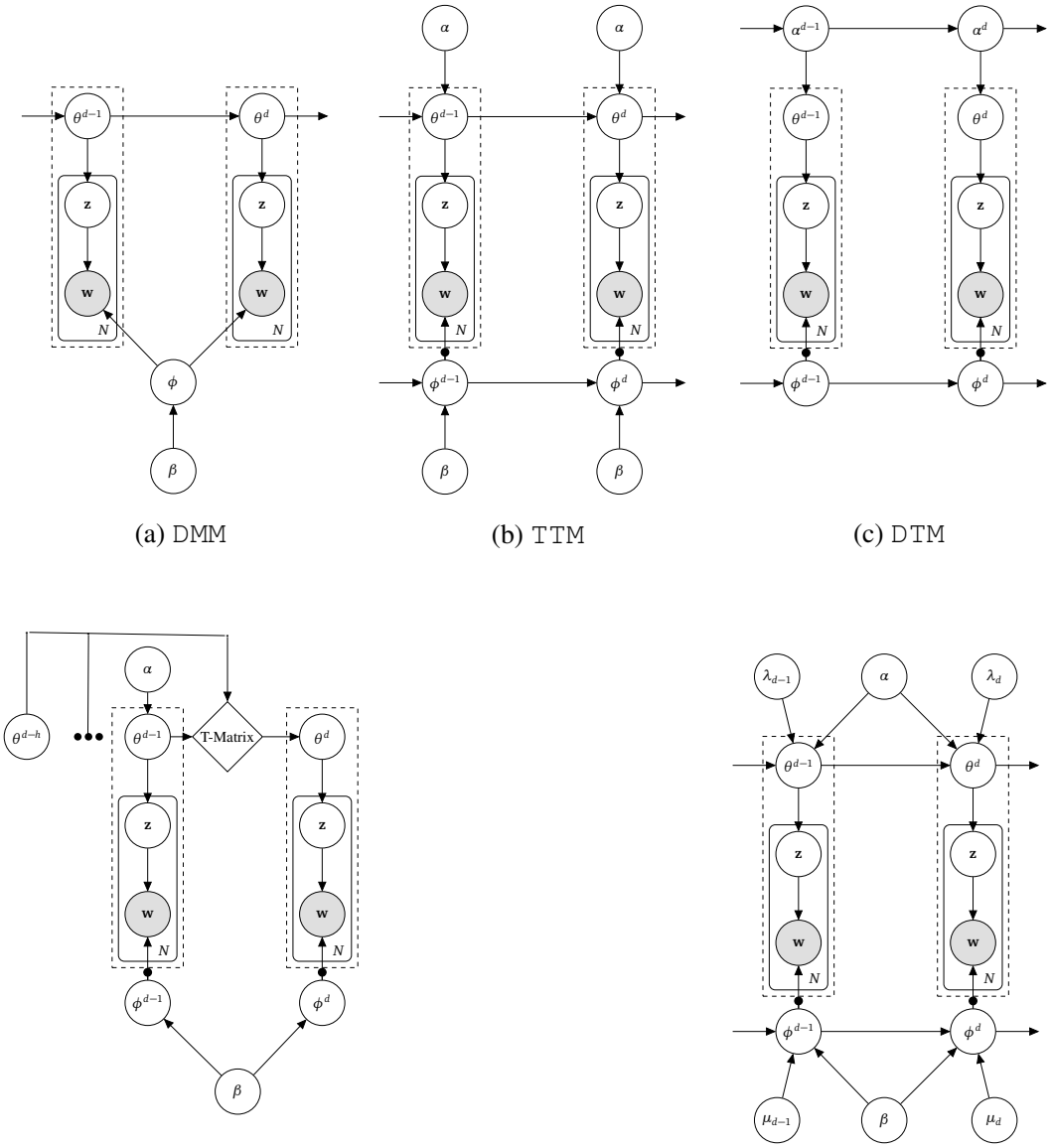
Here μ_d is again a uniformly distributed parameter that controls the tradeoff between the prior β and the learned topic-word distributions ϕ^{d-1}. As usual ϕ_k^{d-1} is the word

distribution of topic k. The conditional mean of each component of ϕ_k^d is given by:

$$\mathbb{E}[\phi_k^d | \phi^{d-1}] = \frac{\beta + \mu_d \phi_k^{d-1}}{V\beta + \mu_d} \tag{3.4}$$

and is approximately the value of the same component of document $(d-1)$ when the two documents are strongly dependent.

Lastly, as one can note, by setting $\lambda_d = \mu_d = 0, \forall d, 2 \leq d \leq D$, one "forgets" the dependencies between consecutive documents. The streaming model is in this case identical to the standard LDA model.



(a) DMM (b) TTM (c) DTM

(d) TM−LDA (e) ST−LDA−[D|C]

Figure 3.1: Graphical models for Dynamic Mixture Models (DMM, [Wei et al., 2007]), Topic Tracking Models (TTM, [Iwata et al., 2009]), Dynamic Topic Models (DTM, [Blei and Lafferty, 2006]), Temporal LDA (TM−LDA, [Wang et al., 2012]) and Streaming-LDA (ST−LDA−[D|C])

3.1.2 Inference with gibbs sampling for ST−LDA−D

As mentioned before, the parameters α and β are considered fixed. The other parameters can be estimated through Gibbs sampling, with Metropolis-Hasting updates for the

parameters λ_d and β_d. We give here the update formulas of each parameter.

For θ, one has:

$$\begin{aligned}
\theta^d &\sim P(\theta^d|\theta^{d-1}, z^d, w^d, \alpha, \beta, \lambda_d, \phi^{d-1}, \phi^d, \mu_d) \\
&= \frac{P(\theta^d, \theta^{d-1}, z^d, w^d, \alpha, \beta, \lambda_d, \phi^{d-1}, \phi^d, \mu_d)}{P(\theta^{d-1}, z^d, w^d, \alpha, \beta, \lambda_d, \phi^{d-1}, \phi^d, \mu_d)} \\
&= \frac{P(z^d|\theta^d)P(\theta^d|\theta^{d-1}, \alpha, \lambda_d)}{P(z^d|\alpha)} \\
&= \frac{(\prod_{n=1}^{N} \theta^d_{z^d_n}) Dir(\alpha + \lambda_d \theta^{d-1})}{\frac{B(\Omega_d+\alpha)}{B(\alpha)}} \\
&= \frac{B(\alpha)B(\alpha + \lambda_d\theta^{d-1} + \Omega_d)}{B(\alpha + \Omega_d)B(\alpha + \lambda_d\theta^{d-1})} \times \\
&\quad Dir(\Omega_d + \alpha + \lambda_d\theta^{d-1})
\end{aligned} \tag{3.5}$$

where Ω_d is defined as in [Wang, 2008] and represents the d^{th} row of the $D \times K$ count matrix Ω, with $\Omega_{d,k}$ being the number of times that topic k is assigned to words in document d.

The update for ϕ^d_k, $1 \leq k \leq K$ is similar:

$$\begin{aligned}
\phi^d_k &\sim P(\phi^d_k|\theta^{d-1}, \theta^d, z^d, w^d, \alpha, \beta, \lambda_d, \phi^{d-1}_k, \mu_d) \\
&= \frac{P(\phi^d_k, \phi^{d-1}_k, \theta^{d-1}, \theta^d, z^d, w^d, \alpha, \beta, \lambda_d, \mu_d)}{P(\phi^{d-1}_k, \theta^{d-1}, \theta^d, z^d, w^d, \alpha, \beta, \lambda_d, \mu_d)} \\
&= \frac{P(w^d|z^d, \phi^d_k)P(\phi^d_k|\phi^{d-1}_k, \beta, \mu_d)}{P(w^d|z^d, \beta)} \\
&= \frac{B(\beta)B(\beta + \mu_d\phi^{d-1}_k + \Psi_k)}{B(\beta + \Psi_k)B(\beta + \mu_d\phi^{d-1}_k)} \times \\
&\quad Dir(\Psi_k + \beta + \mu_d\phi^{d-1}_k)
\end{aligned} \tag{3.6}$$

where Ψ_k is again defined as in [Wang, 2008] and represents the k^{th} row of a $K \times V$ count matrix, $\Psi_{k,v}$ being the number of times that topic k is assigned to word v in the documents seen so far.

The Gibbs update for z is the same as the one for the standard `LDA` model:

$$\forall k, 1 \leq k \leq K, P(z_v^d = k|\theta^d, \phi^d) = \frac{P(z_v^d = k|\theta^d) \times P(w_n^d = v|z_v^d = k, \phi^d)}{\sum_j \left(P(z_v^d = j|\theta^d) \times P(w_n^d = v|z_v^d = j, \phi^d)\right)} = \frac{\theta_k^d \times \phi_{k,v}^d}{\sum_j \theta_j^d \times \phi_{j,v}^d} \tag{3.7}$$

Finally, for λ_d and μ_d, one can not directly compute Gibbs updates as the normalizing factor for the distribution of λ given all the other parameters can not be computed exactly. One can nevertheless rely on a Metropolis-Hasting procedure, detailed in Appendix A.1.

3.2 Copula-based dependencies for LDA

Model ST−LDA−D captures topic and word-topic dependencies through Dirichlet distributions, which allow one to balance the influence of the priors (α and β) and of the topic and topic-word distributions of the previous document. We introduce now another extension of LDA in which the dependencies between the topics of consecutive documents are modeled through copulas, which constitute a generic tool to model dependencies and do not rely on a specific distribution. We first provide a brief overview of copulas, prior to describe our model.

3.2.1 Basics on copulas

For every $p \geq 2$, a p–dimensional copula is a p–variate density function on $[0,1]^p$, whose univariate marginals are uniformly distributed on $[0,1]$. Copulas are particularly useful when modeling dependencies between random variables. Indeed, the joint cumulative distribution function (CDF) $F_{X_1,\cdots,X_p}$ of any random vector $\mathbf{X} = (X_1, \cdots, X_p)$ can be written as a function of its marginals, as follows:

Theorem 3.1 *(**Sklar's theorem** Theorem 2.3.3 of [Nelsen, 2007]) Let $F_{X_1,\cdots,X_p}$ be a p–dimensional distribution function with marginals $F_{X_1}, \cdots, F_{X_p}$. Then there exists a copula C with uniform marginals such that:*

$$F_{X_1,\cdots,X_p}(x_1, \cdots, x_p) = C(F_{X_1}(x_1), \cdots, F_{X_p}(x_p))$$

Furthermore, when the CDF $F_{X_1,\cdots,X_p}$ is continuous, the copula is unique.

Copulas represent a general way of modeling the dependencies between random variables, from complete independence to equality. If the random variables $X_1, \cdots, X_p$ are pairwise independent, their copula is the so–called *independency copula*:

$$F_{X_1,\cdots,X_p}(x_1, \cdots, x_p) = F_{X_1}(x_1) \cdots F_{X_p}(x_p)$$

whereas in the case $X_1 = \cdots = X_d$, one gets the *comonotonicity copula*:

$$F_{X_1,\cdots,X_p}(x_1, \cdots, x_p) = \min_{i \in \{1,\cdots,p\}} F_{X_i}(x_i)$$

Several copula families have been defined in the literature, among which the Archimedean copulas ([Nelsen, 2007, Ch. 4]), particularly interesting in our case. A p–dimensional Archimedean copula C with generator ψ is defined as:

$$C_p(u;\psi) := \psi(\psi^{-1}(u_1) + \cdots + \psi^{-1}(u_p)), u \in [0,1]^p$$

where ψ is a continuous, decreasing function, from $[0,\infty]$ to $(0,1)$, strictly decreasing on $[0, \inf\{t : \psi(t) = 0\}]$, and satisfying:

$$\psi(0) = 1,\ \psi(\infty) = \lim_{t\to\infty} \psi(t) = 0$$

Archimedean copulas have the following interesting properties:

- They are symmetric, that is invariant by any permutation of their coordinates, which is important when dealing with exchangeable random variables, as is the case here[1];
- They are associative: for any $(u_1, \cdots, u_p) \in [0,1]^p$, one has:

$$\begin{aligned} &C_{p-1}(C_2(u_1,u_2;\psi),u_3,\cdots,u_p;\psi) \\ &= C_{p-1}(u_,\cdots,u_{p-2},C_2(u_{p-1},u_p;\psi);\psi) \end{aligned}$$

  This means that the dependency properties are the same whatever the way we group the random variables.

In this study, we further consider a particular case of the Archimedean copulas, namely the one–parameter family of Franck copula, defined, for any $\lambda \in \mathbb{R} \setminus \{0\}$, as:

$$C_\lambda(u,v) = -(1/\lambda)\ln(1 + \frac{(e^{-\lambda u}-1)(e^{-\lambda v}-1)}{e^{-\lambda}-1}) \tag{3.8}$$

When $\lambda \to 0$, one approaches the independency copula, whereas $\lambda = \infty$ yields the comonotonicity copula. Lastly, for any $\lambda \in \mathbb{R} \setminus \{0\}$, C_λ is twice differentiable on $[0,1]^2$ so that the copula function admits a density, denoted in the sequel c_λ.

$$\begin{aligned} c_\lambda(u,v) &= \frac{\partial^d C_\lambda(u,v)}{\partial u \partial v} \\ c_\lambda(u,v) &= \frac{\lambda[1-e^{-\lambda}][e^{-\lambda(u+v)}]}{([1-e^{-\lambda}]-(1-e^{-\lambda u})(1-e^{-\lambda v}))^2} \end{aligned}$$

[1]The LDA model is based on the assumption that topics are infinitely exchangeable within a document.

By varying λ from 0 to ∞, Franck copula allows one to model all the possible dependencies between two random variables, from complete independency to equality. Dependency/independency is furthermore controlled by a single parameter, λ, which makes parameter estimation both easier and more robust.

3.2.2 Presentation of ST−LDA−C model

Instead of generating the topic distribution of each document θ^d independently, as is done in standard LDA we bind, as for our first model, ST−LDA−D, the topic distributions θ^{d-1} and θ^d of consecutive documents, this time by using copulas, and more precisely Franck copula.

One can not however directly use Sklar's theorem as it does not extend to joint distributions over random vectors. This means that if we are given two random vectors $\mathbf{X}_1, \mathbf{X}_2$, one can not claim that there exists a copula C such that, for any $(\mathbf{x_1}, \mathbf{x_2}) \in [0,1]^{p_1} \times [0,1]^{p_2}$:

$$F_{\mathbf{X}_1,\mathbf{X}_2}(\mathbf{x_1}, \mathbf{x_2}) = C(F_{\mathbf{X}_1}(\mathbf{x_1}), F_{\mathbf{X}_2}(\mathbf{x_2}))$$

except in very specific situation as when $\mathbf{X_1}$ and $\mathbf{X_2}$ are independent for example. One can nevertheless relate latent topics θ^{d-1} and θ_d through their components. Indeed, the topic Dirichlet distribution can be decomposed into univariate Gamma distributions with parameters $(\alpha, 1)$, denoted $Ga(\alpha)$:

Theorem 3.2 *(from Theorem 2.1 of [Ng et al., 2011]) A random vector θ follows a Dirichlet distribution $Dir(\alpha)$ iff there exists a random vector $\mathcal{T} \sim Ga(\alpha) \otimes \cdots \otimes Ga(\alpha)$ such that:*

$$\theta \overset{(\mathcal{L})}{=} \frac{\mathcal{T}}{\|\mathcal{T}\|_{\ell_1}} \tag{3.9}$$

where $\overset{(\mathcal{L})}{=}$ means "equality in distribution". In addition, if we are given $\theta \sim Dir(\alpha)$ and $R \sim Ga(K\alpha)$ independent, then $\mathcal{T} = R\theta \sim Ga(\alpha) \otimes \cdots \otimes Ga(\alpha)$.

To bind the topic distributions θ^{d-1} and θ^d of two consecutive documents, we thus consider the associated vectors $\mathcal{T}^{d-1}$ and $\mathcal{T}^d$, and bind them coordinate per coordinate using Franck copula. For the word-topic distributions, we use the same coupling between consecutive documents as the one used in model ST−LDA−D, as a tighter coupling through copulas would be too costly. We will come back to this issue in Section 3.4.

In the sequel for any $\gamma > 0$, f_γ (resp. F_γ) denotes the pdf (resp. cdf) of the Gamma distribution with parameters $(\gamma, 1)$. The global generative model is thus as follows:

1. Generate the first document according to the standard LDA model
2. For each document d, $2 \leq d \leq D$:

(a) Generate $\lambda_d \sim U[0, \tau_\lambda]$

(b) Generate $\mu_d \sim U[0, \tau_\mu]$

(c) For each topic k, $1 \leq k \leq K$:

- Generate $\mathscr{T}_k^d$ whose conditional density w.r.t. $\mathscr{T}_k^{d-1}$ is:

$$P(\mathscr{T}_k^d | \mathscr{T}_k^{d-1}) = f_\alpha(\mathscr{T}_k^d)\, c_{\lambda_d}(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d))$$

- Generate $\phi_k^d | \phi_k^{d-1} \sim Dir(\beta + \mu_d \phi_k^{d-1})$

(d) Set $\theta^d = \mathscr{T}^d / \|\mathscr{T}^d\|_{\ell_1}$

(e) For each word n, $1 \leq n \leq N$ in d:

- Choose a topic assignment: $z_n^d \sim mult(1, \theta^d)$
- Choose the word w_n^d from the topic z_n^d with probability $P(w_n^d | z_n^d) = \phi^d_{z_n^d, w_n^d}$

where $\mathscr{T}_k^d$ represents the k^{th} coordinate of the vector $\mathscr{T}^d$, and follows a distribution $Ga(\alpha)$ according to Theorem 3.2. We refer to the corresponding model as `ST-LDA-C`. Figure 3.1 provides a graphical representation of this model, together with the ones of previous models.

3.2.3 Inference with gibbs sampling for `ST-LDA-C`

The updates for z^d, ϕ^d and μ_d are identical to the ones for model `ST-LDA-D`. For λ_d, one gets:

$$P(\lambda_d | \mathscr{T}^{d-1}, \mathscr{T}^d, z^d, w^d, \alpha, \beta, \phi^{d-1}, \phi^d, \mu_d) \propto$$
$$P(\lambda_d) \prod_{k=1}^{K} f_\alpha(\mathscr{T}_k^{d-1}) f_\alpha(\mathscr{T}_k^d) c_\lambda(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d))$$

The same Metropolis-Hasting procedure as the one used for model `ST-LDA-D` and detailed in Appendix A.1 can then be used.

For θ^d, one needs first to estimate the conditional probability of the random vector $\mathscr{T}^d$ with respect to the other parameters. This expression can be factored as follows:

$$P(\mathscr{T}^d | \mathscr{T}^{d-1}, z^d, w^d, \alpha, \beta, \lambda_d, \phi^{d-1}, \phi^d, \mu_d) = \frac{P(\mathscr{T}^d | \mathscr{T}^{d-1}, \alpha, \lambda_d) P(z^d | \mathscr{T}^d)}{P(z^d | \alpha)}$$

As in the classical context of LDA, one has $P(z^d|\alpha) = B(\Omega_d + \alpha)/B(\Omega_d)$ where Ω_d is defined as before. By assumption on the distribution of the random vectors $(\mathscr{T}^{d-1}, \mathscr{T}^d)$:

$$P(\mathscr{T}^d|\mathscr{T}^{d-1}, \alpha, \lambda_d) = \prod_{k=1}^{K} f_\alpha(\mathscr{T}_k^d) c_\lambda(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d))$$

Developing $P(z^d|\mathscr{T}^d)$ as detailed in Appendix A.2, finally leads to:

$$\begin{aligned} P(\mathscr{T}^d|\mathscr{T}^{d-1}, z_d, w_d, \alpha, \beta, \lambda_d, \phi^{d-1}, \phi^d, \mu_d) \propto & (\sum_{k=1}^{K} \mathscr{T}_k^d)^{-N} \\ & \times \prod_{k=1}^{K} f_{(\Omega_{d,k}+\alpha-1)}(\mathscr{T}_k^d) \times c_\lambda(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d)) \end{aligned} \tag{3.10}$$

Each $\mathscr{T}_k^d$ can then be estimated through the Metropolis-Hasting procedure presented in Appendix A.1; θ^d is finally obtained from $\mathscr{T}^d$ through Eq. 3.9.

3.3 Non parametric extension

The standard LDA model on which we have based our developments can be generalized in order to dispense with specifying the number of latent topics. Such a generalization amounts to consider a non-parametric extension based on Hierarchical Dirichlet Processes (HDPs) illustrated in Figure 3.2(a) and referred to here as iLDA for infinite LDA. Indeed, HDPs introduce a prior over the Dirichlet distribution used in LDA that leads to a model with an *a priori* infinite number of topics ([Heinrich, 2011]). However, for any collection, the number of active topics is always finite and determined during inference.

We here describe the basic definition of Dirichlet Process in brief, then we discuss three different interpretations on the Dirichlet [rocess. The first one based on the Stick-Breaking representation, the second one based on a Polya urn construction named Chinese Restaurant Process, and the last one formed by a limit of finite mixture models. Dirichlet Process was first etablished by [Ferguson, 197]. As [Teh et al., 2006] explained the Dirichlet Process, one assume Θ and B as two measurable spaces and G_0 as a probability measure on this spaces. Now one consider α_0 as a positive real number, then a Dirichlet Process of $DP(\alpha_0; G_0)$ can be defined as a distribution of a random probability measure like G over Θ and B spaces in the way that for any finite measurable partition $(A_1, A_2, \ldots, Ar)$ in Θ space, the random vector $(G(A_1), \ldots, G(A_r))$ will be a finite-dimensional Dirichlet distribution with parameters of $(\alpha_0 G_0(A_1), \ldots, \alpha_0 G_0(A_r))$. We can write $G \sim DP(\alpha_0; G_0)$ when G is a random probability distribution given by a Dirichlet Process. It means:

$$(G(A_1), \ldots, G(A_r)) \sim Dir(\alpha_0 G_0(A_1), \ldots, \alpha_0 G_0(A_r)) \tag{3.11}$$

As it is mentioned before, there exist three perspectives for the Dirichlet Process that we here detail them to choose one of them based on their characteristics.

3.3.1 Stick-Breaking representation for dirichlet process

The stick-breaking representation is formed by sequences of independent and identically distributed random variables $(\pi'_k)_{k=1}^{\infty}$ and $(\phi_k)_{k=1}^{\infty}$ as below:

$$\pi'_k|\alpha_0, G_0 \sim Beta(1, \alpha_0) \qquad and \qquad \phi_k|\alpha_0, G_0 \sim G_0$$

One can define a random measure G as

$$\pi_k = \pi'_k \prod_{l=1}^{k-1}(1-\pi'_l) \qquad and \qquad G = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k} \tag{3.12}$$

Where $\pi = (\pi_k)_{k=1}^{\infty}$ to satisfy the constraint $\sum_{k=1}^{\infty} \pi_k = 1$ and δ_{ϕ} is a probability estimation concentrated on ϕ. Similar to the measures drawn from a Dirichlet process, these random variables are discrete with probability one. [Sethuraman, 1994] proved that G as defined in this construction is a same random probability measure distributed according to $DP(\alpha_0, G_0)$. Using this definition we may write π as a random probability measure on the positive integers. Therefore, we may draw π from a $GEM(\alpha_0)$ distribution [Pitman, 2002].

3.3.2 Chinese restaurant process for dirichlet process

The second perspective of Dirichlet process is based on the Polya urn construction introduced by [Blackwell and MacQueen, 1973]. The Polya urn scheme uses the property of DP that drawing from a Dirichlet process is discrete and also implies a clustering attitude. In fact, The Polya urn scheme is not referring to G directly, instead it uses the draws from random measure G. Again one assume $\theta_1, \theta_2, \ldots$ a sequence of independent and identically distributed random variables distributed based on G. Here it means, the random variables $\theta_1, \theta_2, \ldots$ are conditionally independent and hence exchangeable given G. [Blackwell and MacQueen, 1973] showed that one can consider the conditional distributions of θ_i given the rest of $\theta_1, \ldots, \theta_{i-1}$, and G integrated out as follows:

$$\theta_i|\theta_1, \ldots, \theta_{i-1}, \alpha_0, G_0 \sim \sum_{l=1}^{i-1} \frac{1}{i-1+\alpha_0}\delta_{\theta_l} + \frac{\alpha_0}{i-1+\alpha_0}G_0 \tag{3.13}$$

Here again δ_θ is a probability measure concentrated on θ. This conditional distributions can be interpreted as a simple urn model. In this model, for each atom we assume a ball of a distinct color. The balls are drawn with equal probability and when a ball is drawn from the urn it should be get back to the urn together with another ball from the same color. Additionally, by drawing from G_0 with probability that is proportional to α_0, a new atom can be created means a ball with new color is added to the urn. Equation 3.13 has another part shows that θ_i has a positive probability of drawing a ball similar to the previous draws. In this model, there is also intrinsically a reinforcement; it is expressed the more oa ball with a color is drawn, the more probable it is to be drawn again in the future.
Another property of this representation is the clustering property that can be implied using a different interpretation of the Polya urn scheme which is close to the Chinese Restaurant Process [Aldous, 1985]. Chinese Restaurant Process turns out to be useful for generalizing the Dirichlet Process in a simple and meaningful way. For having another representation of Polya urn, we assume a new set of random variables that show different values for the atoms. We define $\phi_1, \ldots, \phi_K$ to be the different values that are supposed to be taken by $\theta_1, \ldots, \theta_{i-1}$, and let m_k be the number of times $\theta_{i'}$ for $1 \leq< i' < i$ are equal to ϕ_k, then we can redefine the equation 3.13 as:

$$\theta_i|\theta_1, \ldots, \theta_{i-1}, \alpha_0, G_0 \sim \sum_{k=1}^{K} \frac{m_k}{i-1+\alpha_0}\delta_{\phi_k} + \frac{\alpha_0}{i-1+\alpha_0}G_0 \tag{3.14}$$

Now the Chinese Restaurant Process interpretation would be like this: assume we have a Chinese restaurant with an unlimited number of tables. For each costumer who gets enter to the restaurant there is a θ_i corresponds to that customer and the distinct values of ϕ_k correspond to the tables that the customers is going to sit at. The i^{th} customer will sit at the table labeled by ϕ_k, with probability proportional to the number of customers of m_k that already seated in table ϕ_k (we now set $\theta_i = \phi_k$), and will sit at a new table with probability proportional to α_0 where we need to increment the K, and draw the new $\phi_K \sim G_0$ and set $\theta_i = \phi_K$ in the model.

3.3.3 Finite mixture models for dirichlet process

As [Rasmussen, 1999, Green, 2001, Ishwaran and Zarepour, 2002] have shown, another representation of Dirichlet Process can be inferred by a limit over sequence of finite mixture models, with infinite number of mixture components. This limiting process forms the third perspective over Dirichlet process. We suppose L mixture components with mixing proportions of $\pi = (\pi_1, \ldots, \pi_L)$. The πs were denoted to the weights associated with atoms in random measure G in the stick-breaking model. Here in this model, one can deliberately redefine π in this way as they are completely relevant in two models. In fact, [Pitman, 1996] showed with the limit $L \to \infty$ these π vectors are equivalent regards to a random permutation of their entries having a size biased. In this model π is drawn from a Dirichlet distribution with symmetric hyper-parameters $(\alpha_0/L, \ldots, \alpha_0/L)$ and ϕ_k sampled from a categorical distribution over G_0 and devoted to a random variable associated with the mixture component of k. Finally one can draw an observation x_i from the mixture model by picking a specific mixture component z_i with probability given by the mixing proportions π. The model is as follows:

$$\begin{aligned} \pi|\alpha_0 &\sim Dir(\alpha_0/L, \ldots, \alpha_0/L) & z_i|\pi &\sim \pi \\ \phi_k|G_0 &\sim G_0 & x_i|z_i, (\phi_k)_{k=1}^L &\sim F(\phi_{z_i}) \end{aligned} \tag{3.15}$$

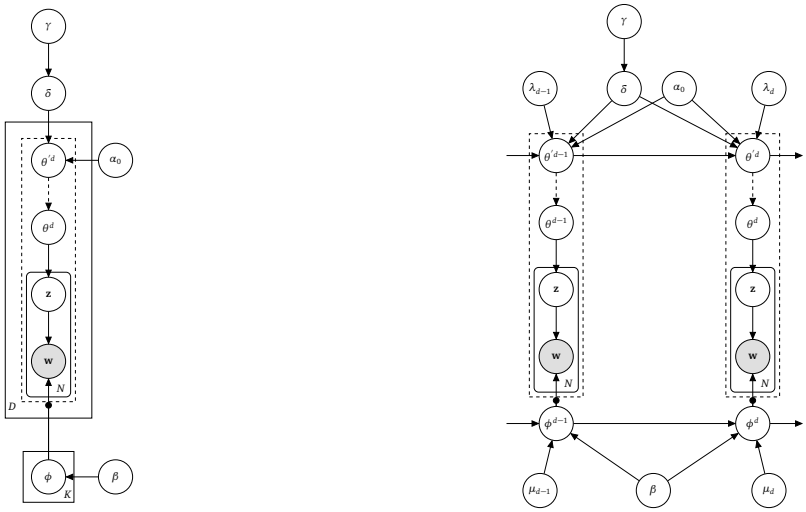
Assuming $G^L = \sum_{k=1}^L \pi_k \delta_{\phi_k}$ [Ishwaran and Zarepour, 2002] showed that for every function of f which is integrable regarding G_0 when $L \to \infty$ we have:

$$\int f(\theta) dG^L(\theta) \to \int f(\theta) dG(\theta) \tag{3.16}$$

This shows that the marginal distribution on the observations $x_1, \ldots, x_n$ will be the same as the one in Dirichlet process model.

There are consecutively three different perspectives on the Hierarchical Dirichlet Process by incorporating an appropriate non-parametric prior to Dirichlet Process based on the Stick-Breaking construction or based on a Polya urn model(Chines Restaurant Process) or based on a limit of finite mixture models to infinite. Because of the decomposition it provides on the latent topics, we rely here on the stick-breaking construction.

3.3.4 Stick-breaking construction for iLDA



(a) HDP (b) CopHDP

Figure 3.2: Graphical models for non parametric extensions of LDA (left, iLDA model of [Teh et al., 2006]) and of streaming LDA (right, model CopHDP). Both extensions are based on Hierarchical Dirichlet Processes; we make use here of the stick-breaking construction for these processes.

The generative process for iLDA based on the stick-breaking construction (illustrated in Figure 3.2(a)) goes as follows:

1. Draw a base distribution $\delta|\gamma \sim GEM(\gamma)$. This amounts to generate indenpendent $\delta_1, \cdots, \delta_k, \cdots$ variables as follows:

$$\begin{aligned} \delta_k^{'} &\sim Beta(1,\gamma) \text{ for } && k = 1, \ldots, \infty \\ \delta_k &= \delta_k^{'} \prod_{\ell=1}^{k-1} (1-\delta_\ell^{'}) \end{aligned} \tag{3.17}$$

   where γ is a concentration parameter for δ. By construction, $\sum_k \delta_k = 1$.

2. Then, for each document d, draw $\theta^d|\alpha_0, \delta \sim DP(\alpha_0, \delta)$. This amounts to generate each coordinate θ_k^d $(k = 1, \ldots, \infty)$ according to:

$$\theta_k^{'d} \sim Beta(\alpha_0\delta_k,\, \alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)) \qquad and \qquad \theta_k^d = \theta_k^{'d}\prod_{\ell=1}^{k-1}(1-\theta_\ell^{'d})$$

where α_0 plays the role here of a scaling parameter.

3. Once θ^d has been generated, one can proceed with the generation, for each position in the document, of the topics z_n^d and then of the word w_n^d after having drawn $\phi \sim Dir(\beta)$ as in standard LDA.

We now introduce an extension of the above model that takes into account dependencies between topics using copulas.

3.3.5 Copula-based extension for iLDA

Similarly to the development proposed in Section 3.1, one can incorporate dependencies between topics of consecutive documents by coupling the variables θ'^d on each dimension. This leads to the following generative model, illustrated in Figure 3.2(b):

1. Draw δ following equation (3.17),
2. Then:
   - For the first document:
     - For each k,
     $$\theta_k^{'1} \sim Beta(\alpha_0\delta_k, \alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)) \quad and \qquad \theta_k^1 = \theta_k^{'1}\prod_{\ell=1}^{k-1}(1-\theta_\ell^{'1})$$
     - Then generate the document according to the standard LDA model.
   - For each document d, $2 \leq d \leq D$:
     (a) Generate $\lambda_d \sim U[0, \tau_\lambda]$
     (b) Generate $\mu_d \sim U[0, \tau_\mu]$
     (c) For each topic k,
       - Let G_k (resp g_k) denote the cdf (resp pdf) of the Beta distribution with parameters $(\alpha_0\delta_k, \alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell))$.
       - Generate $\theta_k^{'d}$ whose conditional density w.r.t. $\theta_k^{'d-1}$ is:
       $$P(\theta_k^{'d}|\theta_k^{'d-1}) = g_k(\theta_k^{'d})\, c_{\lambda_d}(G_k(\theta_k^{'d-1}), G_k(\theta_k^{'d}))$$
       Then set:
       $$\theta_k^d = \theta_k^{'d}\prod_{\ell=1}^{k-1}(1-\theta_\ell^{'d}) \tag{3.18}$$

- Generate $\phi_k^d|\phi_k^{d-1} \sim Dir(\beta + \mu_d \phi_k^{d-1})$

(d) For each word n, $1 \leq n \leq N$ in d:

- Choose a topic assignment: $z_n^d \sim Mult(1, \theta^d)$
- Choose the word w_n^d from the topic z_n^d with probability $P(w_n^d|z_n^d) = \phi^d_{z_n^d, w_n^d}$

As before, we rely on Franck copula, defined in Eq. 3.8.

3.3.6 Inference with gibbs sampling for CopHDP

Follwing [Teh and Jordan, 2010], one can sample δ using:

$$\delta_1, \ldots, \delta_K, \delta_{K+1}|\gamma \sim Dirichlet(m_{.1}, \ldots, m_{.K}, \gamma) \tag{3.19}$$

where $m_{.k}$ is number of times that δ_k, as a base proportion, has been used to create a new topic from the Dirichlet Process. As [Heinrich, 2011] mentioned, simulating how new topics are created in document d using δ_k is a sequence of Bernoulli trials. Furthermore, as shown in [Antoniak, 1974]:

$$P(m_{d,k} = m)|z, m^{-d,k}, \delta) = \frac{\Gamma(\alpha_0 \delta_k)}{\Gamma(\alpha_0 \delta_k + \Omega_{d,k})} s(\Omega_{d,k}, m)(\alpha_0 \delta_k)^m \tag{3.20}$$

where $s(n, m)$ are unsigned Stirling numbers of the first kind and $\Omega_{d,k}$ counts the number of time a word occurred for document d and topic k. By sampling $m.k$ from equation 3.20, one can simply draw base proportions δ_k for $k \in 1, \ldots, K+1$ using Equation 3.19. Note that γ is used in this equation to create a new topic, indexed by $K+1$.

The estimation of θ'^d is based on:

$$p(\theta'^d|\theta'^{d-1}, z^d, w^d, \alpha_0\delta, \beta, \lambda_d, \mu_d, \phi^d, \phi^{d-1}) = \frac{p(\theta'^d, \theta'^{d-1}, z^d, w^d, \lambda_d, \mu_d, \phi^d, \phi^{d-1}|\alpha_0\delta, \beta)}{p(\theta'^{d-1}, z^d, w^d, \lambda_d, \mu_d, \phi^d, \phi^{d-1}|\alpha_0\delta, \beta)}$$

With:

$$p(\theta'^d, \theta'^{d-1}, z^d, w^d, \lambda_d, \mu_d, \phi^d, \phi^{d-1}|\alpha_0\delta, \beta) = p(w^d|z^d, \phi^d)p(z^d|\theta'^d)p(\theta'^d|\theta'^{d-1}, \alpha_0\delta)p(\theta'^{d-1}|\alpha_0\delta)$$

And:

$$p(\theta'^{d-1}, z^d, w^d, \lambda_d, \mu_d, \phi^d, \phi^{d-1}|\alpha_0\delta, \beta) = p(w^d|z^d, \phi^d)p(\theta'^{d-1}|\alpha_0\delta)p(z^d|\alpha_0\delta)$$

So:

$$p(\theta'^{d}|\theta'^{d-1}, z^d, w^d, \alpha_0\delta, \beta, \lambda_d, \mu_d, \phi^d, \phi^{d-1}) = \frac{p(z^d|\theta'^{d})p(\theta'^{d}|\theta'^{d-1}, \alpha_0\delta)}{p(z^d|\alpha_0\delta)}$$

Analogous to equation 3.10 and from Appendix A.3 we have:

$$p(\theta'^{d}|\theta'^{d-1}, z^d, w^d, \ldots, \phi^{d-1}) \propto \prod_{k=1}^{K+1} g_{\Omega_{d,k}+\alpha_0\delta_k, \sum_{m=k+1}^{K+1}\Omega_{d,m}+\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d}) \times \prod_{k=1}^{K+1} c_\lambda(G_{\alpha_0\delta_k, \alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d-1}), G_{\alpha_0\delta_k, \alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d})) \quad (3.21)$$

Each $\theta_k'^{d}$ can then be estimated through a Metropolis-Hasting procedure based on Eq. 3.21. Finally one can find θ_k^d using equation 3.18.

The estimation of λ, μ, z and ϕ follows the same procedure as the one for `ST-LDA-C`, while taking care of potentially added topics (see below).

3.4 Computational considerations

The word-topic distributions $\phi_k^d (1 \leq k \leq K)$ can be estimated in the same way as θ^d is estimated, as mentioned in Section 3.2. However, this would entail running $K \times V$ Metropolis-Hasting procedures, which is problematic as soon as the collections considered are relatively large. We thus proposed in Section 3.2 to estimate these distributions through Eq. 3.6, for both `ST-LDA-D` and `ST-LDA-C`, as only $K \times V$ Gibbs sampling updates are required. If this estimation procedure is faster, it may still be too slow for really large collections. Theorem 3.2 nevertheless suggests a way to approximate $\phi_k^d (1 \leq k \leq K, 2 \leq d \leq D)$ through Gamma updates, as follows:

1. For each word v in d, generate $t_{k,v} \sim Ga(\beta + \phi_{k,v}^{d-1})$
2. For each word v in the vocabulary $\mathcal{V}$, $\phi_{k,v}^d \leftarrow \frac{t_{k,v}}{\sum_{v \in \mathcal{V}} t_{k,v}}$

where β corresponds to the real parameter (*i.e.*, the constant value that makes up the V dimensional vector of priors). The quantities $t_{k,v}$ are first initialized through $t_{k,v} \sim Ga(\beta)$, and updated each time a new document is encountered.

As one can note, this update primarily concerns the words present in the current document (step 1), the components for the other words being just renormalized (step 2). This contrasts with Eq. 3.6 in which the contribution of all words is resampled for each document via a multivariate Dirichlet distribution. The above procedure simplifies this by relying on the univariate equivalent of the Dirichlet distribution, namely the Gamma distribution, and by binding the variables through the renormalization step. It is faster as it involves only $K \times N$ samplings from a Gamma distribution instead of K samplings from a multivariate, $V (V >> N)$ dimensional Dirichlet distribution (the $K \times V$ renormalizations in step 2 do not really harm the procedure and are negligible compared to the Dirichlet samplings).

We have observed in practice no difference between this procedure and the more complex ones mentioned before, and make use of it in the remainder of this chapter. In terms of speed, this procedure performed 1.5 times faster on the NIPS collection, which contains long documents and a relatively small vocabulary (*ca.* 12,000 words), and 2 times faster for the TDT4 and Tweets collections, which contain shorter documents with a larger vocabulary, up to 42,000 words (see Section 5.3).

Algorithm 1 summarizes the inference process we rely on for `ST-LDA-D` and `ST-LDA-C`. It makes use of the above procedure to estimate ϕ, referred to as ϕ-

procedure. Algorithm 2 summarizes the inference process for the `CopHDP`. In this inference, we use two additional variables (U_1 for active topics and U_0 for inactive topics) to keep track of the evolution of topics. ϕ is also estimated with the ϕ-procedure above.

Algorithm 1: Inference process for `ST-LDA-[D|C]`

Input: Stream of D documents of length N; number of topics K

Output: For each document d, topic distribution θ^d, word-topic distributions ϕ_k^d $(1 \leq k \leq K)$; for each word v in d, topic assignment z_v^d

// Initialization

for $k = 1$ *to* K, $v \in \mathcal{V}$ **do**
    $t_{k,v} \sim Ga(\beta)$
for $d = 1$ *to* D **do**
    Random initialization of λ_d, μ_d and z_n^d, $1 \leq n \leq N$
$\lambda_1 = \mu_1 = 0$

// Document processing

for $d = 1$ *to* D **do**
    repeat
        For `ST-LDA-D`: update θ^d acc. to Eq. 3.5
        For `ST-LDA-C`:
            (a) update $\mathcal{T}^d$ (Metropolis-Hasting)
            (b) obtain θ^d from $\mathcal{T}^d$ through Eq. 3.9
        Update ϕ_k^d acc. ϕ-procedure
        Update λ_d and μ_d (Metropolis-Hasting), $d > 2$
        Update z_n^d acc. to Eq. 3.7, $1 \leq k \leq K$, $1 \leq n \leq N$
    until *estimates are stable*

Algorithm 2: Inference process for CopHDP

Input: Stream of D documents of length N; initial number of topics K_0.

Output: For each document d, topic distribution θ^d, word-topic distributions ϕ_k^d $(1 \leq k \leq length(U_1))$; for each word v in d, topic assignment z_v^d, number of topics

// Initialization

$U_1 = [1, \ldots, K_0]$ active topics, $U_0 = [\ ]$ inactive topics

for $k = 1$ *to* $length(U_1) + 1$, $v \in \mathcal{V}$ **do**

  $t_{k,v} \sim Ga(\beta)$

  $\delta_k = 1/K_0$

for $d = 1$ *to* D **do**

  Random initialization of λ_d, μ_d and z_n^d, $1 \leq n \leq N$

$\lambda_1 = \mu_1 = 0$

// Document processing

for $d = 1$ *to* D **do**

  repeat

    For each topic sample m acc. to Eq. 3.20, then update δ acc. to Eq. 3.19

    Update θ'^d acc. to Eq. 3.21 (Metropolis-Hasting)

    Obtain θ^d from θ'^d through Eq. 3.18

    Update ϕ_k^d acc. ϕ-procedure

    Update λ_d and μ_d (Metropolis-Hasting), $d > 2$

    for $n = 1$ *to* N **do**

      $topic_{old} = z_n^d$

      Update z_n^d acc. to Eq. 3.7 with $1 \leq k \leq length(U_1) + 1$

      if $z_n^d == K + 1$ **then**

        if U_0 *is empty* **then**

          $topic_{new} = K + 1$

          Append $topic_{new}$ to the end of U_1

          Add a topic coordinate to the end of δ, θ', θ and ϕ

          Update m,δ and eventually θ

        else

          $topic_{new}$ = pop out first element of U_0

          Append $topic_{new}$ to the end of U_1

          Update the $topic_{new}$'s coordinate of δ, θ', θ and ϕ

      if $\Omega^d_{topic_{old}} == 0$ **then**

        remove $topic_{old}$ from U_1 and add it to U_0

        Update m,δ and eventually θ

  until *estimates are stable*

Number of topics = $Length(U_1)$

3.5 Experimental study

We relied on five datasets with different properties for analyzing our methods:

- The NIPS dataset contains 1,500 scientific papers with no time dependency between them. The size of the vocabulary is 12,375; Documents contain 500 unique words in average. The collection was collected from the NIPS proceedings and is relatively homogeneous in terms of the topics covered. This collection allows one to assess whether topic dependencies are still useful in a "loose" context in which there is no clear temporal dependency. It is available at the UCI ML Repository [Lichman, 2013];

- The Multilingual Text and Annotations data set TDT4[2], proposed for topic detection and tracking, has 3,190 original documents in English and a vocabulary comprising 22,965. Documents are ordered by time and correspond to newswire articles extracted from different broadcasts; The number of unique words per document is 100 in average;

- The Tweets dataset is collected using Twitter's streaming API during 20 days from 8/10/2014 to 27/10/2014. The collection contains 72,592 tweets and a vocabulary of size 42,336. Tweets have been sequenced by time and are filtered over health issues using an SVM classifier trained over MeSH categories[3];

- The NYT dataset[4] consistes of articles, ordered by time, from the New York Times global news (from January 1st to December 31st, 2011). A complete description of this dataset can be found in [Yao et al., 2016];

- Lastly, the Tech dataset[5] is a one year (staring on 7th August 2011) excerpt from Techcrunch's blogs. It is also detailed in [Yao et al., 2016]. The documents are relatively long (in average 1,000 unique words) and ordered by time.

Each dataset was separated into training and test sets. The NIPS collection was randomly split into training (90% of the collection) and test (10% of the collection) sets.

[2]Linguistic Data Consortium, The Trustees of the University of Pennsylvania https://catalog.ldc.upenn.edu/LDC2005T16.

[3]https://www.nlm.nih.gov/mesh/

[4]https://github.com/yao8839836/COT/tree/master/data/NYT

[5]https://github.com/yao8839836/COT/tree/master/data/TechCrunch%201%20year%20(3%2C158%20docs)

For TDT4, we used the first 2800 newswires released in time for training, and the last 390 ones for testing. For the Tweets dataset, we used the tweets issued in the first 17 days for training (60,000 documents) and those of the last 3 days (12,000 documents) for testing. For NYT and Tech collections, we used approximately 10% of the documents from the last time stamps as test set. Table 3.1 summarizes the characteristics of these collections.

Table 3.1: Datasets used in our experiments along with their properties.

	NIPS	TDT4	Tweets	Tech	NYT
Documents in Train set	1,350	2,800	60,000	2,800	6,100
Documents in Test set	150	390	12,000	370	678
Vocabulary size	12,375	22,965	42,336	27,870	42,244
# of unique words per doc.	500	100	15	350	500
Words in total	1.9M	0.78M	0.9M	3,5M	1.1M

Evaluation. Results are evaluated over the test set using the widely used perplexity measure that can be calculated by [Blei et al., 2003].

$$perplexity(C^{test}) = \exp\left(\frac{-\sum_{d}\sum_{n}\log\sum_{k}\theta_{k}^{d}\times\phi_{k,v_{n}^{d}}^{d}}{D^{test}\times N}\right) \tag{3.22}$$

where C^{test} denotes the test collection, D^{test} is its size and v_n^d represents the word at position n in document d. The parameters θ_k^d and ϕ_k^d are estimated on the training set. Furthermore, for the TDT4 collection we use the available semantic labels of newswires in the test set in order to evaluate the ability of the models to find documents of the same semantic labels using only their predicted topic distributions (Section 3.5.2). To this aim, we measure ROC curves and AUC of different topic models on TDT4.

Settings and comparisons. For all models, both hyperparameters α and β were fixed to 0.5. γ is also fixed to 2.0 for the non-parametric models considering the constraint of Beta distribution mentioned before. Documents of the NIPS dataset are initially stoplisted, we did not perform further preprocessing of the data nor removed stop words from the TDT4 , Tweets, Tech and NYT documents as for all methods best results are obtained when collections are not filtered.

To validate the streaming LDA models described above, we tested several methods for comparison purposes:

- The first two are LDA models [Blei et al., 2003]: (a) $\texttt{LDA}_1$, which consists in training an LDA model on the whole training data, then fixing ϕ and updating θ for each document in the test set, (b) $\texttt{LDA}_{all}$, which consists in training an LDA model on the whole training data and updating both ϕ and θ for each document in the test set;

- In addition, we considered two state-of-the-art latent models that take into account dependencies between topics: Dynamic Topic Model (DTM) [Blei and Lafferty, 2006] and Temporal LDA (TM−LDA) [Wang et al., 2012]. DTM is certainly the most popular model to take into account topic dependencies. It is furthermore complete in the sense that it integrates both topic and word-topic distributions. TM−LDA is a very recent proposal with nice features;

- We used the standard non-parametric version of LDA, namely the Hierarchical Dirichlet Processe (HDP) model [Teh et al., 2006] that serves as a baseline for the non-parametric mixture topic models.

- We also used the Evolving Hierarchical Dirichlet Process (EHDP), one the most recent hierarchical streaming topic models that obtained good results in streaming environnements [Wang et al., 2017];

- Lastly, we also considered the three streaming LDA models we have introduced (ST−LDA−D ST−LDA−C[6] and CopHDP). For these last three models, τ_λ (see Appendix A.1) is set to 30,000 [7].

All the algorithms were implemented in Python with Numpy and Scipy[8] except DTM that is a C++ implementation tool from [Blei, 2008]. For both training and test, DTM is used considering that each document corresponds to a time slice.

3.5.1 Perplexity results

To measure the perplexity for each model, we estimate θ and ϕ over respectively all documents and all words of the training set. These estimates are then used to evaluate iteratively new ϕ and θ distributions for each document in the test set. This iterative update of ϕ and θ is done for all of the methods except $\texttt{LDA}_1$ in which ϕ is fixed and only θ is updated over the test documents.

[6] This can be found in https://github.com/Hesamalian/StreamingLDA-Copula

[7] This value, upper bounding λ_d, corresponds to a regime of the Franck copula close to comonotonicity.

[8] We are working to release all the programs developed in this study publicly available for research purpose.

Table 3.2 summarizes the perplexity results of all models, on all datasets, with the number of topics varying in the set $\{20, 40, 60, 80\}$ (this number is just used as initial value for the non-parametric models CopHDP and EHDP, whereas it is fixed for the other models). As one can note, on all collections, the best results are obtained with either CopHDP or ST−LDA−C, these two models being almost systematically (18 times out of 20) the best two models (represented in bold and italics in the table). They are followed by ST−LDA−D (which is twice the second best model) and EHDP, then LDA_{all}, HDP and DTM. TM−LDA the temporal LDA model, does not perform well as it is systematically worse than the the standard LDA model represented here by LDA_{all}. This result is however not really surprising as TM−LDA does not make advantage of the fact that the words in the new documents are known. Indeed, this model was designed for a slightly different purpose and its ability to predict future topics is not exploited here. All in all, we see here that the extra flexibility of the ST−LDA−[D|C] and CopHDP models allow them to outperform previously proposed ones.Comparing ST−LDA−[D|C] and CopHDP one can note that the two behave similarly. CopHDP is *a priori* more flexible than ST−LDA−C as the final number of topics is inferred from the data (and not predetermined). However, as one can note, the choice of the initial value for the number of topics impacts the results obtained so that one still has to test several initial values. This said the variation in perplexity according to the number of topics is less important for CopHDP than for ST−LDA−C, suggesting that the former is more stable than the latter on this aspect. On the other hand, it is also more time consuming (see below). Thus, if one does not have *a priori* knowledge on the number of topics and does not have time constraints, then CopHDP should be preferred; otherwise it should be ST−LDA−C.

Table 3.2: Perplexity with respect to different number of topics in $\{20, 40, 60, 80\}$. Best results are in bold, second best in italics.

Data	Topics	LDA_1	LDA_{all}	TM-LDA	DTM	HDP	EHDP	ST-LDA-D	ST-LDA-C	CopHDP
NIPS	20	2068.4	1625.4	2038.7	1737.5	1635.5	1624.7	1620.4	**1612.8**	*1616.6*
	40	2034.5	1534.7	2025.4	1551.2	1511.1	1506.5	1520.9	*1497.6*	**1479.6**
	60	1986.4	1458.1	1985.3	1450.7	1488.3	1460.7	*1450.2*	**1434.5**	1456.6
	80	1890.1	1450.1	1964.3	1418.4	1426.6	1412.9	1410.4	*1401.3*	**1398.7**
TDT4	20	900.8	723.1	876.7	869.1	750.6	746.4	*724.4*	**720.6**	735.2
	40	930.2	768.4	900.3	836.7	788.4	774.2	*758.1*	**752.5**	763.7
	60	960.4	792.7	916.3	820.9	791.2	786.2	784.4	*780.8*	**765.2**
	80	962.3	853.2	924.3	814.2	815.3	806.3	810.4	*802.3*	**784.4**
Tweets	20	470.8	431.8	455.1	559.4	415.3	404.1	393.9	**388.2**	*389.5*
	40	580.3	508.6	520.1	578.2	483.3	476.2	480.1	*474.1*	**447.12**
	60	615.5	577.1	585.2	607.4	563.3	551.7	552.7	*546.8*	**480.2**
	80	690.4	652.2	658.3	637.3	632.6	618.3	621.1	*617.3*	**526.2**
Tech	20	956.8	789.5	913.3	876.2	777.3	753.5	766.2	**741.6**	*742.7*
	40	972.3	801.3	926.4	825.5	784.2	769.3	771.2	*760.6*	**753.8**
	60	985.3	831.6	945.2	814.3	812.2	803.1	785.5	*774.8*	**772.9**
	80	998.9	856.6	973.6	812.7	821.4	806.4	803.5	*794.6*	**786.2**
NYT	20	900.9	723.1	832.1	825.3	725.2	714.4	703.1	**694.1**	*694.4*
	40	905.3	753.1	856.3	785.1	733.4	724.9	714.3	**696.2**	*712.8*
	60	926.2	781.2	888.2	755.2	742.3	731.8	722.1	**708.4**	*723.7*
	80	944.5	816.5	910.4	745.8	792.4	741.3	742.5	**721.4**	*738.4*

To further illustrate the behaviours of the different models, Figure 3.3 shows the evolution of perplexities of the parametric models with 80 topics over the test set, with respect to the training time of each model on the NIPS and TDT4 datasets (the non-parametric models are not considered here as their running time is not comparable to the one of parametric models). The code program of DTM (in C++) generally executes faster

than the other code programs (written in Python), we nevertheless ignore this detail and consider all the curves identically.

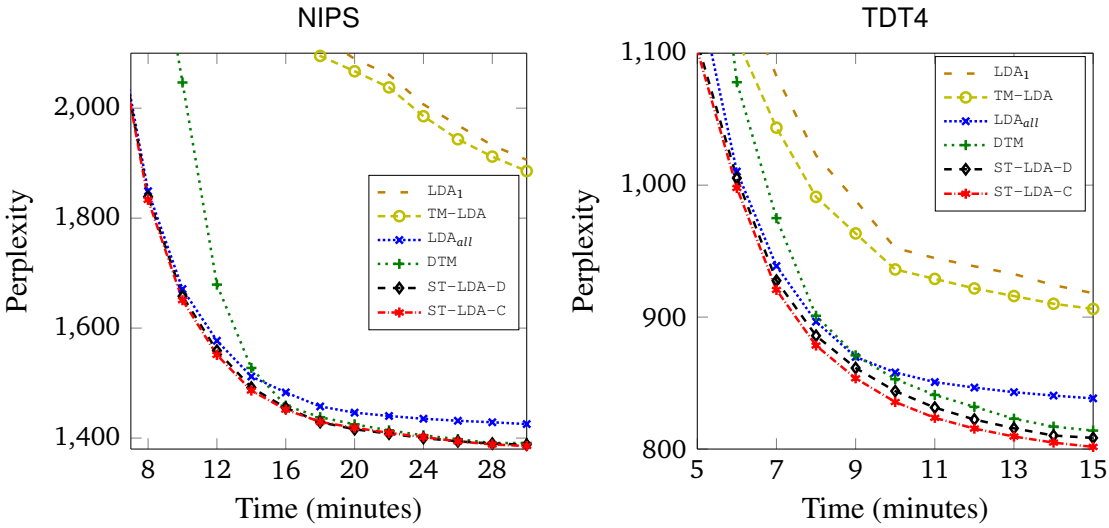


Figure 3.3: Perplexity curves with respect to time for all methods on NIPS and TDT4 collections (80 topics).

As expected, all perplexity curves decrease monotonically with respect to time. On both datasets, perplexity curves of ST−LDA−D and ST−LDA−C lower-bound the other curves on all iterations. On the NIPS dataset, DTM becomes competitive with the two others, at the end of the iterations, while on TDT4, where test documents come in a stream, ST−LDA−C stands clearly as the best model. These results show the ability of ST−LDA−C to capture dependencies between topics in document streams. Further, we note that at the beginning of iterations where dependencies are not yet apparent, the perplexity curves of both models are very similar to the one of LDA_{all}. This is in line with our assertion of the previous section that both models reduce to LDA in the case where topics are independent. As noted above, TM−LDA is not competitive in this setting as it does not make advantage of the fact that the words in the new, arriving documents are known.

In addition, Figure 3.4 illustrates the evolution of the perplexity on the Tweets dataset with 80 topics[9] when new tweets are continuously considered and used to estimate the parameters of the model (this experiment parallels the one presented in [Blei and Lafferty, 2006]). As once can note, all models need roughly the same amount of data (ca. 2,000 tweets) prior to have stable estimates of their parameters. The perplexity curves continue

[9] As before, this value is fixed for parametric models and serves as initial value for the non-parametric ones.

to decrease when new tweets are observed, but the decrease is less marked. TM−LDA and LDA_{all} do not behave well on this dataset and are slgihtly less stable (the perplexity increases after 2,000 tweets, prior to slowly decreasing again). A similar instability can be observed for DTM after 11,000 tweets. In contrast, the other models (ST−LDA−D, ST−LDA−C, HDP, EHDP and CopHDP) are more stable, the best performing model being here CopHDP.

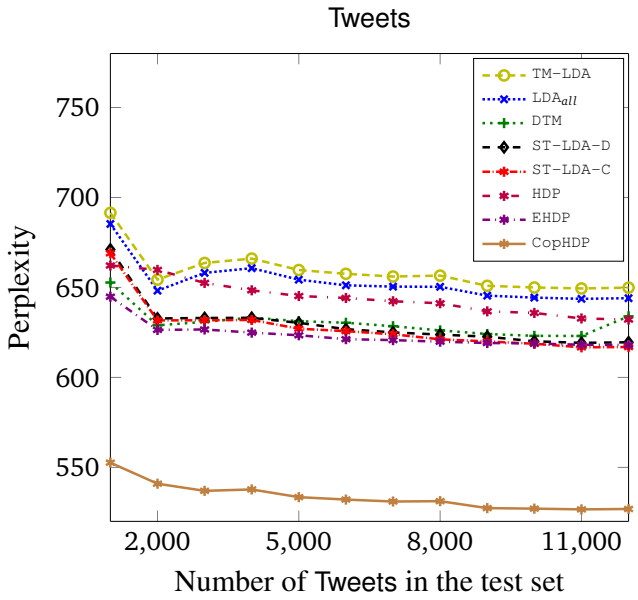


Figure 3.4: Perplexity of each method by number of tweets added to the test set (80 topics).

Table 3.3: Time consumption (in minutes) till convergence and perplexity reached (80 topics). Best method is in bold, second best in italics.

Data		LDA_1	LDA_{all}	TM−LDA	DTM	HDP	EHDP	ST−LDA−D	ST−LDA−C	CopHDP
NIPS	Time	41.5	36.4	39.3	33.6	65.8	72.7	*32.3*	**31.1**	54,2
	Perp	1890.1	1450.1	1964.3	1418.4	1426.6	1412.9	1410.4	*1401.3*	**1398.7**
TDT4	Time	20.4	17.3	19.7	16.2	30.2	33.1	*15.8*	**15.1**	28.3
	Perp	962.3	853.2	924.3	814.2	815.3	806.3	810.4	*802.3*	**784.4**

Lastly, Table 3.3 provides the running time for training the methods on the NIPS and

TDT4 as well as the perplexity obtained (again considering 80 topics)[10]. Convergence is here defined by the fact that the relative perplexity between two consecutive iterations is no more than 10^{-3}. As one can note, as expected, the parametric models run faster than the non-parametric models. Among the parametric models, ST−LDA−D and ST−LDA−C are by far the fastest ones. Similarly, CopHDP is the fastest model among the non-parametric family (that also contains HDP and EHDP) and the best model overall.

The fact that ST−LDA−D and ST−LDA−C run faster than the standard LDA models may seem surprising. Indeed, an iteration for ST−LDA−D and ST−LDA−C is slower than an iteration for LDA. The explanation lies here in the fact that the number of iterations required for convergence is lower for ST−LDA−D and ST−LDA−C than for the other models. The same applies for CopHDP and explains why it is faster than HDP.

[10] All experiments on a processor 3 GHz Intel Core i7 with memory 8 GB 1600 MHz DDR3.

3.5.2 Ability to detect semantic correlations

We further investigate on the ability of models to find topics that can detect documents of the same semantic class. For doing so, we used the TDT4 collection for which some documents are assigned semantic classes by experts. We hence use the cosine measure or the λ_d parameter of ST−LDA−C and CopHDP, to detect consecutive documents in the test set of this collection that are found similar on the basis of their topic distributions; two consecutive documents are considered as similar if the cosine measure of their topic distributions (resp. estimated λ_d - line 13 Algorithm 1) is higher than a given threshold. If two consecutive and similar documents share the same semantic label, we count them as a true positive; if they do not share the same semantic label, we count them as false positive. By changing the threshold, we can plot the ROC curves for the corresponding method.

Figure 3.5 depicts ROC curves of DTM, EHDP, TM−LDA, ST−LDA−C and CopHDP defined over 8 different thresholds taken in the set [0.2 0.5 0.7 0.86 0.89 0.92 0.95 0.98] for the cosine measure and [0.5 1 2 5 10 15 20 50] for λ_d when the number of topics is fixed to 20 and to 80.

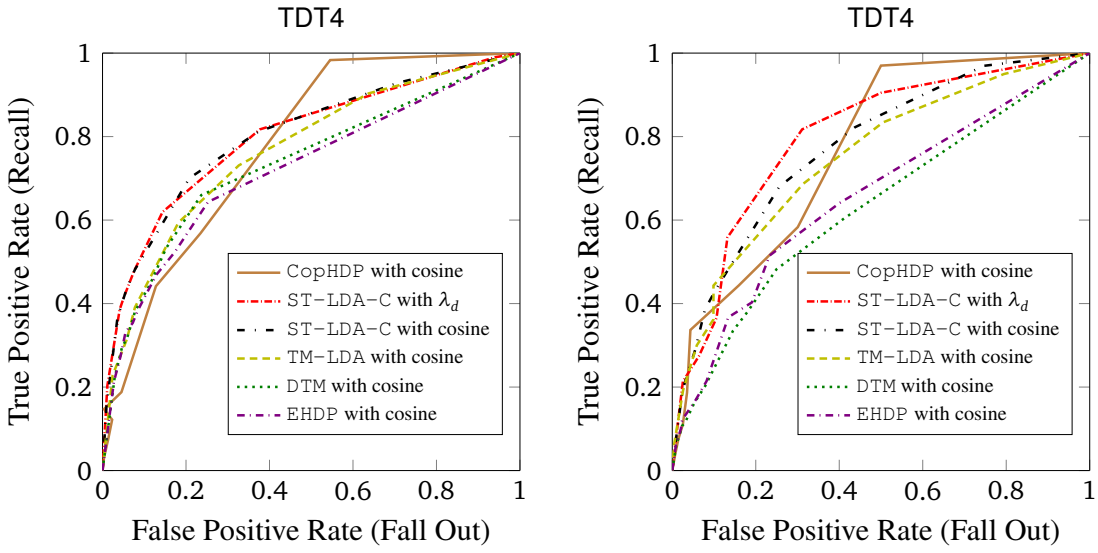


Figure 3.5: ROC curves of "semantic class matching" methods working over the topic distributions found by DTM, TM−LDA, ST−LDA−C and CopHDP for the number of topics fixed to 20 (left) and 80 (right).

In order to compare between the different ROC curves, we estimated the area under them, shown in Table 3.4. From these results it is clear that topic distributions found

by ST−LDA−C and CopHDP are more able to detect these semantic classes than topic distributions of DTM, EHDP and TM−LDA.

Table 3.4: Areas under the ROC curves of figure 3.5.

Methods	20 (Fig. 3.5, left)	80 (Fig. 3.5, right)
ST−LDA−C with λ_d	0.7982	**0.8306**
ST−LDA−C with cosine	**0.8004**	0.7755
CopHDP with cosine	0.775	0.7702
TM−LDA with cosine	0.7652	0.7349
EHDP with cosine	0.7201	0.6562
DTM with cosine	0.7357	0.6301

Finally, to illustrate the role of λ_d, we pictorially illustrate the correlation between the estimated λ_d and the topic distributions of three consecutive documents (Figure 3.6) with identical labels in the TDT4 collection. As one can see, the distributions of topics in the three pairs of consecutive documents with high λ_d are similar. In addition, the two most probable topics of the document pairs retained in Figure 3.7, also taken from TDT4, do not share any word when λ_d is small and are almost identical when λ_d is high. These examples illustrate the fact that λ_d is a good indicator of the topic dependencies between documents.

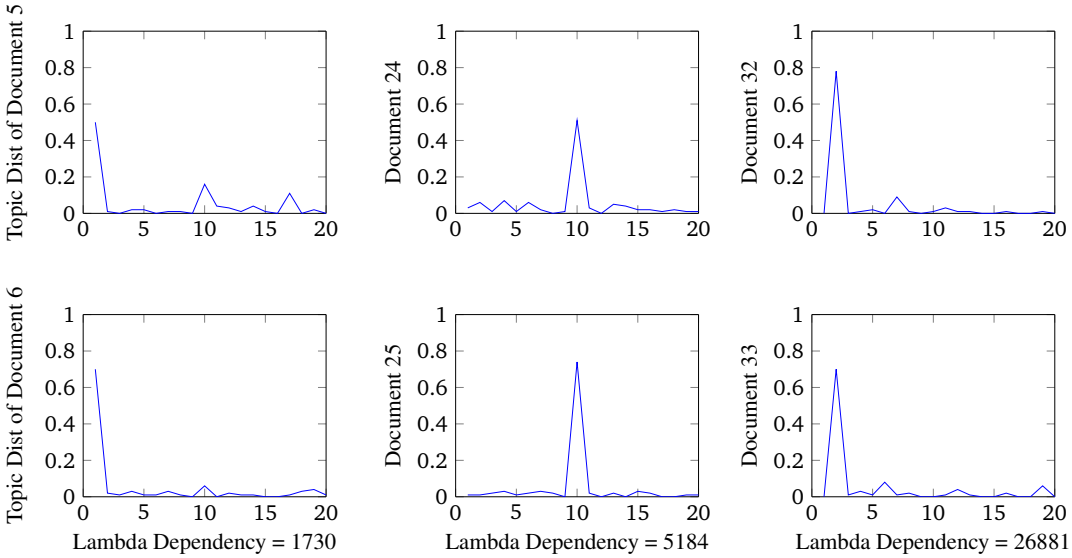


Figure 3.6: Topic distribution of three pairs consecutive documents that have the same topic (*Olympic* - left, *Election* - middle, *Sport* - right) and subject labels in TDT4 dataset (20 topics).

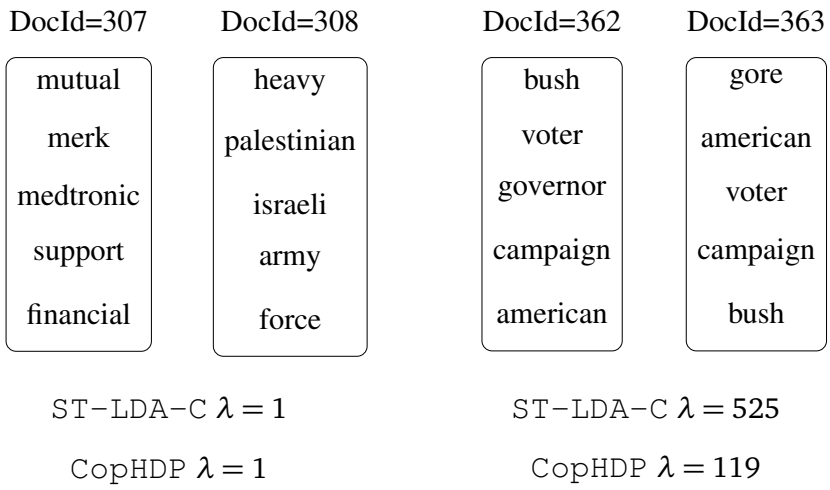


Figure 3.7: 5 most frequent words of the most probable topic (20 topics)

3.6 Summary

We have proposed in this chapter new models for modelling topic and word-topic dependencies between consecutive documents in document streams. The first model is a direct extension of Latent Dirichlet Allocation model (LDA) and makes use of a Dirichlet distribution to balance the influence of the LDA prior parameters wrt to topic and word-topic distribution of the previous document. The second extension makes use of copulas, which constitute a generic tool to model dependencies between random variables. Lastly, the third model is a non-parametric extension of the second one through the integration of copulas in the stick-breaking construction of Hierarchical Dirichlet Processes. Our experiments, conducted on five standard collections that have been used in several studies on topic modelling, show that our proposals outperform previous ones, as dynamic topic models, temporal LDA and the Evolving Hierarchical Processes, both in terms of perplexity and for tracking similar topics in a document streams. Compared to previous proposals, our models have extra flexibility and can adapt to situations where there is in fact no dependencies between the documents.

In the future, we plan to develop versions of these models that scale well, following the improvements on the inference methods for LDA, proposed in streams [Yao et al., 2009] or in online settings [Hoffman et al., 2010, Banerjee and Basu, 2007].

Chapter 4

Integrating text structure to LDA using copulas

A limitation inherent from the bag-of-words representation in such state-of-the-art models concerns the independence assumption: given their topics, words are assumed to occur independently. While this exchangeability assumption greatly impacts the involved computations and, in particular, the calculations of the conditional probabilities, it is rather naive and unrealistic [Heinrich, 2005]. As another limitation caused by the exchangeability assumption, the grouping of words in topically coherent spans, that is contiguous text spans like sentences, is lost.

On the other hand, text structure generally contains useful information that could be leveraged in inference process. Sentences or phrases, for instance, are by definition text spans complete in themselves that convey a concise statement. To better illustrate how text structure could help in topic identification, consider the example of Figure 4.1. It illustrates the topics inferred by `LDA` for the words (excluding stop-words) of a sentence drawn from a Wikipedia page. At the sentence level, one could argue that the sentence is generated by the "Cinema" topic since it discusses a film and its authors. `LDA`, however, fails and assigns several topics to the words of the sentence. Importantly, several of those topics like "Elections" and "Inventions" are unrelated. In finer text granularity, `LDA`also fails to assign consistent topics in noun-phrases like "film noir classic" and entities like "Brian Donlevy". A binding mechanism among the topics of the words of a sentence, or a phrase, could have prevented those limitations and taking simple text structure into account would be beneficial.

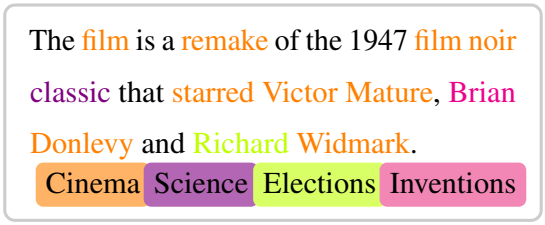


Figure 4.1: Applying LDA on Wikipedia documents.

Motivated by the previous example, we propose to incorporate text structure in the form of sentence or phrase boundaries as an intermediate structure in LDA. We plan to model this binding mechanism with copulas. Copulas have been found to be a flexible tool to model dependencies in the fields of risk management and finance [Embrechts et al., 2002]. They are a family of distribution functions that offer a flexible way to model the joint probability of random variables using only their marginals. This results in decoupling the marginal distributions by the underlying dependency. These properties make them appealing and some preliminary studies have started investigating their integration into different learning tasks [Wilson and Ghahramani, 2010, Tran et al., 2015, Amoualian et al., 2016].

The remainder of this chapter is organized as follows: the main contribution of this article is presented in next section, in which we propose to bind the latent topics that generate the words of a segment using copulas. We show that sampling word topics from copulas offers an elegant way to impose different levels and types of correlation between them. Section 4.3 then illustrates the behavior of copulaLDA, the copula-based version of LDA introduced in Section 4.1.2, while Section 4.4 concludes the chapter.

4.1 Integrating text structure to LDA

In this section we develop copulaLDA (hereafter copLDA), that extends LDA by integrating simple text structure in the model using copulas [Balikas et al., 2016b]. We assume that the topics that generate the terms of coherent text spans are bound. A strong binding signifies high probability for the terms to have been generated by the same topic. Therefore, as we show, the conditional independence of topics given the per-document topic distributions does not hold. Before presenting the generative and inference processes of copLDA, we shortly discuss the idea of *coherent text spans*.

Each sentence is a coherent, meaningful segment of text and we consider them as coherent text spans in this study. However, each sentence can be further decomposed into smaller segments through syntactic analysis. Figure 4.2 illustrates the output of a shallow parsing step of the example sentence of Figure 4.1, generated using the Stanford Parser.[1] Among these different segments, noun phrases play a particular role as they are, for instance, at the basis of terminology extraction that aims at capturing concepts from a document. Noun phrases usually constitute a semantic unit, pertaining to a given concept related to few, related topics. For this reason, we also consider noun phrases as coherent text spans in this study. Another advantage of the two types of coherent text spans we consider (whole sentences and noun phrases) is that they can be easily extracted using shallow parsing techniques, and one needs not resort to complex syntactic analysis in practice.

The film is a remake of *the 1947 film noir classic* that starred *Victor Mature*, *Brian Donlevy* and *Richard Widmark*.

Figure 4.2: Shallow parsing using the Stanford Parser. Contiguous words in italics denote a noun-phrase.

[1] http://nlp.stanford.edu/software/lex-parser.shtml

4.1.1 Apply copulas to random variables

As it mentioned in Section 3.2.1, Copulas are interesting because they separate the dependency structure of random variables from their marginals. Formally [Nelsen, 2007, Trivedi and Zimmer, 2007], a p-dimensional copula C is a p-variate distribution function with $C : \mathbb{I}^p = [0,1]^p \rightarrow [0,1]$ whose univariate marginals are uniformly distributed on $\mathbb{I}$ and $C(u_1, \ldots, u_p) = P(U_1 \leq u_1, \ldots, U_p \leq u_p)$. Copulas allow one to explicitly relate joint and marginal distributions, through Sklar's theorem [Sklar, 1959]. Once again, we present this theorem:

Theorem 4.1 *Let F be a p-dimensional distribution function with univariate margins $F_1, \ldots, F_p$. Let A_j denote the range of F_j. Then there exists a copula C such that for all $(x_1, \ldots, x_p) \in \mathbb{R}^p$*

$$F(x_1, \ldots, x_p) = C(F_1(x_1), \ldots, F_d(x_p)) \tag{4.1}$$

Furthermore, when $F_1, \ldots, F_p$ are all continuous, then C is unique.

As a result any multivariate distribution F can be decomposed into its marginals F_i, $i \in \{1, \ldots, p\}$ and a copula, allowing to study the multivariate distribution independently of the marginals. Sklar's theorem also provides a way of sampling multivariate distributions with a large number of random variables using copulas: $F(x_1, \ldots, x_p) = F\left(F_1^{-1}(u_1), \ldots, F_p^{-1}(u_p)\right) = P[U_1 \leq u_1, \ldots, U_p \leq u_p] = C(u_1, \ldots, u_p)$. Hence, to sample F it suffices to sample the dependence structure modeled by copulas and then transform the obtained sample in the marginals of interest using the probabilistic integral transform. We illustrate this transformation for one variable in Figure 4.3. Sampling the copula returns, for each variate, a sample as the one indicated in the histogram of the y axis. One can then transform the sample using the quantile (F^{-1}) of an arbitrary marginal.

Before proceeding further, we visit some extreme conditions of dependence illustrating the respective copulas that model them: (1) *Independence*, which is a frequently assumed simplification in topic models and is obtained with $\prod_{i=1}^{p} u_i$, and (2) *Co-monotonicity*, which is the complete, positive correlation between the random variables u_p, obtained with $\min(u_1, \ldots, u_p)$.

In the rest of our development we will be using a particular family of copulas, the Archimedean copulas. Archimedean copulas are widely used copulas and are defined with respect to a generator function ψ. They take the form: $C(u_1, \cdots, u_d) = \psi^{-1}(\psi(u_1) + \cdots +$

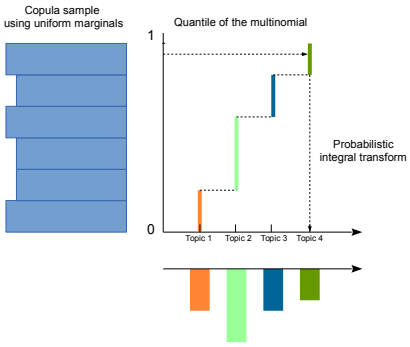


Figure 4.3: The transformation of a random variate to multinomial (or arbitrary) marginals. The arrows illustrate the generalized inverse; the histograms in y (resp. x) axis depict the distributions of the initial (resp. transformed) samples.

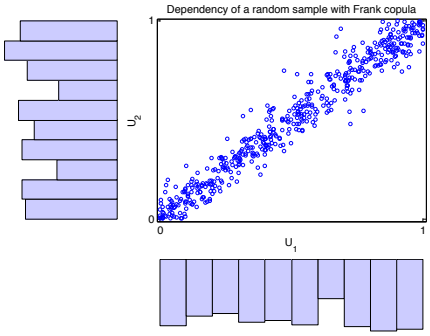


Figure 4.4: The positive correlation imposed to two random variates when sampling from a Frank copula with $\lambda = 25$. The histograms in x (resp. y) axis show the distributions of each of the variates that generate the scatterplot.

$\psi(u_d)$). A special case of Archimedean copulas corresponds to Frank copulas, which are obtained by setting: $\psi_\lambda(u) = \frac{-1}{\lambda}\log(1-(1-e^{-\lambda})e^{-u})$. When $\lambda \to 0$, the Frank copula approaches the independency copula; when $\lambda \to \infty$ it approaches the co-monotonicity copula. Hence, the Frank copula allows one to model all dependencies between complete independence to perfect dependence while varying λ from 0 to ∞. Therefore, λ can be seen as an additional hyper-parameter to be tuned or learned from the data. Figure 4.4 illustrates the positive dependence between two random variables sampled from a Frank copula with $\lambda = 25$. To sample from the Archimedean copulas, we rely on the algorithm proposed by [Marshall and Olkin, 1988], which was further improved in [McNeil, 2008, Hofert, 2011] and implemented in the R language [Hofert et al., 2011].

4.1.2 Extending LDA with copulas

As mentioned above, copulas provide a nice way to bind random variables. We are making use of them here to bind word-specific topics (the z variables in LDA) within coherent text spans, the rationale being that coherent text spans can not be generated by many different, uncorrelated topics. This leads us to the following generative model:

- For each topic $k \in [1, K]$, choose a per-word distribution: $\phi_k \sim Dir(\beta)$, with $\phi_k, \beta \in \mathbb{R}^{|V|}$
- For each document d_i, $i \in \{1, \ldots, D\}$:
  - Choose a per-document topic distribution: $\theta_i \sim Dir(\alpha)$, with $\theta_i, \alpha \in \mathbb{R}^{|K|}$
  - Sample number of segments in d_i: $S_i \sim Poisson(\xi)$;
  - For each segment $s_{i,j}, j \in \{1, \ldots, S_i\}$:
    - Sample number of words: $N_{i,j} \sim Poisson(\xi_d)$;
    - Sample topics $\mathscr{Z}_{i,j} = (z_{i,j,1}, \ldots, z_{i,j,N_{i,j}})$ from a distribution admitting $Mult(1, \theta_i)$ as margins and C as copula;
    - Sample words $W_{i,j} = (w_{i,j,1}, \ldots, w_{i,j,N_{i,j}})$: $w_{i,j,n} \sim Mult(1, \phi_{z_{i,j,n}})$, $1 \leq n \leq N_{i,j}$.

There are two main differences between copLDA and LDA. Firstly, the former assumes a hierarchical structure in the documents: the topics that generate the words in the coherent segments exhibit topical correlation, hence the conditional independence assumption between the terms of a segment given the document per-topic distribution (θ_i) no longer holds. Secondly, this topical correlation is modeled using copulas. Figure 4.5 provides the graphical model for copLDA. For clarity, we draw each word in a coherent segment S $(w1, \ldots, w_N)$ to make the dependencies explicit. Notice how the topics of those words depend on both the copula parameter λ and the per-document topic distribution θ.

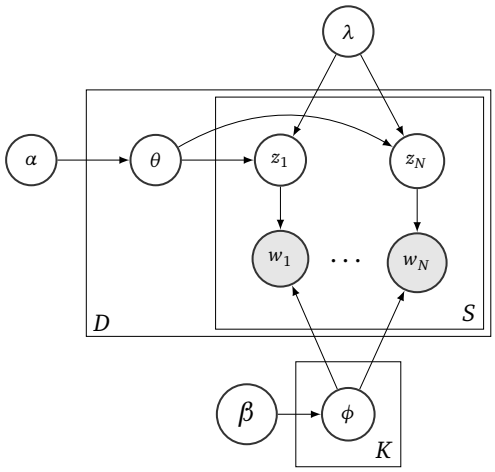


Figure 4.5: The copLDA generative model. We model the dependency between the topics underlying a segment with copulas.

The hyper-parameters α and β correspond to priors of the model. Following [Blei et al., 2003], we assume them here to be symmetric and we fix them to $\frac{1}{K}$, with K the number of topics retained. The hyper-parameter λ is chosen after exploration of a grid of possible values, and is the same for the whole corpus. We choose the value that minimizes perplexity.

4.1.3 Inference with Gibbs sampling for copLDA

The parameters of the above model, that are ϕ, θ and the topics of each segment $\mathscr{Z}_{i,j} = (z_{i,j,1}, \cdots, z_{i,j,N_{i,j}})$, can be directly estimated through Gibbs sampling. Denoting Ω and Ψ the count matrices such that $\Omega = (\Omega_{i,k})$ (resp. $\Psi = (\Psi_{k,v})$) represents the count of word belonging to topic k assigned to document d_i (resp. the count of word v being assigned to topic k), the Gibbs updates for θ and ϕ are the same as the ones for the standard LDA model [Blei et al., 2003]:

$$\theta_i \sim Dir(\alpha + \Omega_i) \quad \text{and} \quad \phi_k \sim Dir(\beta + \Psi_k) \tag{4.2}$$

The update for the variables z is obtained as follows:

$$p(\mathscr{Z}_{i,j}|\mathscr{Z}_{-i,j}, W, \Theta, \Phi, \alpha, \beta, \lambda) = \frac{p(\mathscr{Z}_{i,j}, \mathscr{Z}_{-i,j}, W|\Theta, \Phi, \alpha, \beta, \lambda)}{p(\mathscr{Z}_{-i,j}, W|\Theta, \phi, \alpha, \beta, \lambda)} =$$
$$\frac{p(\mathscr{Z}_{i,j}, W_{i,j}|\Theta, \Phi, \lambda)p(\mathscr{Z}_{-i,j}, W_{-i,j}|\Theta, \Phi, \lambda)}{p(W_{i,j}|\Theta, \phi)p(\mathscr{Z}_{-i,j}, W_{-i,j}|\Theta, \Phi, \lambda)} = \frac{p(\mathscr{Z}_{i,j}, W_{i,j}|\Theta, \Phi, \lambda)}{\sum_{\mathscr{Z}_{i,j}} p(\mathscr{Z}_{i,j}, W_{i,j}|\Theta, \Phi, \lambda)} =$$
$$\frac{p(W_{i,j}|\mathscr{Z}_{i,j}, \Phi)p(\mathscr{Z}_{i,j}|\Theta, \lambda)}{\sum_{\mathscr{Z}_{i,j}} p(W_{i,j}|\mathscr{Z}_{i,j}, \Phi)p(\mathscr{Z}_{i,j}|\Theta, \lambda)} \sim p(W_{i,j}|\mathscr{Z}_{i,j}, \Phi)p(\mathscr{Z}_{i,j}|\Theta, \lambda) = p(\mathscr{Z}_{i,j}|\Theta, \lambda)\prod_{n=1}^{N_{i,j}} \phi_{w_{i,j,n}, z_{i,j,n}} \tag{4.3}$$

where W, Θ and Φ stand for the whole parameter set of w, θ and ϕ and the probability outside the product in the last step admits a copula C_λ and $Mult(1, \theta_i)$ as margins. As is standard in topic models, the notation $-i, j$ means excluding the information for i, j. Note that in case where $\lambda \to 0$, the words of a segment become conditionally independent given the per-document distribution and one recovers the non collapsed Gibbs sampling updates of LDA.

From the expression of Eq. (4.3), a simple acceptance/rejection algorithm can be formulated: (1) Sample a random variable of pdf $p(\mathscr{Z}_{i,j}|\Theta, \lambda)$ using copula, and, (2) Accept the sample with probability $p(W_{i,j}|\mathscr{Z}_{i,j}, \Phi) = \prod_{n=1}^{N_{i,j}} \phi_{w_{i,j,n}, z_{i,j,n}}$. Algorithm 3 summarizes the inference process.

4.2 Computational considerations

As the values of $\phi_{w_{i,j,1},z_{i,j,1}} \times \cdots \times \phi_{w_{i,j,n},z_{i,j,n}}$ tend to be very low, the acceptance/rejection sampling step described above is very slow in practice (see below). We propose here to speed it up by considering, for each word $w_{i,j,n}$ in a given segment, not the exact probability of $z_{i,j,n}$, but its mean (noted M) over all the other words in the segment:

$$M(z_{i,j,n}|\mathscr{Z}_{-i,j}, W, \Theta, \Phi, \alpha, \beta, \lambda) = \sum_{w_{ij,l}, l \neq n} \sum_{z_{ij,l}, l \neq n} P(\mathscr{Z}_{i,j}|\mathscr{Z}_{-i,j}, W, \Theta, \Phi, \alpha, \beta, \lambda) \propto \phi_{w_{i,j,n}} \theta_{d,z_{i,j,n}}$$

as $\sum_{w_{ij,l}} \phi_{w_{i,j,l}} = 1$. Note that the above form is a marginalization of $P(\mathscr{Z}_{i,j}|\mathscr{Z}_{-i,j}, W, \Theta, \Phi, \alpha, \beta, \lambda)$ and thus defines a valid probability and a valid Gibbs sampler, even though on a joint distribution that slightly differs from the original one.

Algorithm 3: A Gibbs Sampling iteration for `copLDA`

Input: documents' words grouped in segments, α, β, K, Copula family and its parameter λ
//Initialize counters Ψ, Ω
for document $d_i, i \in [1, D]$ **do**
  for segment $s_{i,j} : j \in \{1, \ldots, S_i\}$ **do**
    Draw a random vector $U = (U_1, \ldots, U_{N_{i,j}})$ that admits a copula C_λ
    do
      /* If the mean approximation is used, the loop is done once, ignoring the acceptance condition */
      for words $w_{i,j,k}, k \in [1, W_{N_{i,j}}]$ in $s_{i,j}$ **do**
        Decrease counter variables Ψ, Ω
        Get $z_{i,j,k}$ by transforming U_k to Mult. marginals with the generalized inverse
        Assign topic $z_{i,j,k}$ to $w_{i,j,k}$
        Increase counters Ψ, Ω
    while Accept the new segment topic assignments with probability $\phi_{w_{i,j,1},z_{i,j,1}} \times \cdots \times \phi_{w_{i,j,n},z_{i,j,n}}$

Figure 4.6 compares the perplexity scores achieved in 200 documents from the Wikipedia dataset "Wiki46" of Table 4.1 by the `copLDA` model, when considering noun-phrases as coherent spans, with and without rejection sampling. We repeat the experiment 10 times and also plot the standard deviation. We first note that approximating Algorithm 1 by ignoring the rejection sampling step results in slightly worse performance. On the other hand, without the rejection sampling, `copLDA` converges faster in terms of iterations. Furthermore, the cost in terms of running time of a single iteration is significantly smaller: for instance, for 30 iterations with rejection sampling, the algorithm needs almost 6 hours, that is 100 times more than the 3.5 minutes needed without the rejection sampling. Hence, in the rest of the study, for scaling purposes, we adopt the above mean approximation.

4.3 Experimental study

Models In our experiments, we compare the following topic models: (1) $\texttt{copLDA}_{\texttt{sen}}$ that considers sentences as coherent segments, (2) $\texttt{copLDA}_{\texttt{np}}$ that considers noun-phrases as coherent segments, (3) `LDA` as proposed in [Blei et al., 2003] using the collapsed Gibbs sampling inference of [Griffiths and Steyvers, 2004], and (4) `senLDA` described in [Balikas et al., 2016a] using its public implementation. For $\texttt{copLDA}_{\texttt{x}}$ models, we use the Frank copula which was reported to obtain the best performance in similar tasks [Amoualian et al., 2016] and was also found to achieve the best performance in our local validation settings compared to Gumbel and Clayton copulas. We have implemented the models using Python;[2] for sampling the Frank copulas we used the R `copula` package [Hofert et al., 2011] and `rPY`.[3] As mentioned in Section 4.1.2, λ is set to 2 for $\texttt{copLDA}_{\texttt{sen}}$ and to 5 for $\texttt{copLDA}_{\texttt{np}}$ (values which we found to perform well in every dataset we tried). Furthermore, the hyper-parameters α and β where set to $1/K$, where K is the number of topics, which was selected from $\{50, 100, 200, 300, 400\}$ for each dataset. For the shallow parsing step, required for $\texttt{copLDA}_{\texttt{np}}$, we used the Stanford Parser [Klein and Manning, 2003]. The text pre-processing steps performed are: lower-casing, stemming using the Snowball Stemmer and removal of numeric strings.

Datasets We have used the following publicly available data collections to test the performance of the topic models: (1) 20NG (20 news groups), which is a standard text dataset for such tasks as provided by [Bird et al., 2009], (2) Reuters (Reuters-21578, the "ModApte" version), also discussed in [Bird et al., 2009], (3) TED, that is transcriptions of TED talks released in the framework of the International Workshop on Spoken Language Translation 2013 evaluation campaign[4] (we have merged the train, development and test parts and we selected the transcriptions with at least one associated label among the 15 most common in the data[5]), (4) Wiki_x, with $x \in \{15, 37, 46\}$ and PubMed, both excerpts[6] from the Wikipedia dataset of [Partalas et al., 2015] and the PubMed dataset of [Tsatsaronis et al., 2015] used in [Balikas et al., 2016a], and (5) "Austen", where we concatenated three

[2]The models used in this chapter are available for research purposes at `https://github.com/balikasg/topicModelling`.

[3]https://pypi.python.org/pypi/rpy2

[4]`http://workshop2013.iwslt.org/59.php`

[5]Technology, Culture, Science, Global Issues, Design, Business, Entertainment, Arts, Politics, Education, Art, Creativity, Health, Biology and Music.

[6]`https://github.com/balikasg/topicModelling/tree/master/data`

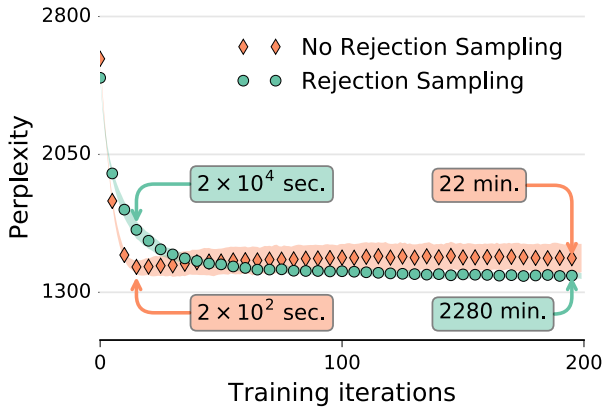


Figure 4.6: The effect of rejection sampling in efficiency and perplexity performance.

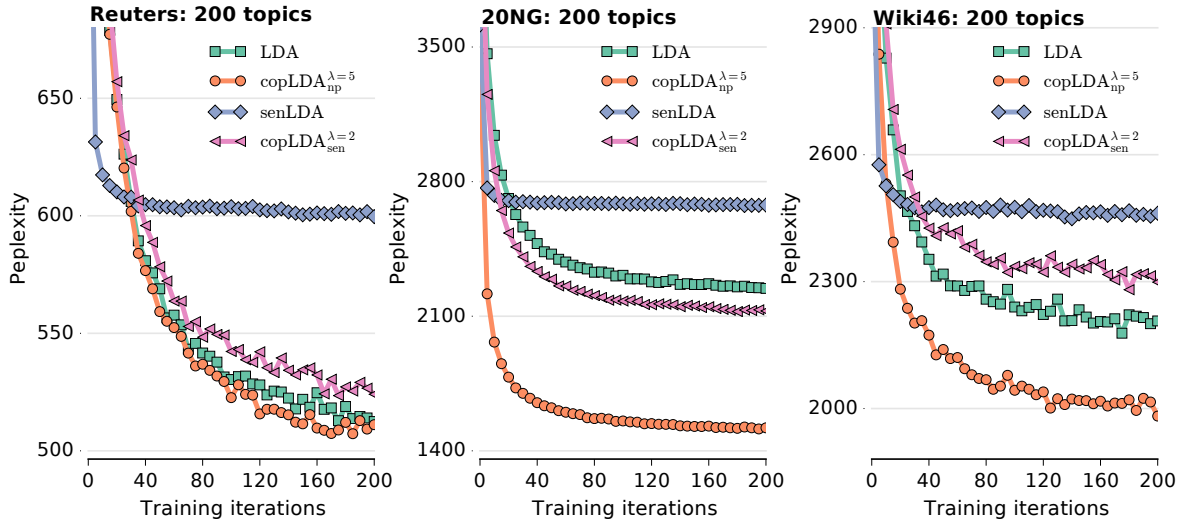


Figure 4.7: The perplexity curves of the investigated models for 200 Gibbs sampling iterations and different datasets.

books[7] written by Jane Austen, available from the Gutenberg project (each paragraph is considered as a document). Table 4.1 presents some basic statistics for these datasets.

[7]We used the books: Emma, Persuasion, Sense. We considered each paragraph as a document.

Table 4.1: The basic statistics, the perplexity and the classification scores of the datasets used.

	Basic Statistics				Perplexity Scores				Classification (MiF_1) scores			
	Docs.	$\|N\|$	$\|V\|$	Classes	senLDA	copLDA$_{sen}$	LDA	copLDA$_{np}$	senLDA	copLDA$_{sen}$	LDA	copLDA$_{np}$
20NG	19,056	1.7M	75.4K	20	2636	2083	2200	**1483**	0.5622	0.6328	0.6246	**0.6490**
TED	1,096	1.16M	30.4K	15	2099	1812	1805	**1775**	0.4612	0.4678	0.4633	**0.4764**
PubMed	5498	1.09M	28.7K	50	1601	1385	1384	**1085**	0.6666	**0.7525**	0.7406	0.7431
Reuters	10,788	875K	21.4K	90	579	512	501	**499**	0.7504	0.7692	**0.7893**	0.7851
Wiki15	1,198	162K	13.4K	15	2988	2766	2640	**2397**	0.6920	0.7230	0.74	**0.7403**
Wiki37	2,459	317K	19.7K	37	3103	2871	2711	**2395**	0.5717	0.6053	**0.6447**	0.6220
Wiki46	3,657	478K	23.4K	46	2220	2280	2135	**1978**	0.5326	0.6170	**0.6599**	0.6326
Austen	5,262	170K	6.3K	-	1110	898	**798**	805	-	-	-	-

Manual inspection of the topics We begin by comparing LDA and copLDA$_{np}$. For presentation purposes, we train the two topic models using the Wiki$_{47}$ dataset with 10 topics and we illustrate the top-10 words learned for each topic by the two models in Table 4.2. As one can note, since the two models have been trained on the same data with the same training parameters, the identified topics are very similar. This said, copLDA$_{np}$ manages to produce arguably better topics. This is for example the case for the topic "Birth"; although both models assign high probability to words like "born" and "american" due to the content of the dataset, copLDA$_{np}$ manages to identify several words corresponding to months which makes the topic more thematically consistent and easier to interpret compared to its LDA counterpart. In the same line, Table 4.3 visualizes the inferred topics for parts of the Wiki$_{47}$ dataset. Notice here that given the topic interpretations of Table 4.2, both models manage to identify intuitive topics. Note however how in most of the cases the text structure information used by copLDA$_{np}$ helps to obtain consistent topics to generate noun-phrases like "crime thriller film" and "raspy voice", a consistency that LDA is lacking.

Intrinsic evaluation: perplexity We present in Table 4.1 the perplexity scores achieved by the 4 models in each of the datasets we examined. We split each dataset in two parts with 80%/20% of the documents: we use the former for learning the model and the second for calculating the perplexity scores. First note that copLDA$_{np}$ achieves the lowest scores in most of the datasets. LDA is the second best performing model, whereas the third one is copLDA$_{sen}$. We believe that the difference between copLDA$_{sen}$ and copLDA$_{np}$ stems from the fact that perplexity is an evaluation measure that is calculated on the basis of words. Hence, considering sentences as coherent spans whose topics are bound results in

Table 4.2: The top-10 words of `copLDA` (upper half) and `LDA` (lower half) in the Wiki46 dataset.

Profession	Science	Books	Art	Cinema	Places	Music	Birth	Elections	Inventions
profession	univers	book	art	film	state	record	born	elect	california
world	research	new	new	televis	unit	music	american	canadian	plant
footbal	scienc	work	work	role	us	band	known	parti	use
wrestl	professor	american	paint	appear	township	album	best	member	invent
play	work	publish	york	also	school	song	actress	liber	flower
born	institut	time	american	actor	univers	also	decemb	minist	compani
american	award	author	artist	born	serv	produc	june	hous	north
championship	prize	also	museum	play	war	releas	april	canada	patent
team	born	year	painter	seri	nation	new	juli	serv	inventor
first	receiv	york	studi	star	build	singer	januari	conserv	found
known	univers	book	art	film	township	record	play	elect	work
wrestl	research	new	new	born	state	music	footbal	canadian	first
born	scienc	american	york	televis	counti	band	born	serv	year
world	professor	author	paint	role	us	album	american	parti	photograph
profession	work	publish	american	actor	california	song	tour	member	design
american	institut	novel	work	appear	michigan	also	golf	liber	state
name	born	time	artist	also	plant	singer	year	hous	new
wrestler	prize	also	painter	seri	civil	releas	profession	minist	use
best	studi	writer	museum	actress	popul	produc	first	state	also
championship	award	magazin	born	american	flower	american	season	born	build

Table 4.3: The discovered topics underlying the words of example documents for `LDA` (left) and `copLDA` (right). The parts of the documents in italics indicate the noun-phrases obtained by the Stanford Parser. The text colours refer to the topics described in Table 4.2.

Kiss of Death is *a 1995 crime thriller film* starring *David Caruso Samuel L. Jackson* and *Nicolas Cage*. *The film* is a *very loosely* based remake of *the 1947 film noir classic* of *the same name* that starred *Victor Mature*, *Brian Donlevy* and *Richard Widmark*.

Bertram Stern (born 3 *October 1929*) is *an American fashion and celebrity portrait photographer*.

Dana Hill (born *Dana Lynne Goetz* in *Los Angeles*, California; *May 6, 1964 - July 15, 1996*) was *an American actress and voice actor* with *a raspy voice and childlike appearance*, which allowed her to play *adolescent roles* well into *her 20s*.

Kiss of Death is *a 1995 crime thriller film* starring *David Caruso Samuel L. Jackson* and *Nicolas Cage*. *The film* is a *very loosely* based remake of *the 1947 film noir classic* of *the same name* that starred *Victor Mature*, *Brian Donlevy* and *Richard Widmark*.

Bertram Stern (born 3 *October 1929*) is *an American fashion and celebrity portrait photographer*.

Dana Hill (born *Dana Lynne Goetz* in *Los Angeles*, California; *May 6, 1964 - July 15, 1996*) was *an American actress and voice actor* with *a raspy voice and childlike appearance*, which allowed her to play *adolescent roles* well into *her 20s*.

less flexibility and this is reflected in higher perplexity scores. However, using copulas results in more flexibility than assigning the same topic in each term of the sentence which is illustrated in the performance difference between $\texttt{copLDA}_{\text{sen}}$ and senLDA. The former being more flexible, due to the copulas, performs better. In the same line, Figure 4.7 illustrates the perplexity curves of the hold-out documents for the four models on three of the datasets of Table 4.1 for 200 Gibbs sampling iterations. Note that senLDA is the model with the fastest convergence rate with respect to the number of Gibbs iterations. On the other hand, LDA, $\texttt{copLDA}_{\text{sen}}$ and $\texttt{copLDA}_{\text{np}}$ require the same number of iterations, which depends on the dataset. $\texttt{copLDA}_{\text{np}}$ manages to achieve the lowest perplexity scores: notice its steep curves in the first iterations.

Extrinsic evaluation: text classification To further highlight the merits of copLDA, we also present in Table 4.1 the classification results for the datasets used. The reported scores are the averages of 10-fold cross-validation. We use the per-document topic distributions as classification features fed to Support Vectors Machines (SVMs). We have used the implementation of [Pedregosa et al., 2011] with $C = 1$ for the SVM regularization parameter. For the multi-label datasets (TED and PubMed) we employed one-versus-rest: the SVMsreturn every category with a positive distance from the separating hyper-planes. As one can note, $\texttt{copLDA}_{\text{np}}$ and LDA achieve the highest MiF scores in most of the datasets, without a clear advantage to one vs the other. Binding the topics of sentence words with copulas improves over the results of senLDA: $\texttt{copLDA}_{\text{sen}}$ performs only slightly worse than LDA and $\texttt{copLDA}_{\text{np}}$ on most datasets and outperforms them, only slightly again, on one dataset.

4.4 Summary

In this chapter, we proposed copLDA that extends LDA to incorporate the topical dependencies within sentences and noun-phrases using copulas. We have shown empirically the advantages of considering text structure and incorporating it in LDA with copulas. In our future work we plan to integrate procedures to learn the λ parameter of Frank copulas and to investigate ways to model not only dependencies within text segments like noun-phrases, but also dependencies between such segments with nested copulas.

Chapter 5

Topical coherence in LDA-based models through induced segmentation

Since the seminal works of [Hofmann, 1999] and [Blei et al., 2003], there have been several developments in probabilistic topic models. Many extensions have indeed been proposed for different applications, including ad-hoc information retrieval [Wei and Croft, 2006], clustering search results [Zeng et al., 2004] and driving faceted browsing [Mimno and McCallum, 2007]. However, the majority of these studies follow the initial exchangeability assumption of `pLSI` and `LDA`, stipulating that words within a document are interdependent. In most of these studies, the initial exchangeability assumptions of `PLSA` and `LDA`, stipulating that words within a document are interdependent, has led to incoherent topic assignments within semantically meaningful text units, even though the importance of having topically coherent phrases is generally admitted [Griffiths et al., 2005]. More recently, [Balikas et al., 2016b] has shown that binding topics, so as to obtain more coherent topic assignments, within such text segments as noun phrases improves the performance (*e.g.* in terms of perplexity) of `LDA`-based models. The question nevertheless remains as to which segmentation one should rely on.

Furthermore, text segments can refer to topics that are barely present in other parts of the document. For example, the segment "*the Kurdish regional capital*" in the sentence[1] "*A thousand protesters took to the main street in Erbil, the Kurdish regional capital, to condemn a new law requiring all public demonstrations to have government permits.*" refers to geography in a document that is mainly devoted to politics. Relying on a single

[1] This sentence is taken from New York Times news (NYT) collection described in Section 5.3.

topic distribution, as done in most previous studies including [Balikas et al., 2016b], may prevent one from capturing those segment specific topics.

Furthermore, recent studies have pointed out that, *perplexity*, the generally accepted measure to evaluate the performance of topic models cannot capture the coherence in topic assignments and proposed other alternative measures, such as the Normalized Pointwise Mutual Information (NPMI) [Mimno et al., 2011], as accurately modeling and capturing such units can be crucial for down-stream NLP tasks, and for many case studies involving for example the visualization of results, the importance of having topically coherent phrases is generally admitted [Griffiths et al., 2005].

Text units such as documents, sentences, phrases, segments and even chunks can be related in the content. Therefore, as we have discussed, a topic model that is capable to integrate these structures for generating a context, can be more accurate and natural in terms of parameter estimation. This language model will become more realistic if it follows a flexible and controllable way to incorporate these dependent structures for discovering the latent topics. Also topic model can generate various level of a text division simultaneously. Intuitively applying a method to cohere the topic of each unit and assigns the same topic for more words in each level, makes model closer to the ideal. Recently many researches have been proposing different binding tehcnices for capturing dependency within a text ([Blei and Lafferty, 2006] for document level, [Du et al., 2010a] for segment level, [Balikas et al., 2016b] for chunks level) but they still suffer from the lack of having different level of cohesion at the same time.

In this chapter, we propose a novel `LDA`-based model that automatically segments documents into topically coherent sequences of words, while relying on both document and segment specific topic distributions so as to capture fine grained differences in topic assignment to words [Amoualian et al., 2017]. The coherence between topics is ensured through *copulas* [Elidan, 2013] that bind the topics associated to the words of a segment. In addition, this model relies on both document and segment specific topic distributions so as to capture fine grained differences in topic assignments. A simple switching mechanism is used to select the appropriate distribution (document or segment specific) for assigning a topic to a word. We show that this model naturally encompasses other state-of-the-art `LDA`-based models proposed to accomplish the same task, and that it outperforms these models over six publicly available collections in terms of perplexity, Normalized Pointwise Mutual Information (NPMI), a measure used to assess the coherence of topics with documents, and the Micro F1-measure in a text classification context.

This chapter is structured as follows: In Sections 5.1.1 and 5.1.2 we present the

models accompanying Gibbs Sampling inference based on the incorporation of copula for chaining the topics of the words within each segment that we estimate jointly with our generative model using an efficient segmentation way. Section 5.3 exposes the competence of the model intrinsically and extrinsically using distant metrics (perplexity, classification accuracy, topic coherence and visualization) compared with different seminal topic models. We apply 6 kinds of well-known collection for topic modeling having various properties (different amount of vocabulary, words and documents, labeled and unlabeled) to evaluate the ability of this method in the different setting of experiment. Eventually, in Section 5.4 we conclude our methods and illustrate the main clues for the future contributions.

5.1 Joint latent model for topics and segments

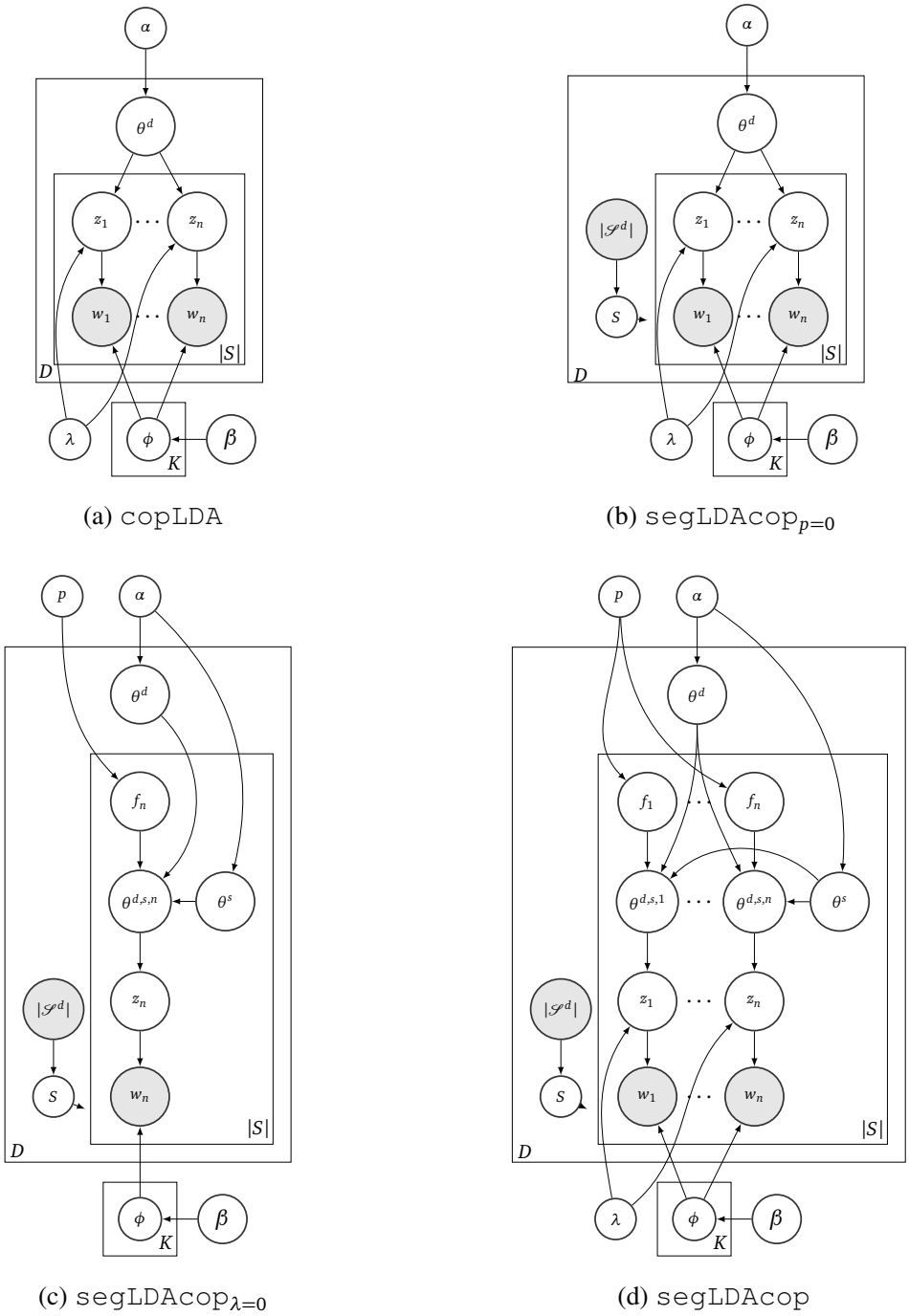
We define here a *segment* as a topically coherent sequence of contiguous words. By topically coherent, we mean that, even though words in a segment can be associated to different topics, these topics are usually related. This view is in line with the one expressed in [Balikas et al., 2016b], in which a latent topic model, referred to as `copLDA` in the remainder, includes a binding mechanism between topics within coherent text spans, defined in their study as noun phrases (NPs). The relation between topics is captured through a copula that provides a joint probability for all the topics used in a segment. That is, to generate words in a segment, one first jointly generates all the word specific topics z via a copula, and then generates each word in the segment from its word specific topic and the word-topic distribution ϕ. Figure 5.1(a) illustrates this.

Following what we discussed in Section 3.2.1, Copulas are particularly useful when modeling dependencies between random variables, as the joint cumulative distribution function (CDF) $F_{X_1,\cdots,X_n}$ of any random vector $\mathbf{X} = (X_1, \cdots, X_n)$ can be written as a function of its marginals, according to Sklar's Theorem [Nelsen, 2007]:

$$F_{X_1,\cdots,X_n}(x_1, \cdots, x_p) = C(F_{X_1}(x_1), \cdots, F_{X_n}(x_n))$$

where C is a copula. For latent topic models, as discussed in Chapter 3 and [Amoualian et al., 2016], Frank's copula is particularly interesting as (a) it is invariant by permutations and associative, as are the words and topics z in each segment due to the exchangeability assumption, and (b) it relies on a single parameter (denoted λ here) that controls the strength of dependence between the variables and is thus easy to implement. In Frank's copula, when the parameter λ approaches 0, the variables are independent of each other, whereas when λ approaches $+\infty$, the variables take the same value. For further details on copulas, we refer the reader to [Nelsen, 2007].

One important problem, however, with `copLDA` is its reliance on a predefined segmentation. Although the information brought by the segmentation based on NPs helps to improve topic assignment, it may not be flexible enough to capture all the possible segments of a text. It is easy to correct this problem by considering all possible segmentations of a document and by choosing the most appropriate one at the same time that topics are assigned to words. This is illustrated in Figure 5.1(b), where a segmentation S is chosen from the set $\mathscr{S}^d$ of possible segmentations for a document d, and where each segment in S are generated in turn. We refer to the associated model as $\texttt{segLDAcop}_{p=0}$ for reasons



(a) `copLDA` (b) $\texttt{segLDAcop}_{p=0}$

(c) $\texttt{segLDAcop}_{\lambda=0}$ (d) `segLDAcop`

Figure 5.1: Graphical model for Copula `LDA` (`copLDA`), extension of Copula `LDA` with segmentation ($\texttt{segLDAcop}_{p=0}$), `LDA` with segmentation and topic shift ($\texttt{segLDAcop}_{\lambda=0}$) and complete model (`segLDAcop`).

that will become clear later.

Another point to be noted about `copLDA` (and $\texttt{segLDAcop}_{p=0}$) is that the topics used in each segment come from the same document specific topic distribution θ^d. This

entails that, in these models, one cannot differentiate the main topics of a document from potential segment specific topics that can explain some parts of it. Indeed, some text segments can refer to topics that are barely present in other parts of the document; relying on a single topic distribution may prevent one from capturing those segment specific topics.

It is possible to overcome this difficulty by generating a segment specific topic distribution as illustrated in Figure 5.1(c) (this model is referred to as $\texttt{segLDAcop}_{\lambda=0}$, again for reasons that will become clear later). However, as some words in a segment can be associated to the general topics of a document, we introduce a mechanism to choose, for each word in a segment, a topic either from the segment specific topic distribution θ^s or from the document specific topic distribution θ^d (this mechanism is similar to the one used for routes and levels in [Paul and Girju, 2010]). The choice between them is based on the Bernoulli variable f, as explained in the generative story given below.

The above developments can be combined in a single, complete model, illustrated in Figure 5.1(d) and detailed below. We will simply refer to this model as `segLDAcop`.

5.1.1 Complete generative model

As in standard LDA based models, with V denoting the size of the vocabulary of the collection and K the number of latent topics, β and ϕ^k, $1 \leq k \leq K$, are V dimensional vectors, α and θ (*i.e.*, $\theta^d, \theta^s, \theta^{d,s,n}$) are K dimensional vectors, whereas z_n takes value in $\{1, \cdots, K\}$. Lower indices are used to denote coordinates of the above vectors. Lastly, Dir denotes the Dirichlet distribution, Cat the categorical distribution (which is a multinomial distribution with one draw) and we omit, as is usual, the generation of the length of the document. The complete model segLDAcop is then based on the following generative process:

1. Generate, for each topic $k, 1 \leq k \leq K$, a distribution over the words: $\phi^k \sim Dir(\beta)$;
2. For each document $d, 1 \leq d \leq D$:
   (a) Choose a document specific topic distribution: $\theta^d \sim Dir(\alpha)$;
   (b) Choose a segmentation S of the document uniformly from the set of all possible segmentations $\mathscr{S}^d$: $P(S) = \frac{1}{|\mathscr{S}^d|}$;
   (c) For each segment s in S:
      (i) Choose a segment specific topic distribution: $\theta^s \sim Dir(\alpha)$;
      (ii) For each position n in s, choose $f_n \sim Ber(p)$ and set:

$$\theta^{d,s,n} = \begin{cases} \theta^s & \text{if } f_n = 1 \\ \theta^d & \text{otherwise} \end{cases}$$

      (iii) Choose topics $\mathscr{Z}^s = \{z_1, \ldots, z_n\}$ from Frank's copula with parameter λ and marginals $Cat(\theta^{d,s,n})$;
      (iv) For each position n in s, choose word w_n: $w_n \sim Cat(\phi^{z_n})$.

As on can note, the generative process relies on a segmentation uniformly chosen from the set of possible segmentations (step *2.b*) to generate related topics within each segment (Frank's copula in step *2.c.(iii)*), the distribution underlying each word specific topic z_n being either specific to the segment or general to the document (steps *2.c.(i)* and *2.c.(ii)*). The other steps are similar to the standard LDA steps.

As in almost all previous studies on LDA, α and β are considered fixed and symmetric, each coordinate of the vector being equal: $\alpha_1 = \cdots = \alpha_K$. The hyperparameters p ($\in [0, 1]$) of the Bernoulli distribution and λ ($\in [0, +\infty]$) of Frank's copula respectively regulate the

choice between the segment specific and the document specific topic distributions and the strength of the dependence between topics in a segment. As for the other hyperparameters, we consider them fixed here (the values for all hyperparameters are given in Section 5.3).

As mentioned before, all the models presented in Figure 5.1 are special cases of the complete model segLDAcop: hence $\texttt{segLDAcop}_{\lambda=0}$ is obtained by dropping the topic dependencies, which amounts to setting λ to (a value close to) 0, $\texttt{segLDAcop}_{p=0}$ is obtained by relying only on the topic distribution obtained for the document, which amounts to setting p to 0, and the previously introduced copLDA model is obtained by setting p to 0, and fixing the segmentation.

5.1.2 Inference with gibbs sampling for segLDAcop

The parameters of the complete model can be directly estimated through Gibbs sampling. The Gibbs updates for the parameters ϕ and θ are the same as the ones for standard LDA [Blei et al., 2003]. The parameters f_n are directly estimated through: $f_n \sim Ber(p)$. Lastly, for the variables z, we follow the same strategy as the one described in [Balikas et al., 2016b] and based on [Amoualian et al., 2016], leading to:

$$P(\mathcal{Z}^s|\mathcal{Z}^{-s}, W, \Theta, \Phi, \lambda) = p(\mathcal{Z}^s|\Theta, \lambda)\prod_n \phi_{w_n}^{z_n}$$

where W denotes the document collection, and Θ and Φ the sets of all θ and ϕ^k, $1 \leq k \leq K$, vectors. $p(\mathcal{Z}^s|\Theta, \lambda)$ is obtained by Frank's copula with parameter λ and marginals $Cat(\theta^{d,s,n})$. As is standard in topic models, the notation $-s$ means excluding the information from s.

From the above equation, one can formulate an acceptance/rejection algorithm based on the following steps: (a) sample $\mathcal{Z}^s$ from $p(\mathcal{Z}^s|\Theta, \lambda)$ using Frank's copula, and (b) accept the sample with probability $\prod_n \phi_{w_n}^{z_n}$, where n runs over all the positions in segment s.

5.2 Efficient segmentation

As topics may change from one sentence to another, we assume here that segments cannot overlap sentence boundaries. The different segmentations of a document are thus based on its sentence segmentations. In the remainder, we use L to denote the maximum length of a segment and $g(M;L)$ to denote the number of segmentations in a sentence of length M, each segment comprising at most L words.

Generating all possible segmentations of a sentence and then selecting one at random is not an efficient process as the number of segments rapidly grows with the length of the sentence. In practice, however, one can define an efficient segmentation on the basis of the following proposition, the proof of which is given in Appendix A.4:

Proposition 5.2.1 *Let l_i^s be the random variable associated to the length of the segment starting at position i in a sentence of length M (positions go from 1 to M and l_i^s takes value in $\{1,\cdots,L\}$). Then $P(l_i^s = l) := \frac{g(M+1-i-l);L)}{g(M+1-i;L)}$ defines a probability distribution over l_i^s.*

Furthermore, the following process is equivalent to choosing sentence segmentations uniformly from the set of possible segmentations.

```
From pos. 1, repeat till end of sentence:
(a) Generate segment length acc. to P;
(b) Add segment to current segmentation;
(c) Move to position after the segment.
```

In practice, we thus replace steps *2.b* and *2.c* of the generative story by a loop over all sentences, and in each sentence use the process described in Prop, 5.2.1. Furthermore, as described in Appendix A.4, the values of g needed to compute $P(l_i^s = l)$ can be efficiently computed by recurrence.

5.3 Experimental study

We conducted a number of experiments aimed at studying the impact of simultaneously segmenting and assigning topics to words within segments using the proposed segLDAcop model.

Datasets: We considered six publicly available datasets derived from Pubmed[2] [Tsatsaronis et al., 2015], Wikipedia [Partalas et al., 2015], Reuters[3] and New York Times (NYT)[4] [Yao et al., 2016]. The first two collections were considered in [Balikas et al., 2016a], we followed their setup by considering 3 subsets of Wikipedia with different number of classes (namely, Wiki0, Wiki1 and Wiki2). The Reuters dataset comes from Reuters-21578, Distribution 1.0 as investigated in [Bird et al., 2009] and the NYT dataset is collected from full text of New York Times global news, from January 1st to December 31st, 2011.

These collections were processed following [Blei et al., 2003] by removing a standard list of 50 stop words, lemmatizing, lowercasing and keeping only words made of letters. To deal with relatively homogeneous collections, we also removed documents that are too long. The statistics of these datasets, as well as the admissible maximal length for documents, in terms of the number of words they contain, can be found in Table 5.1.

Settings: We compared our models (segLDAcop$_{p=0}$, segLDAcop$_{\lambda=0}$, segLDAcop) with three models, namely the standard LDA model, and two previously introduced models aiming at binding topics within segments:

1. LDA: Standard Latent Dirichlet Allocation implemented using collapsed Gibbs sampling inference [Griffiths and Steyvers, 2004][5]. Note that there are neither segmentation nor topic binding mechanisms in this model;
2. senLDA: Sentence LDA, introduced in [Balikas et al., 2016a], which forces all words within a sentence to be assigned to the same topic. The segments considered thus correspond to sentences, and the binding between topics within segments is maximal as all word specific topics are equal;
3. copLDA: Copula LDA, introduced in [Balikas et al., 2016b] already discussed before, which relies on two types of segments, namely NPs (extracted with the

[2] https://github.com/balikasg/topicModelling/tree/master/data

[3] https://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection

[4] https://github.com/yao8839836/COT/tree/master/data

[5] http://gibbslda.sourceforge.net

Table 5.1: Dataset statistics.

	Wiki0	Wiki1	Wiki2
# words	32,354	70,954	103,308
– *vocabulary size*	7,853	12,689	14,715
# docs	1,014	2,138	3,152
– *maximal length*	100	100	100
# labels	17	42	53
	Pubmed	Reuters	NYT
# words	104,683	192,562	237,046
– *vocabulary size*	12,779	10,479	17,773
# docs	2,059	6,708	2,564
– *maximal length*	75	50	200
# labels	50	83	-

`nltk.chunk` package [Bird et al., 2009]) and single words. In addition, a copula is also used to bind topics within NPs, from the document specific topic distribution.

Both `senLDA` and `copLDA` implementations, can be found in `https://github.com/balikasg/topicModelling`.

In all models α and β play a symmetric role and are respectively fixed to $1/K$, following [Asuncion et al., 2009]. For copula based models, λ is set to 5, following [Balikas et al., 2016b]. As already discussed, p is set to 0 for $\texttt{segLDAcop}_{p=0}$; it is set to 0.5 for `segLDAcop` so as not to privilege *a priori* one topic distribution (document or segment specific) over the other. For sampling from Frank's copula, we relied on the R copula package [Hofert et al., 2011]. We chose L (the maximum length of a segment) using line search for $L \in [2, 5]$ and used $L = 3$ in all our experiments. Finally, to illustrate the behaviors of the different models with different number of topics, we present here the results obtained with $K = 20$ and $K = 100$.

We now compare the different models along three main dimensions: perplexity, use of topic representations for classification and topic coherence.

5.3.1 Perplexity results

Table 5.2: Perplexity with respect to different number of topics (20 and 100).

Models	Wiki0		Wiki1		Wiki2		Pubmed		Reuters		NYT	
	20	100	20	100	20	100	20	100	20	100	20	100
LDA	853.7	370.9	1144.6	541.1	1225.2	570.6	1267.8	628.7	210.6	118.8	1600.1	1172.1
senLDA	958.4	420.5	1236.7	675.3	1253.1	625.2	1346.3	674.3	254.3	173.6	1735.9	1215.3
copLDA	753.1	264.3	954.1	411.5	1028.6	420.6	1031.5	483.2	206.3	101.3	1551.5	1063.2
segLDAcop$_{p=0}$	670.2	235.4	904.2	382.4	975.7	409.2	985.5	459.3	194.2	96.7	1504.2	1033.2
segLDAcop$_{\lambda=0}$	655.1	222.1	890.3	370.2	949.2	404.3	971.3	451.2	190.1	91.3	1474.6	1014.3
segLDAcop	**621.2**	**213.5**	**861.2**	**358.6**	**934.7**	**394.4**	**960.4**	**442.1**	**182.1**	**87.5**	**1424.2**	**992.3**

We first randomly split here all the collections, using 75% of them for training, and 25% for testing.

In order to see how well the models fit the data and following [Blei et al., 2003], we first evaluated the methods in terms of perplexity again defined as:

$$Perplexity = \exp\left(\frac{-\sum_{d\in D}\sum_{w\in d}\log\sum_{k=1}^{K}\theta_k^d\phi_w^k}{\sum_{d\in D}|d|}\right),$$

where d is a test document from the test set D, and $|d|$ is the total number of words in d, and K is the total number of topics. The lower the perplexity is, the better the model fits the test data. Table 5.2 shows perplexities of different methods for $K = 20$ and $K = 100$ topics.

From Table 5.2, it comes out that the best performing model in terms of perplexity over all datasets and for different number of topics is segLDAcop. Further, segLDAcop$_{\lambda=0}$, that uses both document and segment specific topic distributions, performs better than segLDAcop$_{p=0}$, which in turn outperforms copLDA, bringing evidence that using all possible segmentations rather than only NPs unit extracted using a chunker yields a more flexible and natural topic assignment.

segLDAcop also converges faster than the other methods to its minimum as it is shown in Figure 5.2, depicting the evolution of perplexity of different models over the number of iterations on the NYT collection (a similar behavior is observed on the other collections).

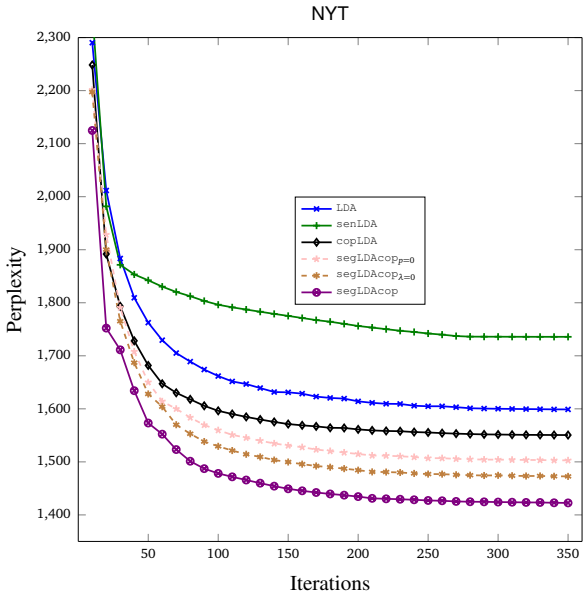


Figure 5.2: Perplexity with respect to training iteration on NYT collection (20 topics).

5.3.2 Topical induced representation for classification

Some studies compare topic models using extrinsic tasks such as document classification. In this case, it is possible to reduce the dimensionality of the representation space by using the induced topics [Blei et al., 2003]. In this study, we first randomly splitted the datasets, except NYT that does not contain class information, into training (75%) and test (25%) sets. We then applied `SVMs`with a linear kernel; the value of the hyperparameter C was found by cross-validation over the training set $\{0.01, 0.1, 1, 10, 100\}$. For datasets where certain documents have more than one label (Pubmed, Reuters), we used the one-versus-all[6] approach for performing multi-label classification.

In Table 5.3, we report the Micro F1 (MiF) score of different models on the test sets. Again, the best results are obtained with `segLDAcop`, followed by `segLDAcop`$_{\lambda=0}$. This shows the importance of relying on both document and segment specific topic distributions. As conjectured before, our model is able to captures fine grained topic assignments within documents. In addition, all models relying on an inferred segmentation (`segLDAcop`$_{p=0}$, `segLDAcop`$_{\lambda=0}$, `segLDAcop`) outperform the models relying on fixed segmentations (sentences or NPs). This shows the importance of being able to discover flexible segmentations for assigning topics within documents.

[6]class sklearn.multiclass.OneVsRestClassifier

Table 5.3: MiF score (percent) with respect to different number of topics (20 and 100).

Models	Wiki0		Wiki1		Wiki2		Pubmed		Reuters	
	20	100	20	100	20	100	20	100	20	100
LDA	55.3	63.5	42.4	51.4	41.2	48.7	54.1	63.5	75.5	82.7
senLDA	41.4	53.2	33.5	44.5	36.4	40.9	50.2	62.5	69.4	74.2
copLDA	51.2	62.7	43.4	52.1	40.8	46.5	53.5	63.1	75.2	81.5
segLDAcop$_{p=0}$	59.1	64.2	44.8	51.2	42.3	50.1	55.4	63.1	76.8	82.5
segLDAcop$_{\lambda=0}$	61.1	67.4	46.5	53.8	44.1	52.2	57.1	65.2	79.6	84.4
segLDAcop	**62.3**	**68.4**	**48.4**	**55.2**	**44.8**	**53.5**	**59.3**	**66.5**	**80.2**	**85.1**

5.3.3 Topic coherence

Another common way to evaluate topic models is by examining how coherent the produced topics are. Doing this manually is a time consuming process and cannot scale. To overcome this limitation the task of automatically evaluating the coherence of topics produced by topic models received a lot of attention [Mimno et al., 2011]. It has been found that scoring the topics using co-occurrence measures, such as the pointwise mutual information (PMI) between the top-words of a topic, correlates well with human judgments [Newman et al., 2010]. For this purpose an external, large corpus is used as a meta-document where the PMI scores of pairs of words are estimated using a sliding window.

$$PMI(w_i, w_j) = \log \frac{P(w_i, w_j)}{P(w_i)P(w_j)}$$

As discussed above, calculating the co-occurrence measures requires selecting the top-N words of a topic and performing the manual or automatic evaluation. Hence, N is a hyper-parameter to be chosen and its value can impact the results. Very recently, [Lau and Baldwin, 2016] showed that N actually impacts the quality of the obtained results and, in particular, the correlation with human judgments. In their work, they found that aggregating the topic coherence scores over several topic cardinalities leads to a substantially more stable and robust evaluation.

Following the findings of [Lau and Baldwin, 2016] and using [Newman et al., 2010]'s equation, we present in Figure 5.3 the topic coherence scores as measured by the Normalized Pointwise Mutual Information (NPMI) . Their values are in [-1,1], where in the limit of -1 two words w_1 and w_2 never occur together, while in the limit of +1 they always occur

together (complete co-occurrence).

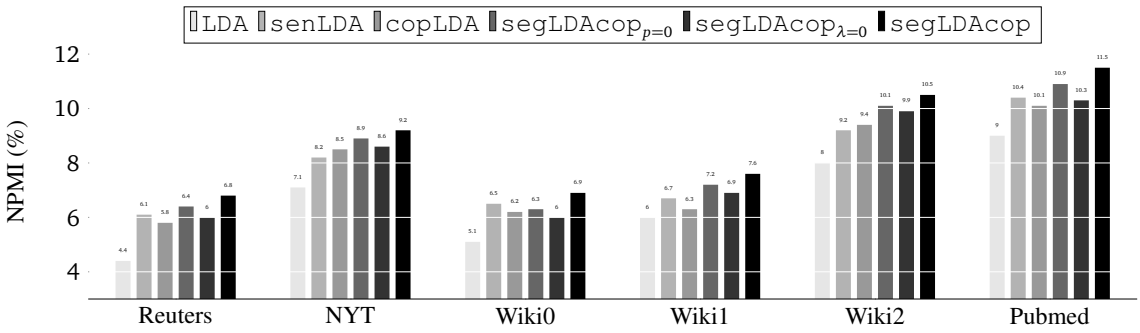


Figure 5.3: Topic coherence (NPMI) score with respect to 100 of topics.

For the reported scores, we aggregate the topic coherence scores over three different topic cardinalities: $N \in \{5, 10, 15\}$. segLDAcop model which uses copulas and segmentation together, shows the best score for the given reference meta-data (Wikipedia) in all of the datasets. It should be noted that segLDAcop$_{\lambda=0}$ which has not copula binder inside the model has less improvement against the segLDAcop$_{p=0}$ which has the copula. This means using copula has more effect on the topic coherence than only the segment-specific topic distribution.

5.3.4 Visualization

In order to illustrate the results obtained by `segLDAcop`, we display in Figure 5.5 the top 10 most probable words over 5 topics ($K = 20$) for the Reuters dataset, for both `segLDAcop` and `LDA`. In `segLDAcop`, topic 1, the top-ranked words are mostly relevant to the topic "date" (*e.g.*, march, january, year, fall, february, week). However, a similar topic learned by `LDA` appears to involve less such words (year, january, february), indicating a less coherent topic. It almost happens for the rest of categories.

Figure 5.4 illustrates another aspect of our model, namely the possibility to detect topically coherent segments. In particular, as one can note, the sentence is segmented in six parts by our model, the first one is a NP, *Ralph Borsodi* where one single topic is assigned to both words, there are other NPs and segments which have the same way in topics assignment and our model has cohered their topics. The data-driven approach we have adopted here can discover such fine grained differences, something the approaches based on fixed segmentations (either based on sentences or NPs), are less likely to achieve.

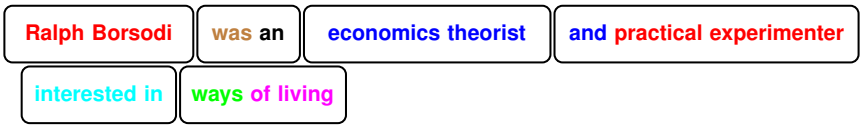


Figure 5.4: Topic assignments with segmentation boundaries using `segLDAcop`. Colors are topics (examples from Wiki0 including stopwords with 20 topics).

Topic1	march, fell, rose, january, rise, year, fall, february, pct, week	fell, mln, year, january, dlrs, rise, rose, pct, billion, february
Topic2	currency, bank, pct, cut, rate, day, prime, exchange, interest, national	billion, prime, day, rate, dlrs, pct, reserve, federal, fed, bank
Topic3	term, agreement, acquire, buy, sell, unit, acquisition, corp, company, sale	term, dlrs, buy, company, sell, unit, corp, acquisition, sale, mln
Topic4	approved, american, common, split, merger, company, board, stock, share, shareholder	acquire, mln, company, common, stock, shareholder, share, corp, merger, dlrs
Topic5	tokyo, life, intent, letter, buy, insurance, yen, japan, dealer, dollar	central, european, japan, yen, ec, dollar, bank, rate, dealer, market

Figure 5.5: Top-10 words of `segLDAcop` (left) vs `LDA` (right) for the Reuters (5 out of 20 topics).

5.4 Summary

In this chapter, we have introduced an LDA-based model that generates topically coherent segments within documents by jointly segmenting documents and assigning topics to their words. The coherence between topics is ensured through Frank's copula, that binds the topics associated to the words of a segment. In addition, this model relies on both document and segment specific topic distributions so as to capture fine grained differences in topic assignments. We have shown that this model naturally encompasses other state-of-the-art LDA-based models proposed to accomplish the same task, and that it outperforms these models over six publicly available collections in terms of perplexity, Normalized Pointwise Mutual Information (NPMI), a measure used to assess the coherence of topics with documents, and the Micro F1-measure in a text classification context. Our results confirm the importance of a flexible segmentation as well as a binding mechanism to produce topically coherent segments.

As regards complexity, it is true that more complex models, as the one we are considering, are more prone to underfitting (when data is scarce) and overfitting than simpler models. This said, the experimental results on perplexity (in which the word-topic distributions are fixed) and on classification (based on the topical induced representations) suggest that our model neither underfits nor overfits compared to simpler models. We believe that this is due to the fact that the main additional parameters in our model (the segment specific topic distribution) do not really add complexity as they are drawn from the same distribution as the standard document specific topics. Furthermore, the parameters p and f are simple parameters to choose between these two distributions.

The comparison with other segmentation methods is also an important point. While state-of-the-art supervised segmentation models can be used before applying the LDA model, we note such a pipeline approach comes with several limitations. The approach requires external annotated data to train the segmentation models, where certain domain and language specific information need to be captured. By contrast, our unsupervised approach learns both segmentations and topics jointly in a domain and language independent manner. Furthermore, existing supervised segmentation models are largely designed for a very different purpose with strong linguistic motivations, which may not align well with our main goal in this chapter which is improving topic coherence in topic modeling. Similarly, unsupervised approaches, used for example in the TDT (Topic Detection and Tracking) campaigns or more recently in [Du et al., 2013], usually consider coarse-grained topics, that

can encompass several sentences. In contrast, our approach aims at identifying fine-grained topics associated with coherent segments that do not overlap sentence boundaries. These considerations, explain the choice of the baselines retained: they are based on segments of different granularities (words, NPs, sentences) that do not overlap sentence boundaries.

In the future, we plan on relying on other inference approaches, based for example on variational Bayes known to yield better estimates for perplexity [Asuncion et al., 2009]; it is however not certain that the gain in perplexity one can expect from the use of variational bayes approaches will necessarily result in a gain in, say, topic coherence. Indeed, the impact of the inference approach on the different usages of latent topic models for text collections remains to be better understood.

Chapter 6

Conclusion

The goal of this thesis was to explore the problem of summarizing and discovering topics in a big collection of text dataset. Topic models as a solution to describe the semantics of a text corpus, are based on the concept that documents of a collection of words are mixtures of topics, where topics are vectors of probability distribution over words. As a matter of fact, a topic model is a generative model for the document and the words belong to them. It makes a specific probabilistic procedure to generate the words and consecutively the documents that contain them. Latent Dirichlet Allocation (LDA, [Blei et al., 2003]) as a probabilistic Bayesian topic model used to describe a corpus of D documents, associated with a vocabulary of size V. LDA based on the idea that documents in the collection are represented using random mixtures over hidden topics and each topic is identified by a distribution over words of the vocabulary associated with corpus.

In this work, we tried to study the main challenges with LDA: An important characteristic of LDA is that each document is generated independently from the previous ones. This is not a realistic assumption in different settings, as document streams and also an interesting objective in topic model can be to examine topic evolution and transitions, that in this case, LDA is not able to capture this evolution. Also, in LDA, word-order is not relevant and they are generated independently. This assumption called *Exchangeability* and has a direct influence on LDA to facilitate the inference development. Nonetheless, This is not again a realistic assumption as we may miss important information with various orders. Also words can be divided into different semantically coherent units such as Segments, Chunks, Sentences and Phrases that are not captured in LDA.
Regarding these two problems, we first positioned the recent relevant works in Chapter

2 and then introduced our models respectively for the former challenge in Chapter 3 and for the later challenge in Chapter 4 and 5. These models are based on the integration of Copula into LDA as a tool to capture dependencies between random variables.
Our distinct motivation to solve the problems using copula was the integrability of this tool into multinomial distribution on random variables that LDA utilizes for the topics. Copula is also capable of showing all the situations that may happen for the random variables like topics distribution and topic-words distribution in LDA, from completely independent to totally dependent. Among all of the families of copula and different functions of each family, we relied here on Archimedean copulas as they are symmetric, that is invariant by any permutation of their coordinates, which is important when dealing with exchangeable random variables, they are associative, meaning that the dependency properties are the same whatever the way we group the random variables. In the sequel, we used Frank function of Archimedean family which suits better with our problems where by varying its the only hyper-parameter λ from 0 to 1, this function allows one to model all the possible dependencies between two random variables, from complete independency to equality.

In Chapter 3, we have proposed new models for modeling topic and word-topic dependencies between consecutive documents in document streams. The first model is a direct extension of LDAand makes use of a Dirichlet distribution to balance the influence of the LDA prior parameters wrt to topic and word-topic distribution of the previous document. The second extension makes use of copulas, which constitute a generic tool to model dependencies between random variables. Lastly, the third model is a non-parametric extension of the second one through the integration of copulas in the stick-breaking construction of Hierarchical Dirichlet Processes. Our experiments, conducted on five standard collections that have been used in several studies on topic modeling, show that our proposals outperform previous ones, as dynamic topic models, temporal LDA and the Evolving Hierarchical Processes, both in terms of perplexity and for tracking similar topics in a document streams. Compared to previous proposals, our models have extra flexibility and can adapt to situations where there is, in fact, no dependencies between the documents. In the future, we plan to develop versions of these models that scale well, following the improvements on the inference methods for LDA, proposed in streams [Yao et al., 2009] or in online settings [Hoffman et al., 2010, Banerjee and Basu, 2007].

In Chapter 4, we proposed copLDA that extends LDA to incorporate the topical dependencies within sentences and noun-phrases using copulas. We have shown empirically the advantages of considering text structure and incorporating it in LDA with copulas.
In our future work we plan to integrate procedures to learn the λ parameter of Frank

copulas and to investigate ways to model not only dependencies within text segments like noun-phrases, but also dependencies between such segments with nested copulas.

In Chapter 5, we have introduced a LDA-based model that generates topically coherent segments within documents by jointly segmenting documents and assigning topics to their words. The coherence between topics is ensured through Frank's copula, that binds the topics associated to the words of a segment. In addition, this model relies on both document and segment specific topic distributions so as to capture fine-grained differences in topic assignments. We have shown that this model naturally encompasses other state-of-the-art LDA-based models proposed to accomplish the same task, and that it outperforms these models over six publicly available collections in terms of perplexity, Normalized Pointwise Mutual Information (NPMI), a measure used to assess the coherence of topics with documents, and the Micro F1-measure in a text classification context. Our results confirm the importance of a flexible segmentation as well as a binding mechanism to produce topically coherent segments.

In the future, we plan on relying on other inference approaches, based for example on variational Bayes known to yield better estimates for perplexity [Asuncion et al., 2009]; it is however not certain that the gain in perplexity one can expect from the use of variational bayes approaches will necessarily result in a gain in, say, topic coherence. Indeed, the impact of the inference approach on the different usages of latent topic models for text collections remains to be better understood.

Appendices

A.1 Metropolis-Hasting procedure

The Metropolis-Hasting procedure is based on the following steps:

1. Generate an initial value of x: draw $x^1 \sim P_{\text{prior}}(x)$
2. Initialize $j = 1$
3. Repeat till sequence is stable
   (a) Draw $x \sim q$, where q represents the "jump" function
   (b) Draw $u \sim U[0,1]$
   (c)
$$\alpha = \begin{cases} \frac{\Pi(x^j)q(x)}{\Pi(x)q(x^j)} & \text{if } \Pi(x^j)q(x) < \Pi(x)q(x^j) \\ \frac{\Pi(x)q(x^j)}{\Pi(x^j)q(x)} & \text{otherwise} \end{cases}$$
   (d) If $u \leq \alpha$, then $x^{j+1} = x$; $x^{j+1} = x^j$ otherwise

For $x = \lambda_d$, one has:

$$\begin{aligned} & P(\lambda_d|\theta^{d-1}, \theta^d, z^d, w^d, \alpha, \beta, \phi^{d-1}, \phi^d, \mu_d) \\ \propto \quad & P_{\text{prior}}(\lambda_d)P(\theta^d|\theta^{d-1}, \alpha, \lambda_d) \ := \Pi(\lambda_d) \end{aligned}$$

where $P_{\text{prior}}(\lambda_d) \sim U[0, \tau_\lambda]$. As λ_d should be higher when θ^{d-1} and θ^d are more similar (as in such a case the influence of θ^{d-1} on θ^d is more important), we make use of the following jump function, based on the exponential distribution:

$$q(\lambda_d) = (1 - cos(\theta^{d-1}, \theta^d)) \times e^{-(1-cos(\theta^{d-1}, \theta^d))\times\lambda_d}$$

For $x = \mu_d$, the same distribution is used for the jump function, the cosine being taken between the vectors that correspond to the column-wise concatenation of the columns of each matrix ϕ^{d-1} and ϕ^d. The prior this time is $P(\mu_d) \sim U[0, \tau_\mu]$. Lastly, for $x = \mathcal{T}_k^d$, $P_{\text{prior}}(\mathcal{T}_k^d) \sim Ga(\alpha)$, the jump function also corresponds to gamma distribution, and $\Pi(\mathcal{T}_k^d)$ corresponds to the k^{th} contribution in Eq. 3.10.

A.2 Gibbs sampling updates for ST−LDA−C

We provide here the complete derivation of Eq. 3.10. For any $d \geq 2$, one has:

$$\begin{aligned}
\mathscr{T}^d &\sim P(\mathscr{T}^d|\mathscr{T}^{d-1}, z^d, w^d, \alpha, \beta, \lambda_d, \phi^{d-1}, \phi^d, \mu_d) \\
&= \frac{P(\mathscr{T}^{d-1}|\alpha)P(\mathscr{T}^d|\mathscr{T}^{d-1}, \alpha, \lambda_d)P(z^d|\mathscr{T}^d)P(w^d|z^d)}{P(\mathscr{T}^{d-1}|\alpha)p(z^d|\alpha)P(w^d|z^d)} \\
&= \frac{P(\mathscr{T}^d|\mathscr{T}^{d-1}, \alpha, \lambda_d)P(z^d|\mathscr{T}^d)}{P(z^d|\alpha)}
\end{aligned}$$

Let F_α (resp f_α) denote the cdf (resp pdf) of the Gamma distribution with parameters $(\alpha, 1)$. By assumption:

$$P(\mathscr{T}^d|\mathscr{T}^{d-1}, \alpha, \lambda_d) = \prod_{k=1}^{K} f_\alpha(\mathscr{T}_k^d) c_\lambda(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d)) \tag{1}$$

and, since $\theta^d = \mathscr{T}^d / \left(\sum_{k=1}^{K} \mathscr{T}_k^d\right)$,:

$$P(z^d|\mathscr{T}^d) = \prod_{n=1}^{N} \theta^d_{z_n^d} = \left(\sum_{k=1}^{K} \mathscr{T}_k^d\right)^{-N} \prod_{n=1}^{N} \mathscr{T}^d_{z_n^d}$$

Further, as usual [Wang, 2008]:

$$P(z^d|\alpha) = \int P(z^d|\theta^d)P(\theta^d|\alpha)d\theta^d = \frac{B(\Omega_d + \alpha)}{B(\Omega_d)}$$

Hence:

$$\begin{aligned}
p(\mathscr{T}^d|\mathscr{T}^{d-1}, z^d, \cdots) =& \frac{\left(\sum_{k=1}^{K} \mathscr{T}_k^d\right)^{-N} \prod_{n=1}^{N} \mathscr{T}^d_{z_n^d}}{\left[\prod_{k=1}^{K} \Gamma(\alpha)\right] B(\Omega_d + \alpha)/B(\Omega_d)} \times \\
& \left[\prod_{k=1}^{K} {\mathscr{T}_k^d}^{\alpha-1} \exp^{-\mathscr{T}_k^d} c_\lambda(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d))\right] \\
=& \frac{\left(\sum_{k=1}^{K} \mathscr{T}_k^d\right)^{-N} \prod_{k=1}^{K} {\mathscr{T}_k^d}^{\Omega_{d,k}+\alpha-1}}{\left[\prod_{k=1}^{K} \Gamma(\alpha)\right] B(\Omega_d + \alpha)/B(\Omega_d)} \times \\
& \exp^{-\sum_{k=1}^{K} \mathscr{T}_k^d} \prod_{k=1}^{K} c_\lambda(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d))
\end{aligned}$$

To sum up:

$$P(\mathscr{T}^d|\mathscr{T}^{d-1}, z_d, w_d, \alpha, \beta, \lambda_d, \phi^{d-1}, \phi^d, \mu_d) \propto$$
$$(\sum_{k=1}^{K} \mathscr{T}_k^d)^{-N} \prod_{k=1}^{K} f_{(\Omega_{d,k}+\alpha-1)}(\mathscr{T}_k^d) \times \prod_{k=1}^{K} c_\lambda(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d))$$

Since we have proportion again we can use Metropolis-Hasting same as Appendix A.1 for sampling $\mathscr{T}^d$, rather than having Dirichlet distribution we use Frank copula joint distribution. We need also update λ parameter of copula distribution. We can access the $\lambda_d|\mathscr{T}^{d-1}, \mathscr{T}^d, z^d, w^d, \alpha, \beta, \phi^{d-1}, \phi^d, \mu_d$ assuming equation (1):

$$p(\lambda_d|\mathscr{T}^{d-1}, \mathscr{T}^d, z^d, w^d, \alpha, \beta, \phi^{d-1}, \phi^d, \mu_d) \propto$$
$$p(\lambda_d) \prod_{k=1}^{K} f_\alpha(\mathscr{T}_k^{d-1}) f_\alpha(\mathscr{T}_k^d) c_\lambda(F_\alpha(\mathscr{T}_k^{d-1}), F_\alpha(\mathscr{T}_k^d))$$

By getting benefit from Metropolis-Hasting algorithm with same configure as model ST−LDA−D, λ can be estimated.

A.3 Gibbs sampling updates for CopHDP

We provide here the complete derivation of Eq. 3.21. For any $d \geq 2$, one has:

$$p(\theta'^d|\theta'^{d-1}, z^d, w^d, \alpha_O\delta, \lambda_d, \mu_d, \phi^d, \phi^{d-1}, \beta) = \frac{p(z^d|\theta'^d)p(\theta'^d|\theta'^{d-1}, \alpha_0\delta)}{p(z^d|\alpha_0\delta)}$$

Let $G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}$ (resp $g_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}$) denote the cdf (resp pdf) of the Beta distribution with parameters $(\alpha_0\delta_k, \alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell))$. By assumption:

$$p(\theta'^d|\theta'^{d-1}, \alpha_0\delta) = \prod_{k=1}^{K+1} g_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^d) c_\lambda(G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d-1}), G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^d)) \tag{2}$$

and, since $\theta_k^d = \theta_k'^d \prod_{\ell=1}^{k-1}(1-\theta_\ell'^d)$,:

$$\begin{aligned} P(z^d|\theta'^d) &= \prod_{n=1}^{N} \theta_{z_n^d}^d = \prod_{k=1}^{K+1}(\theta_k'^d)^{\Omega_{d,k}} \prod_{k=1}^{K+1}\left(\prod_{\ell=1}^{k-1}(1-\theta_\ell'^d)\right)^{\Omega_{d,k}} \\ &= \prod_{k=1}^{K+1}(\theta_k'^d)^{\Omega_{d,k}} \prod_{k=1}^{K+1}\left(\prod_{m=k+1}^{K+1}(1-\theta_k'^d)\right)^{\Omega_{d,m}} \\ &= \prod_{k=1}^{K+1}(\theta_k'^d)^{\Omega_{d,k}} \times \prod_{k=1}^{K+1}(1-\theta_k'^d)^{\sum_{m=k+1}^{K+1}\Omega_{d,m}} \end{aligned}$$

Further, as usual [Wang, 2008]:

$$P(z^d|\alpha) = \int P(z^d|\theta^d)P(\theta^d|\alpha)d\theta^d = \frac{B(\Omega_d+\alpha)}{B(\Omega_d)}$$

Hence, using the explicit expression of the Beta distribution we deduce that

$$p(\theta'^{d}|\theta'^{d-1}, z^d, w^d, \alpha_O\delta, \lambda_d, \mu_d, \phi^d, \phi^{d-1}, \beta) = \frac{\Gamma(\alpha_0)^{K+1}\prod_{k=1}^{K+1}(\theta_k'^{d})^{\Omega_{d,k}} \times \prod_{k=1}^{K+1}(1-\theta_k'^{d})^{\sum_{m=k+1}^{K+1}\Omega_{d,m}}}{\left[\prod_{k=1}^{K+1}\Gamma(\alpha_0\delta_k)\Gamma(\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)\right]B(\Omega_d+\alpha)/B(\Omega_d)} \times$$
$$\left[\prod_{k=1}^{K+1}(\theta_k'^{d})^{\alpha_0\delta_k-1}(1-\theta_k'^{d})^{\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)-1}\right] \times$$
$$\prod_{k=1}^{K+1}\left[c_\lambda(G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d-1}), G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d}))\right]$$
$$= \frac{\Gamma(\alpha_0)^{K+1}\prod_{k=1}^{K+1}(\theta_k'^{d})^{\Omega_{d,k}+\alpha_0\delta_k-1}}{\left[\prod_{k=1}^{K+1}\Gamma(\alpha_0\delta_k)\Gamma(\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)\right]B(\Omega_d+\alpha)/B(\Omega_d)} \times$$
$$\prod_{k=1}^{K+1}(1-\theta_k'^{d})^{\sum_{m=k+1}^{K+1}\Omega_{d,m}+\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)-1} \times$$
$$\prod_{k=1}^{K+1}c_\lambda(G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d-1}), G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d})))$$

leading to:

$$p(\theta'^{d}|\theta'^{d-1}, z^d, w^d, \alpha_O\delta, \lambda_d, \mu_d, \phi^d, \phi^{d-1}, \beta) \propto \prod_{k=1}^{K+1} g_{\Omega_{d,k}+\alpha_0\delta_k, \sum_{m=k+1}^{K+1}\Omega_{d,m}+\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d}) \times$$
$$\prod_{k=1}^{K+1}c_\lambda(G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d-1}), G_{\alpha_0\delta_k,\alpha_0(1-\sum_{\ell=1}^{k-1}\delta_\ell)}(\theta_k'^{d}))$$

A.4 An efficient segmentation

Let us recall the property presented before:

Proposition A.4.1 *Let l_i^s be the random variable associated to the length of the segment starting at position i in a sentence of length M (positions go from 1 to M and l_i^s takes value in $\{1, \cdots, L\}$). Then $P(l_i^s = l) := \frac{g(M+1-i-l);L)}{g(M+1-i;L)}$ defines a probability distribution over l_i^s.*

Furthermore, the following process is equivalent to choosing sentence segmentations uniformly from the set of possible segmentations.

```
From pos. 1, repeat till end of sentence:
(a) Generate segment length acc. to P;
(b) Add segment to current segmentation;
(c) Move to position after the segment.
```

Proof Any segmentation of the sentence of length M starts with either a segment of length 1, a segment of length 2, $\cdots$, or a segment of length L. Thus, $g(M;L)$ can be defined through the following recurrence relation:

$$g(M;L) = \sum_{l=1}^{L} g(M-l;L) \tag{3}$$

together with the initial values $g(1;L), g(2;L), \cdots, g(L;L)$, which can be computed offline (for example, for $L = 3$, one has: $g(1;3) = 1, g(2;3) = 2, g(3;3) = 4$). Note that $g(1;L) = 1$ for all L.

Thus:

$$\sum_{l=1}^{L} P(l_i^s = l) = \sum_{l=1}^{L} \frac{g(M+1-i-l);L)}{g(M+1-i;L)} = 1$$

due to the recurrence relation on g. This proves the first part of the proposition.

Using the process described above where segments are generated one after another according to P, for a segmentation S, comprising $|S|$ segments, let us denote by $l_1, l_2, \cdots, l_{|S|}$ the lengths of each segment and by $i_1, i_2, \cdots, i_{|S|}$ the starting positions of each segment (with $i_1 = 1$). One has, as segments are independent of each other:

$$P(S) = \prod_{j=1}^{|S|} P(l^s_{i_j} = l_j) = \prod_{j=1}^{|S|} \frac{g(M+1-(i_j+l_j);L)}{g(M+1-i_j;L)}$$
$$= \frac{g(M-l_1;L)}{g(M;L)} \frac{g(M-l_1-l_2;L)}{g(M-l_1;L)} \cdots = \frac{1}{g(M;L)}$$

as $g(1;L) = 1$. This concludes the proof of the proposition. □

Furthermore, as one can note from Eq. 3, the various elements needed to compute $P(l^s_i = l)$ can be efficiently computed, the time complexity being equal to $O(M)$. In addition, as the number of different sentence lengths is limited, one can store the values of g to reuse them during the segmentation phase.

Publications

1. Hesam Amoualian, Marianne Clausel, Eric Gaussier and Massih-Reza Amini. (2016). Streaming-LDA: A Copula-based Approach to Modeling Topic Dependencies in Document Streams. In Proceedings of the 22nd International ACM SIGKDD Conference on Knowledge Discovery and Data Mining. ACM.

2. Hesam Amoualian, Wei Lu, Eric Gaussier, Georgios Balikas, Massih-Reza Amini and Marianne Clausel. (2017). Topical Coherence in LDA-based Models through Induced Segmentation. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics . ACL.

3. Georgios Balikas, Hesam Amoualian, Marianne Clausel, Eric Gaussier and Massih-Reza Amini. (2016). Modeling Topic Dependencies in Semantically Coherent Text Spans with Copulas. In Proceedings of the 26th International Conference on Computational Linguistics. COLING.

4. Hesam Amoualian, Marianne Clausel, Eric Gaussier. (2017). Une Approche á Base de Copules pour Modéliser les Dépendances entre les Thémes dans un Flux de Documents. In Proceedings of the 19th Cap La Conférence sur l'Apprentissage automatique. CAp.

5. Hesam Amoualian, Marianne Clausel, Eric Gaussier and Massih-Reza Amini. (2017). Copula-based Parametric and Non-parametric LDA Models for Document Streams. Submitted in the Journal of Machine Learning under review.

Bibliography

[Ahmed and Xing, 2008] Ahmed, A. and Xing, E. (2008). Dynamic Non-Parametric Mixture Models and The Recurrent Chinese Restaurant Process : with Applications to Evolutionary Clustering. *Proceedings of The Eighth SIAM International Conference on Data Mining (SDM2008)*.

[Ahmed and Xing, 2010] Ahmed, A. and Xing, E. P. (2010). Timeline: A dynamic hierarchical dirichlet process model for recovering birth/death and evolution of topics in text stream. In Grünwald, P. and Spirtes, P., editors, *UAI*, pages 20–29. AUAI Press.

[Aldous, 1985] Aldous, D. (1985). Exchangeability and related topics. In *École d'Été St Flour 1983*, pages 1–198. Springer-Verlag. Lecture Notes in Math. 1117.

[Amoualian et al., 2016] Amoualian, H., Clausel, M., Gaussier, E., and Amini, M.-R. (2016). Streaming-lda: A Copula-based Approach to Modeling Topic Dependencies in Document Streams. In *Proceedings of the 22nd International ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM.

[Amoualian et al., 2017] Amoualian, H., Lu, W., Gaussier, E., Balikas, G., Amini, M.-r., and Clausel, M. (2017). Topical coherence in LDA-based models through induced segmentation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. ACL.

[Antoniak, 1974] Antoniak, C. E. (1974). Mixtures of dirichlet processes with applications to bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174.

[Asuncion et al., 2009] Asuncion, A., Welling, M., Smyth, P., and Teh, Y. W. (2009). On smoothing and inference for topic models. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 27–34, Arlington, Virginia, United States. AUAI Press.

[Balikas et al., 2016a] Balikas, G., Amini, M.-R., and Clausel, M. (2016a). On a topic model for sentences. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM.

[Balikas et al., 2016b] Balikas, G., Amoualian, H., Clausel, M., Gaussier, E., and Amini, M. R. (2016b). Modeling topic dependencies in semantically coherent text spans with copulas. In *Proceedings of the 26th International Conference on Computational Linguistics*. COLING.

[Banerjee and Basu, 2007] Banerjee, A. and Basu, S. (2007). Topic models over text streams: A study of batch and online unsupervised learning. In *Proceedings of the 7th SIAM conference on Data Mining, SDM*.

[Beeferman et al., 1999] Beeferman, D., Berger, A., and Lafferty, J. (1999). Statistical models for text segmentation. *Machine Learning*, 34(1):177–210.

[Bird et al., 2009] Bird, S., Klein, E., and Loper, E. (2009). *Natural language processing with Python*. " O'Reilly Media, Inc.".

[Blackwell and MacQueen, 1973] Blackwell, D. and MacQueen, J. (1973). Ferguson distributions via Polya urn schemes. *The Annals of Statistics*, 1:353–355.

[Blei, 2008] Blei, D. M. (2008). Free C++ implementation for dtm. `https://www.cs.princeton.edu/~blei/topicmodeling.html`.

[Blei and Lafferty, 2006] Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 113–120, New York, NY, USA. ACM.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning*, 3:993–1022.

[Boyd-Graber and Blei, 2009] Boyd-Graber, J. L. and Blei, D. M. (2009). Syntactic topic models. In *Advances in neural information processing systems*, pages 185–192.

[Choi, 2000] Choi, F. Y. Y. (2000). Advances in domain independent linear text segmentation. In *Proceedings of the 1st North American Chapter of the Association for Computational Linguistics Conference*, NAACL 2000, pages 26–33, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.*, 20(3):273–297.

[Deerwester et al., 1990] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

[Derrode and Pieczynski, 2013] Derrode, S. and Pieczynski, W. (2013). Unsupervised data classification using pairwise markov chains with automatic copulas selection. *Computational Statistics & Data Analysis*.

[Du et al., 2010a] Du, L., Buntine, W., and Jin, H. (2010a). A Segmented Topic Model Based on the Two-parameter Poisson-Dirichlet Process. *Journal of Machine learning*, 81(1):5–19.

[Du et al., 2013] Du, L., Buntine, W., and Johnson, M. (2013). Topic Segmentation with a Structured Topic Model. In *Proceedings of HLT-NAACL*, pages 190–200.

[Du et al., 2010b] Du, L., Buntine, W. L., and Jin, H. (2010b). Sequential latent dirichlet allocation: Discover underlying topic structures within a document. In *IEEE Computer Society, ICDM*.

[Dunson, 2006] Dunson, D. B. (2006). Bayesian dynamic modeling of latent trait distributions. *Biostatistics*, 7(4):551.

[Eckart and Young, 1936] Eckart, C. and Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218.

[Eisenstein and Barzilay, 2008] Eisenstein, J. and Barzilay, R. (2008). Bayesian unsupervised topic segmentation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 334–343, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Elidan, 2012] Elidan, G. (2012). Copula network classifiers (cncs). In *International Conference on Artificial Intelligence and Statistics*, pages 346–354.

[Elidan, 2013] Elidan, G. (2013). Copulas in Machine Learning. In Jaworski, P., Durante, F., and Härdle, W. K., editors, *Copulae in Mathematical and Quantitative Finance*, volume 213 of *Lecture Notes in Statistics*, pages 39–60. Springer Berlin Heidelberg.

[Embrechts et al., 2002] Embrechts, P., McNeil, A., and Straumann, D. (2002). Correlation and Dependence in Risk Management: Properties and Pitfalls. *Journal of Risk management: value at risk and beyond*, pages 176–223.

[Ferguson, 197] Ferguson, T. S. (197). A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230.

[Fisher, 1997] Fisher, N. I. (1997). Copulas. *Encyclopedia of Statistical Sciences*, pages 159–163.

[Gaber et al., 2005] Gaber, M. M., Zaslavsky, A., and Krishnaswamy, S. (2005). Mining data streams: A review. *ACM SIGMOD Record*.

[Green, 2001] Green, P. J. (2001). Modelling heterogeneity with and without the dirichlet process. *Scandinavian Journal of Statistics*, 28(2):355–375.

[Griffiths and Steyvers, 2004] Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235.

[Griffiths et al., 2005] Griffiths, T. L., Steyvers, M., Blei, D. M., and Tenenbaum, J. B. (2005). Integrating topics and syntax. In Saul, L. K., Weiss, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 17*, pages 537–544. MIT Press.

[Heinrich, 2005] Heinrich, G. (2005). Parameter estimation for text analysis. Technical report, Technical report.

[Heinrich, 2011] Heinrich, G. (2011). Infinite lda – implementing the hdp with minimum code complexity. Technical report, arbylon.net.

[Hofert, 2011] Hofert, M. (2011). Efficiently Sampling Nested Archimedean Copulas. *Journal of Computational Statistics & Data Analysis*, 55(1):57–70.

[Hofert et al., 2011] Hofert, M., Mächler, M., et al. (2011). Nested archimedean copulas meet r: The nacopula package. *Journal of Statistical Software*, 39(9):1–20.

[Hoffman et al., 2010] Hoffman, M. D., Blei, D. M., and Bach, F. (2010). Online learning for latent dirichlet allocation. In *NIPS*.

[Hofmann, 1999] Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22Nd Annual International Conference on Research and Development in Information Retrieval*, SIGIR, pages 50–57, New York, NY, USA. ACM.

[Hong et al., 2011] Hong, L., Dom, B., Gurumurthy, S., and Tsioutsiouliklis, K. (2011). A time-dependent topic model for multiple text streams. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD*.

[Ishwaran and James, 2001] Ishwaran, H. and James, L. F. (2001). Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96:161–173.

[Ishwaran and Zarepour, 2002] Ishwaran, H. and Zarepour, M. (2002). Exact and approximate sum-representations for the Dirichlet process. *The Canadian Journal of Statistics*, 30(2):269–283.

[Iwata et al., 2009] Iwata, T., Watanabe, S., Yamada, T., and Ueda, N. (2009). Topic tracking model for analyzing consumer purchase behavior. In *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, IJCAI'09, pages 1427–1432, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[Kalman and Others, 1960] Kalman, R. E. and Others (1960). A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45.

[Kazantseva and Szpakowicz, 2011] Kazantseva, A. and Szpakowicz, S. (2011). Linear text segmentation using affinity propagation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 284–293, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Klein and Manning, 2003] Klein, D. and Manning, C. (2003). Accurate Unlexicalized Parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430. Association for Computational Linguistics.

[Landauer and Dumais, 1997] Landauer, T. and Dumais, S. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211–240.

[Lau and Baldwin, 2016] Lau, J. H. and Baldwin, T. (2016). The sensitivity of topic coherence evaluation to topic cardinality. In *Proceedings of The Annual Conference of the North American Chapter of the Association for Computational Linguistics, Human Language Technologies, San Diego California, USA, June 12-17, 2016*, NAACL, pages 483–487.

[Lau et al., 2013] Lau, J. H., Baldwin, T., and Newman, D. (2013). On collocations and topic models. *Journal of ACM Trans. Speech Lang. Process.*, 10(3):10:1–10:14.

[Lichman, 2013] Lichman, M. (2013). UCI machine learning repository.

[Liu et al., 2009] Liu, H., Lafferty, J., and Wasserman, L. (2009). The nonparanormal: Semiparametric estimation of high dimensional undirected graphs. *The Journal of Machine Learning Research*, 10:2295–2328.

[M. Wallach et al., 2009] M. Wallach, H., M. Mimno, D., and McCallum, A. (2009). Rethinking LDA: why priors matter. In *Advances in Neural Information Processing Systems 22*, pages 1973–1981. Curran Associates, Inc.

[Marcus et al., 1993] Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: The penn treebank. *Comput. Linguist.*, 19(2):313–330.

[Marshall and Olkin, 1988] Marshall, A. W. and Olkin, I. (1988). Families of multivariate distributions. *Journal of the American Statistical Association*, 83(403):834–841.

[Mcauliffe and Blei, 2008] Mcauliffe, J. D. and Blei, D. M. (2008). Supervised topic models. In *Advances in neural information processing systems*, pages 121–128.

[McLachlan and Peel, 2000] McLachlan, G. J. and Peel, D. (2000). *Finite mixture models*. Wiley Series in Probability and Statistics, New York.

[McNeil, 2008] McNeil, A. (2008). Sampling Nested Archimedean Copulas. *Journal of Statistical Computation and Simulation*, 78(6):567–581.

[McNeil and Nešlehovà, 2009] McNeil, A. J. and Nešlehovà, J. (2009). Multivariate Archimedean copulas, D-monotone functions and ℓ_1-norm symmetric distributions. *Annals of Statistics*.

[Menger, 1942] Menger, K. (1942). Statistical metrics. *Proceedings of the National Academy of Sciences of the United States of America*, 28(12):535–537.

[Mimno and McCallum, 2007] Mimno, D. and McCallum, A. (2007). Organizing the oca: Learning faceted subjects from a library of digital books. In *Proceedings of the 7th Joint Conference on Digital Libraries*, JCDL '07, pages 376–385, New York, NY, USA. ACM.

[Mimno et al., 2009] Mimno, D., Wallach, H. M., Naradowsky, J., Smith, D. A., and McCallum, A. (2009). Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09, pages 880–889, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Mimno et al., 2011] Mimno, D., Wallach, H. M., Talley, E., Leenders, M., and McCallum, A. (2011). Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP, pages 262–272, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Neal, 2003] Neal, R. M. (2003). Slice sampling. *The Annals of Statistics*, 31(3):705–767.

[Nelsen, 2007] Nelsen, R. B. (2007). *An introduction to copulas*. Springer Science & Business Media.

[Newman et al., 2010] Newman, D., Lau, J. H., Grieser, K., and Baldwin, T. (2010). Automatic evaluation of topic coherence. In *Proceedings of The Annual Conference of the North American Chapter of the Association for Computational Linguistics, Human Language Technologies*, NAACL, pages 100–108, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Ng et al., 2011] Ng, K. W., Tian, G.-L., and Tang, M.-L. (2011). *Dirichlet and related distributions: Theory, methods and applications*. John Wiley & Sons.

[Nigam et al., 2000] Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. (2000). Text classification from labeled and unlabeled documents using em. *Mach. Learn.*, 39(2-3):103–134.

[Ostap et al., 2013] Ostap, O., Yarema, O., and Wolfgang, S. (2013). Properties of hierarchical Archimedean copulas. *Statistics & Risk Modeling*.

[Partalas et al., 2015] Partalas, I., Kosmopoulos, A., Baskiotis, N., Artieres, T., Paliouras, G., Gaussier, E., I.Androutsopoulos, Amini, M., and Galinari, P. (2015). LSHTC: A Benchmark for Large-Scale Text Classification. *Journal of CoRR*, abs/1503.08581.

[Paul and Girju, 2010] Paul, M. and Girju, R. (2010). A two-dimensional topic-aspect model for discovering multi-faceted topics. In *Proceedings of the 24th Conference on Artificial Intelligence*, AAAI, pages 545–550. AAAI Press.

[Paul and Dredze, 2011] Paul, M. J. and Dredze, M. (2011). You are what you tweet: Analyzing twitter for public health. In *International Conference on Weblogs and Social Media*.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-Learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[Pevzner and Hearst, 2002] Pevzner, L. and Hearst, M. A. (2002). A critique and improvement of an evaluation metric for text segmentation. *Comput. Linguist.*, 28(1):19–36.

[Pitman, 1996] Pitman, J. (1996). Random discrete distributions invariant under size-biased permutation. *Adv. in Appl. Probab.*, 28(2):525–539.

[Pitman, 2002] Pitman, J. (2002). Poisson-dirichlet and gem invariant distributions for split-and-merge transformations of an interval partition. *Combinatorics, Probability & Computing*, 11(5):501–514.

[Popescul et al., 2001] Popescul, A., Ungar, L. H., Pennock, D. M., and Lawrence, S. (2001). Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, UAI '01, pages 437–444, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[Powell, 1987] Powell, M. J. D. (1987). Radial basis functions for multivariable interpolation: A review. In Mason, J. C. and Cox, M. G., editors, *Algorithms for Approximation*, pages 143–167. Clarendon Press, New York, NY, USA.

[Purver et al., 2006] Purver, M., Griffiths, T. L., Körding, K. P., and Tenenbaum, J. B. (2006). Unsupervised topic modelling for multi-party spoken discourse. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL, pages 17–24, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Rabiner, 1990] Rabiner, L. R. (1990). A tutorial on hidden markov models and selected applications in speech recognition. In Waibel, A. and Lee, K.-F., editors, *Readings in Speech Recognition*, pages 267–296. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[Rasmussen, 1999] Rasmussen, C. E. (1999). The infinite gaussian mixture model. In Solla, S. A., Leen, T. K., and Müller, K.-R., editors, *NIPS*, pages 554–560. The MIT Press.

[Ren et al., 2008] Ren, L., Dunson, D. B., and Carin, L. (2008). The dynamic hierarchical dirichlet process. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 824–831, New York, NY, USA. ACM.

[Rényi, 1959] Rényi, A. (1959). On measures of dependence. *Acta Mathematica Academiae Scientiarum Hungarica*, 10(3):441–451.

[Rosenfeld, 2000] Rosenfeld, R. (2000). Two decades of statistical language modeling: Where do we go from here? In *Proceedings of the IEEE*, volume 88, pages 1270–1278.

[Sakaki et al., 2010] Sakaki, T., Okazaki, M., and Matsuo, Y. (2010). Earthquake shakes twitter users: Real-time event detection by social sensors. In *International Conference on World Wide Web*.

[Salton et al., 1975] Salton, G., Wong, A., and Yang, C. (1975). A Vector Space Model for Automatic Indexing. *Journal of Communications of the ACM*, 18(11):613–620.

[Schweizer and Wolff, 1981] Schweizer, B. and Wolff, E. F. (1981). On nonparametric measures of dependence for random variables. *Ann. Statist.*, 9(4):879–885.

[Sethuraman, 1994] Sethuraman, J. (1994). A constructive definition of Dirichlet priors. *Statistica Sinica*, 4:639–650.

[Shafiei and Milios, 2006] Shafiei, M. M. and Milios, E. E. (2006). Latent dirichlet co-clustering. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pages 542–551. IEEE.

[Shannon, 1948] Shannon, C. (1948). A mathematical theory of communication. *Bell system technical journal*, 27.

[Sklar, 1959] Sklar, M. (1959). *Fonctions de Répartition À N Dimensions Et Leurs Marges*. Université Paris 8.

[Tamura and Sumita, 2016] Tamura, A. and Sumita, E. (2016). Bilingual segmented topic model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, ACL.

[Teh, 2006] Teh, Y. W. (2006). A hierarchical bayesian language model based on pitman-yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL, pages 985–992, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Teh and Jordan, 2010] Teh, Y. W. and Jordan, M. I. (2010). *Hierarchical Bayesian nonparametric models with applications*, chapter ., page . Cambridge University Press.

[Teh et al., 2006] Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

[Tran et al., 2015] Tran, D., Blei, D., and Airoldi, E. (2015). Copula Variational Inference. In *Proceedings of Neural Information Processing Systems 28 NIPS*, pages 3564–3572.

[Trivedi and Zimmer, 2007] Trivedi, P. and Zimmer, D. (2007). *Copula Modeling: An Introduction for Practitioners*. Now Publishers Inc.

[Tsatsaronis et al., 2015] Tsatsaronis, G., Balikas, G., Malakasiotis, P., Partalas, I., Zschunke, M., Alvers, M., Weissenborn, D., Krithara, A., Petridis, S., Polychronopoulos, D., et al. (2015). An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *Journal of BMC bioinformatics*, 16(1):1.

[Wang et al., 2008] Wang, C., Blei, D. M., and Heckerman, D. (2008). Continuous time dynamic topic models. In McAllester, D. A. and Myllymäki, P., editors, *UAI*, pages 579–586. AUAI Press.

[Wang et al., 2011] Wang, C., Paisley, J., and Blei, D. M. (2011). Online variational inference for the hierarchical dirichlet process. In *Proc. of the 14th Int'l. Conf. on Artificial Intelligence and Statistics (AISTATS)*, volume 15, pages 752–760.

[Wang et al., 2009] Wang, D., Zhu, S., Li, T., and Gong, Y. (2009). Multi-document summarization using sentence-based topic models. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 297–300. Association for Computational Linguistics.

[Wang et al., 2017] Wang, P., Zhang, P., Zhou, C., Li, Z., and Yang, H. (2017). Hierarchical evolving dirichlet processes for modeling nonlinear evolutionary traces in temporal data. *Data Mining and Knowledge Discovery*, 31(1):32–64.

[Wang and McCallum, 2006] Wang, X. and McCallum, A. (2006). Topics over time: A non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD*.

[Wang et al., 2007] Wang, X., McCallum, A., and Wei, X. (2007). Topical n-grams: Phrase and topic discovery, with an application to information retrieval. In *Proceedings of the 7th ICDM International Conference on Data Mining*, pages 697–702, Washington, DC, USA. IEEE Computer Society.

[Wang, 2008] Wang, Y. (2008). Distributed gibbs sampling of latent topic models: The gritty details. Technical report, .

[Wang et al., 2012] Wang, Y., Agichtein, E., and Benzi, M. (2012). TM-LDA: efficient online modeling of latent topic transitions in social media. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 123–131, New York, NY, USA. ACM.

[Wei and Croft, 2006] Wei, X. and Croft, W. B. (2006). LDA-based document models for ad-hoc retrieval. In *Proceedings of the 29th Annual International Conference on Research and Development in Information Retrieval*, SIGIR, pages 178–185, New York, NY, USA. ACM.

[Wei et al., 2007] Wei, X., Sun, J., and Wang, X. (2007). Dynamic mixture models for multiple time series. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence, IJCAI*.

[Wilson and Ghahramani, 2010] Wilson, A. and Ghahramani, Z. (2010). Copula Processes. In *Advances in Neural Information Processing Systems 23 NIPS*, pages 2460–2468. Curran Associates, Inc.

[Yao et al., 2009] Yao, L., Mimno, D., and McCallum, A. (2009). Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD*.

[Yao et al., 2016] Yao, L., Zhang, Y., Wei, B., Li, L., Wu, F., Zhang, P., and Bian, Y. (2016). Concept over time: the combination of probabilistic topic model with wikipedia knowledge. *Journal of Expert Systems with Applications*, 60:27 – 38.

[Zeng et al., 2004] Zeng, H.-J., He, Q.-C., Chen, Z., Ma, W.-Y., and Ma, J. (2004). Learning to cluster web search results. In *Proceedings of the 27th Annual International Conference on Research and Development in Information Retrieval*, SIGIR, pages 210–217, New York, NY, USA. ACM.

[Zhu et al., 2006] Zhu, X., Blei, D., and Lafferty, J. (2006). TagLDA: Bringing document structure knowledge into topic models. Technical Report TR-1553, University of Wisconsin, Madison.