algorithms

MDPI

Article

A Comparison of Different Topic Modeling Methods through a Real Case Study of Italian Customer Care

Gabriele Papadia [1], Massimo Pacella [1,*], Massimiliano Perrone [1] and Vincenzo Giliberti [2]

[1] Department of Engineering for Innovation, University of Salento, 73100 Lecce, Italy
[2] IN & OUT S.p.A. a Socio Unico Teleperformance S.E., 74121 Taranto, Italy
* Correspondence: massimo.pacella@unisalento.it

Abstract: The paper deals with the analysis of conversation transcriptions between customers and agents in a call center of a customer care service. The objective is to support the analysis of text transcription of human-to-human conversations, to obtain reports on customer problems and complaints, and on the way an agent has solved them. The aim is to provide customer care service with a high level of efficiency and user satisfaction. To this aim, topic modeling is considered since it facilitates insightful analysis from large documents and datasets, such as a summarization of the main topics and topic characteristics. This paper presents a performance comparison of four topic modeling algorithms: (i) Latent Dirichlet Allocation (LDA); (ii) Non-negative Matrix Factorization (NMF); (iii) Neural-ProdLDA (Neural LDA) and Contextualized Topic Models (CTM). The comparison study is based on a database containing real conversation transcriptions in Italian Natural Language. Experimental results and different topic evaluation metrics are analyzed in this paper to determine the most suitable model for the case study. The gained knowledge can be exploited by practitioners to identify the optimal strategy and to perform and evaluate topic modeling on Italian natural language transcriptions of human-to-human conversations. This work can be an asset for grounding applications of topic modeling and can be inspiring for similar case studies in the domain of customer care quality.

Keywords: topic modeling; latent dirichlet allocation; non-negative matrix factorization; neural-ProdLDA; contextualized topic models; Italian natural language

Citation: Papadia, G.; Pacella, M.; Perrone, M.; Giliberti, V. A Comparison of Different Topic Modeling Methods through a Real Case Study of Italian Customer Care. *Algorithms* **2023**, *16*, 94. https://doi.org/10.3390/a16020094

Academic Editors: Aneesha Bakharia and Khanh Luong

Received: 12 December 2022
Revised: 28 January 2023
Accepted: 6 February 2023
Published: 8 February 2023

1. Introduction

Topic modeling (TM) is a mathematical model used in the field of Machine Learning that allows the identification of recurring patterns of words within a dataset in text format. In the case considered in this paper, the dataset consists of real conversations in the Italian natural language between customers and agents in a customer support center. The use of TM methods within a context such as that of customer care would make it possible to optimize the work of the operators and improve the customer experience through an immediate identification of patterns within the discussion.

The TM methods considered in this paper are as follows: (i) Latent Dirichlet Allocation (LDA) [1]; (ii) Non-negative Matrix Factorization (NMF) [2]; (iii) Neural-ProdLDA [3] and (iv) Contextualized Topic Models (CTM) [4]. The metrics used for evaluation of the efficacy of these algorithms are Topic Diversity [5] and Similarity [6]. The present paper aims to extend the analysis described in [7], investigating the comparison of different fitting methods of LDA-based TM algorithms.

In addition to using an approach based on pre-processing techniques adapted to the Italian-language dataset, it is essential to analyze the results obtained to make the most of the potential derived from using Text Mining techniques. This process can be particularly demanding for the supervision of the Italian language, requiring innovative solutions capable of correctly investigating the case study in question.

This paper is outlined as follows: Section 2 deals with the theoretical foundations of TM and the implemented methods. It also describes the metrics used to evaluate these TM methods. Section 3 is focused on the fundamental step of pre-processing with a focus on the adaptations required by the Italian natural language. Section 4 describes the case study and the results obtained in the performance comparison. Finally, Section 5 provides the conclusion and suggestions for future studies.

1.1. Literature Review

In [7], the authors focused on a probabilistic model based on the LDA to automatically cluster dialogues between customers and agents of customer service in the Italian natural language. Examples of a comparison study of different TM algorithms, such as the probabilistic LDA model and the non-probabilistic model as NMF, are reported in the literature [2].

Authors in [8] proposed an analysis of the evolution of unsupervised TM algorithms starting from the first models dating back to the 90s, arriving at a comparison with recent models. A total of 40 TM models were considered in the comparison study. Amongst these, LDA and Dirichlet Multinomial Mixture (DMM) were discussed in detail. The DMM [9] is a model based on the Dirichlet distribution, with a multivariate generalization of the beta distribution. Similarly, the CTM [10] is a variant of LDA and differs from it in the statistical distribution. In particular, the authors replaced the Dirichlet distribution with a normal distribution and added a covariance matrix between the topics to model the correlation.

Among the most recent models, it is worth mentioning the Embedded Topic Model (ETM) [11], in which words and topics are both represented by vectors in an embedding space. In this case, the generative process is similar to that of the LDA, in which each document has topics structured according to a probabilistic distribution. In [11], the authors also proposed a discussion about the current state of the art regarding TM, introducing their views on the criteria that researchers should adopt to select a TM suitable for their needs.

In [12], the authors proposed a comparison study of different TM methods applied to multiple datasets in the English language. In [12], two evaluation metrics were used, i.e., Topic Coherence and Topic Diversity. Topic Coherence was analyzed through normalized pointwise mutual information [13] and external word embeddings topic coherence [14].

For a more in-depth literature review on TM, the reader is referred to recent surveys [15–17]. In particular, in [17], the authors used a holistic approach to survey TM. They discussed four categories of TM (i.e., algebraic, fuzzy, probabilistic, and neural), presented the wide variety of available models with a structured viewpoint, analyzed these models' characteristics and limitations, and discuss their proper use cases. Further aspects, which are illustrated in the survey study [17], relate to the criteria for appropriate evaluation of topic models, to the applications with some popular software tools that provide an implementation of some models, and to the available datasets and benchmarks.

1.2. Motivation Example

The present study concerns the analysis of conversation transcriptions between customers and agents in a customer care call center. The objective is to support the analysis of text transcription of human-to-human conversations, to obtain reports on customer problems and complaints, and on how an agent has solved them. The aim is to provide customer care service with a high level of efficiency and user satisfaction.

From the studies in the literature, the highest increase in the topics concerning the analysis of spoken dialogues is evident, as highlighted in the papers [18,19]. Topic modeling is the fundamental element within the study addressed in this paper, for which a state-of-the-art review is in [20,21].

One of the main difficulties concerning Natural Language Processing (NLP) related to real-world dialogues is the interpretation of the textual content from the unpredictability of human responses and behaviors [22]. In our specific case, an additional difficulty is due to the processing of dialogues in the Italian natural language, which is still not adequately faced in the literature.

Until now, most representative text pre-processing tools and language models are based on English. Therefore, the majority of data sources used in text-based research are also documented in English. Current NLP methods and algorithms mainly focus on several high-resource languages. Other languages, such as Italian, have been underserved by NLP approaches due to the lack of datasets and support. These challenges make the development of NLP models expensive and time-consuming. Therefore, it is essential to develop research projects at the country level to create NLP applications for domestic languages. The present research study aims to contribute to the literature by developing an NLP application—specifically, a TM algorithm—for the Italian natural language. This work can be an asset for grounding applications of topic modeling and can be inspiring for similar case studies in the area of customer care quality. A few case studies concerning text analysis in the Italian language appeared in the recent literature and can be found in [23–25].

The ultimate goal is to investigate the most performing method in the TM field to adapt it to our case study. This aim would make it possible to improve the customer experience by optimizing the quality of responses to customer requests by operators. In particular, the Topic Visualization tools are valid in the investigation of useful outputs coming from TM [26].

2. Topic Modeling

One of the most used techniques in text mining is topic modeling. The hypothesis at the basis of the method is that the process of generating a text by its author foresees two successive moments: (i) the decision of the topics and (ii) the creation of the document. What makes the method particularly interesting for reading a collection of documents and automatically extracting the topics is its application in the opposite direction [27]. The selection of topics does not require coding or classification of data defined a priori by any human intervention. Therefore, these are unsupervised learning techniques.

For example, in reading a geography magazine, we might expect to find the following topics: mountains, hills, and sea. Of course, each topic is associated with the most probable words. The word "fish" is more likely to appear when referring to the sea rather than the mountain. The topics in the individual articles can remain separate from each other, characterizing each different article, but they can be combined into different measures within each article. The goal of thematic templates is to capture this semantic variability of documents in each topic.

Topics represent clusters of words, which tend to co-occur within the corpus documents, identified by a probabilistic model. A topic is, in fact, a probabilistic distribution of all the words of the vocabulary of a corpus, in which the most probable words are the ones that best describe its content.

The TM uses numerical vectors of the type document $\times$ term, which identify documents as inputs that, subsequently, are transformed into topic $\times$ term and document $\times$ topic vectors. Finally, each set of words is assigned to a specific topic through the document $\times$ term matrix. One of the most used approaches in TM is the *bag of words* (BoW), in which documents are a collection of unordered words. The "term frequency"-"inverse-document-frequency" is a widely-used technique for implementing a BoW in TM by adding the importance of the term in the collection to the count vector. "Term frequency" counts how many times each phrase appears in a document, while "inverse document frequency" determines the extent to which a phrase is used throughout the text. The different topic modeling methodologies implemented in the case study are briefly described below.

2.1. Latent Dirichlet Allocation (LDA)

The LDA algorithm was devised by David Blei et al. and presented in an article in the Journal of Machine Learning Research in 2003 [1]. It is a model of natural language analysis that allows the analyst to understand the semantic meaning of the text and identify the main topics. This algorithm does not require pre-existing annotations on documents and works in two modes:

1. Retrospective topic detection: the algorithm identifies the topics featured in a set of "never seen before" data; after processing them, it groups them into homogeneous clusters.
2. Online new topic detection: the algorithm processes and establishes whether textual data deal with new topics or belongs to the existing clusters.

As shown in Figure 1, the words $w_{m,n}$ $\forall d = 1, \ldots, M$, $\forall n = 1, \ldots, N$ are variables observed within the corpus of documents, while the topics ϕ_k $\forall k = 1, \ldots, K$, are the distribution of topics for each document θ_m $\forall d = 1, \ldots, M$, and the topic assignment for each word are the unknowns of the system that requests from the specific dataset processed. The parameters α and β are the hyperparameters of the model: α represents the document–topic density, and β represents the topic–word density. A large value of α indicates that documents consist of multiple arguments, while low values of α indicate that the documents contain few topics. Similarly, high values of β indicate that the themes are composed of a large number of words and vice versa.

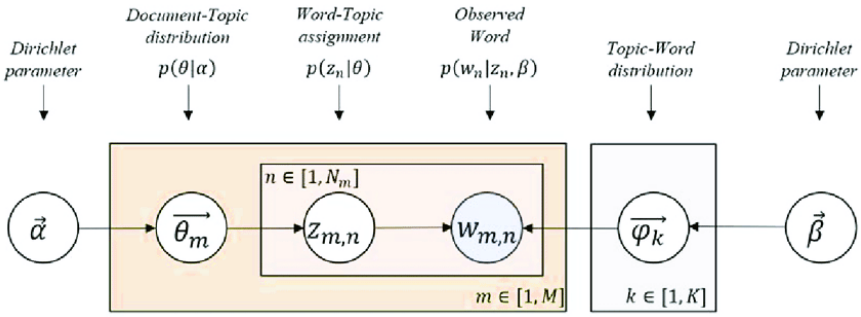

Figure 1. Latent Allocation Dirichlet (LDA) for TM.

Mathematically, the model can be summarized as follows:

$$\begin{aligned} \phi_k &\sim Dirichlet(\vec{\beta}) \\ \theta_m &\sim Dirichlet(\vec{\alpha}) \\ z_{m,n}|\theta_m &\sim Multinomial(\theta_m) \\ w_{m,n}|\phi_{z_{m,n}} &\sim Multinomial(\phi_{z_{m,n}}). \end{aligned} \tag{1}$$

The multinomial distribution represents a discrete multivariate distribution (a generalization of the scalar binomial distribution); the data correspond to the observed set of words $w_{m,n}$ within each transcription d. The posterior distribution of the topic distributions Φ and topical mixtures Θ is given by the posterior conditional probability

$$P(\Phi, \Theta, \mathbf{z}|\mathbf{w}, \vec{\alpha}, \vec{\beta}) = \frac{P(\Phi, \Theta, \mathbf{z}, \mathbf{w}|\vec{\alpha}, \vec{\beta})}{P(\mathbf{w}|\vec{\alpha}, \vec{\beta})}, \tag{2}$$

where **z** and **w** are vectors of topic assignments and words, respectively.

2.2. Non-Negative Matrix Factorization (NMF)

The Non-negative Matrix Factorization (NMF) algorithm, presented by Lee and Seung in 2000 [2], has proven to be an effective model for extracting arguments from a textual data corpus. Contrary to the LDA, which can also be a supervised model, the NMF is necessarily

unsupervised. It also uses dimensionality reduction methods for "non-negative matrices" that are composed of positive or null values only. From a mathematical point of view, it takes a non-negative matrix V and finds non-negative factors W and H such that

$$\mathrm{V} \approx WH \tag{3}$$

Given a set of multivariate n-dimensional vectors, these form the columns of the n x m matrix (V), with m equal to the number of values in the dataset. This matrix is then factorized into a W: n x r matrix, and an H: r x m matrix. r will be smaller than n and m to form matrices smaller than the initial one. It is possible to approximate the function V through a cost function, using as measures of distance two non-negative matrices A and B, where A and B are not equal. The Euclidean distance between the two non-negative matrices will be

$$|A - B|^2 = \sum_{ij} (A_{ij} - B_{ij})^2 \tag{4}$$

2.3. Neural-ProdLDA

Neural-ProdLDA was introduced by Srivastava and Sutton in 2017 [3]. Neural-ProdLDA is an LDA model based on the Autoencoded Variational Inference (AEVB) [28] approach applied to the LDA method. This method tries to solve the problems related to the fact that the adaptation and the changes of the TM to adapt them to contexts that are particularly different from each other require new inference algorithms. A new topic model called ProdLDA is then presented. It replaces the mixture model in LDA with a product of experts.

The Neural-ProdLDA is an LDA model in which the word-level mixture over topics is performed in natural parameter space. In practical terms, compared with the classic LDA model, the parameter β is not normalized, and the distribution is defined as $w_n \mid \beta,\theta$ &$\sim$ Multinomial(1, $\sigma(\beta,\theta)$). From the results of the paper, [3] the Neural-ProdLDA reaches the topics concerning the classic LDA model, requiring, however, the use of the Autoencoded Variational Inference For Topic Mode (AVITM), which consists of the AEVB model applied to the LDA.

The Neural-ProdLDA model can be described as a product of experts, in which r and s are the natural parameters and δ is between [0, 1]. The formula describing the model is presented below.

$$\mathrm{P}(x|\delta r + (1-\delta)s \propto \prod_{i=1}^{N} \sigma(\delta r_i + (1-\delta)s_i)^{x_i} \propto \prod_{i=1}^{N} [r_i^{\delta} \cdot s_i^{1-\delta}]^{x_i} \tag{5}$$

2.4. Contextualized Topic Models (CTM)

Contextualized Topic Models (CTM) were introduced by Bianchi et al. in 2021 [4]. The authors proposed an extended version of the Neural-ProdLDA based on the Variational AutoEncoder (VAE) [8]. The Neural-Prod algorithm trains an inference network that directly maps the BoW into a continuous latent representation. A network decoder then reconstructs the BoW by generating words from the latent model of the document. This representation is sampled from a Gaussian distribution with parameters μ and θ^2. In this model, the authors replace the input of the BoW with pre-trained multilingual representations. Figure 2 shows a scheme regarding the functioning of the CTM model.

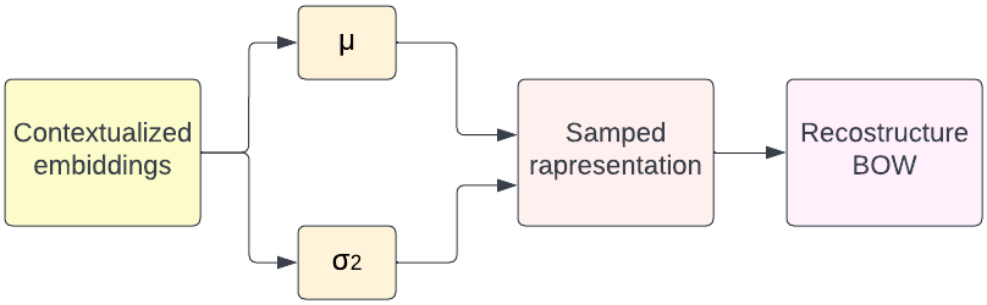

Figure 2. CTM schema.

2.5. Evaluation Metrics

In our study, we used four metrics to evaluate the effectiveness of different algorithms in the case study. We considered the OCTIS framework [29] for pre-processing, training, analysis, and comparison of TM methods (i.e., LDA, NMF, CTM, Neural-LDA) by the four evaluation metrics (diversity, similarity, coherence, and classification score). A schematization of the workflow of the OCTIS framework is presented in the following Figure 3.

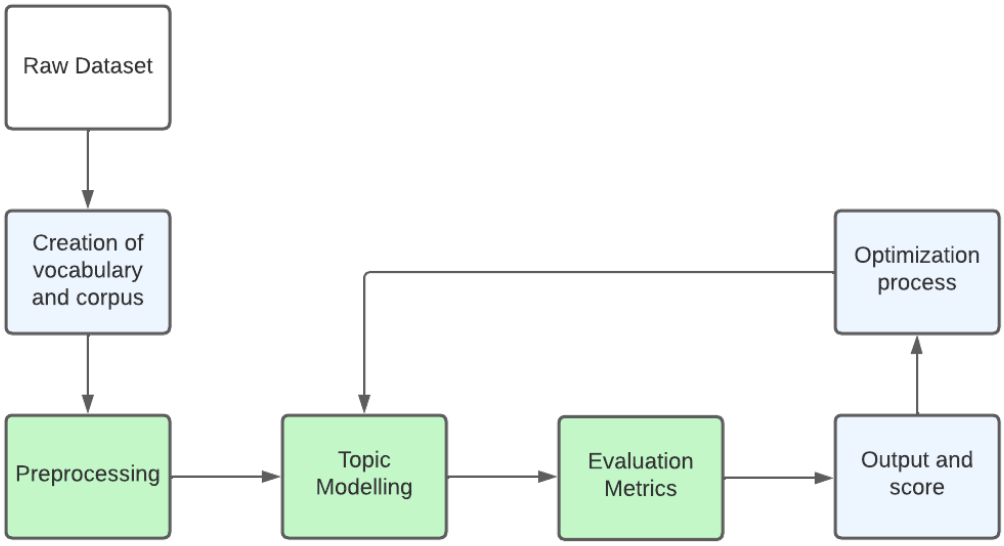

Figure 3. Workflow of the OCTIS framework.

The four evaluation metrics for TM model comparison are summarized in the following list.

- Similarity metrics, based on Ranked-Biased Overlap (RBO) measure [6]. By using this metric, the top words of each topic are compared by a similarity score, which ranges from 0 to 1. RBO is equal to 1 when all the top words of a topic are equal and in the same order, and vice versa. RBO is equal to 0 when the top words of the various topics are completely different from each other.
- Diversity metric [5]: this metric assigns a value based on how much the top-k words of the various topics differ from each other. It is based on the Inverse Ranked-Biased Overlap measure.
- Topic coherence metric [13,30]: this metric defines how the top-k words of a topic are related to each other. The metric ranges between 0 (low coherence) and 1 (high coherence).

- Classification score [31]: to examine the classification accuracy within the training data, we randomly divided the training set into five equal partitions and performed five-fold cross-validation.

  In particular, the extended RBO formula used in topic modeling is as follows:

$$\text{RBO} = \frac{X_d}{d} \cdot p^d + \frac{1-p}{p} \sum_{i=1}^{d} \frac{X_i}{i} \cdot p^i \tag{6}$$

where d is the evaluation depth and X_d is the intersection between two ranked list (T_1 and T_2) at depth d. p is the top-weightedness that controls the effect of the order. When p is 1, the order has no effect, and only the intersection is considered. Smaller p gives more weight to the order of words.

Topic coherence computes the sum of pairwise scores on the top n words $w_1, \ldots, w_n$ used to describe the topic, in the following formulae:

$$\text{Coherence} = \sum_{i<j} score(w_i, w_j) \tag{7}$$

The classification score is calculated as the number of true positives divided by the total number of true positives and false positives. The result ranges from a minimum of 0 to a maximum of 1.

$$\text{Classification score} = TruePositives/(TruePositives + FalsePositives) \tag{8}$$

3. Data Pre-Processing

After defining the TM algorithms included in the comparative study of this paper, it is necessary to investigate the data pre-processing techniques necessary to make the dataset more usable by the TM method. In particular, our process is based on techniques that allow the analysis of the Italian natural language. Generally, after the first step of data acquisition, the main pre-processing phases are defined as follows:

- Tokenization: a process of breaking down captured text data into words, called tokens. One of the most used tools in computer science to perform tokenization is the python library Natural Language Toolkit (NLTK).
- Lowercasing: the process of converting each word to its lowercase (WORD $\rightarrow$ word).
- Stopwords and punctuation removal: the phase in which the stopwords, i.e., meaningless terms such as articles and propositions, and punctuation are removed. In this way, it is possible to lighten the dataset from all those superfluous elements to evaluate the topics.
- Stemming: truncation of words at the root, omitting endings that indicate, for example, gender, number; alterations for nouns and adjectives; or mood, tense, or person for verbs. This makes sense if the root of the word is sufficiently representative of the meaning without the risk of ambiguity.
- Lemming: reduction of the word to a more canonical form—to be used as an alternative to stemming when you want to achieve maximum convergence of words—towards a common lemma that can grasp its meaning. Lemmatization operates at a considerably higher level of complexity than stemming. This phase turns out to be one of the most complex to implement for the Italian language, as there is no Python library capable of performing a correct lemmatization for the natural Italian language.

To better understand the effectiveness of the lemmatization process by comparing it with stemming, it is useful to report an example relating to one of the transcripts present in the dataset used in the case study. For example, conversation number 1 contains the term "sollecitato" (the Italian word for "hasten"). The result of the stemming process, carried out initially, was the following:

$$\underline{Stemming}$$
"sollecitato" → "sollecit"
"sollecito", "sollecitare" → "sollecit"

Below is the output of the lemmatization process for the word "sollecitare"

$$\underline{Lemming}$$
"sollecitato" → "sollecita"

In some cases, lemmatization can produce a verb in the infinitive form as a result of a word canceling the conjugation. The phase following the pre-processing is the creation of the BoW. It consists of the definition of the vocabulary that, using a weight for each word, allows the identification of the topics. The BoW is similar to a list of words without any order and grammatical rules. The main challenge faced during the pre-processing phase was the application of the methods mentioned above for the Italian natural language. The approach to face this problem is described in more detail in the subsequent section.

4. Case Study

The case study is a continuation of the work described in the paper [7]. A dataset consisting of real transcripts of phone conversations between operators (agents) of an assistance service and customers was used. The dataset of transcripts, in the Italian natural language, was provided by a real customer support center. The final goal of applying topic modeling in a case study relating to a customer center is to improve the work performance of real assistants. Knowing the topic almost immediately would allow operators to understand their customer's problems more efficiently, improving the quality of work and the customer experience.

4.1. Italian Pre-Processing

In this case, once the "raw" dataset was received, the pre-processing phase of the Italian language began, already partially described in the previous paper [7]. The algorithm was implemented partly in the Matlab environment and partly in the Python environment. The integration of the two environments took place through the use of particular application program interfaces (APIs). The entire pre-processing step performed is presented in the following Algorithm 1.

Algorithm 1. `Pre-processing`

for `m=1:M`
1. `tokenizedDocument`(d_m)
2. `lower`(d_m)
3. `erasePunctuation`(d_m)
4. `lemming`(d_m)
5. `removeWords`(d_m)/`removeStopWords`(d_m)
6. `bagOfWords`(D)
7. `tfidf`(BoW)

end for

Compared to our previous work in [7], for the present study, we considered lemmatization instead of stemming as the pre-processing and normalization step. The reason is that stemming algorithms convert words into stems that often are not valid word forms, but can still be used to group related words. Having lemmas instead of stems allows for greater quality in the TM process, thus obtaining clearer results from post-processing and resulting in visualization levels. The lemmatization stage was implemented by the Simplemma library (version 0.9.1). This library is a simple multilingual lemmatizer for Python that works on the level of individual words. It has support for different languages, including the Italian natural language. From our empirical study, we determined the Simplemma

library to be very simple, fast, and able to produce good results for the specific case study of this paper.

4.2. Models and Methods Implementation

To compare the performance of TM methods in our study, we used the OCTIS Python libraries. OCTIS is a framework for testing and training datasets currently available in the literature. Despite the OCTIS framework for the comparative tests of our study, it is worth noting that the pre-processing steps on the dataset were performed by a Python library specifically developed for our research. To apply the OCTIS framework, two files were used:

- vocabulary: a .txt file where each line represents a word of the vocabulary.
- corpus: a .tsv file (tab-separated) that contains up to three columns, i.e., the document, the partition, and the label associated with the document (optional).

By the OCTIS framework, we compared the performance of the following TM methods: Latent Dirichlet Allocation, Non-negative Matrix Factorization, Neural-ProdLDA, and Contextualized Topic Modeling. We also conducted experiments for three different values of the topic number k equal to 10, 20, and 25, obtaining the results reported in Tables 1–3, respectively.

Performance comparison was made possible by using four metrics, namely, diversity (refer to Table 4), similarity metrics (refer to Table 5), Coherence metrics (refer to Table 6), and Classification metrics (refer to Table 7). In each table, performance is also reported by varying the number of topics k in each method.

From the results in Tables 4–7, it can be observed that the LDA is the outperforming method for the TM process in our case study. Given that the diversity and similarity metrics range between 0 and 1, we identify optimal values as those that guarantee a satisfactory degree of similarity between the words contained in the topics. The Coherence and Classification metrics also indicate the LDA method as the outperforming approach in our case study. Therefore, the LDA combined with $k = 20$ was the outperforming TM algorithm for the case study of this paper.

Table 1. Top five words for five topics, using $k = 10$.

LDA 1: fax bonifico emettere bolletta fattura
LDA 2: fax bonifico bolletta gas tariffa
LDA 3: tariffa IBAN bolletta energia attivazione
LDA 4: fax bolletta gas luce autolettura
LDA 5: interesse bonifico ultimare presidente saldo
CTM 1: interesse attaccare presidente saldo creare
CTM 2: sbloccare chilowatt voir pratica aggiuntivo
CTM 3: gas vento tariffa modulare conguaglio
CTM 4: contare dettare bolletta pregare attaccare riallacciare
CTM 5: elettronico acqua nota addebito insistere
NMF 1: voltura pratica bonifico attivazione saldo
NMF 2: banca autolettura energia bolletta tariffa
NMF 3: bolletta energia luce autolettura attivazione
NMF 4: bolletta savoir profilo modulo pagare
NMF 5: ricordo bolletta mattina indirizzo documento
Neural-LDA 1: ultimare ufficacia disdetta identità prezzo
Neural-LDA 2: nota riscaldamento documentazione disdire pervenire
Neural-LDA 3: fax luce bonifico salva pensa
Neural-LDA 4: detta dice emette sbaglio pagare
Neural-LDA 5: metano interesse papà caldaia avviso

Table 2. Top ten words for five topics, using $k = 20$.

LDA 1: prezzo bonifico gas proporre conto
LDA 2: fax bolletta gas fattura bonifico
LDA 3: bolletta fax bonifico fattura luce
LDA 4: fax gas bolletta bonifico IBAN
LDA 5: fax bonifico bolletta voltura energia
CTM 1: alza comunicazione conguaglio caldaia mangia
CTM 2 fax allacciare preferire bonifico cordoglio
CTM 3: eventuale approva bonifico accredito inverno
CTM 4: raccomandata fatturare stipendio sbaglio plastica
CTM 5: biro biologo disc meccanismo residenza
NMF 1: bonifico gas scusa fattura IBAN
NMF 2: fax riallaccio fermo combinazione luce
NMF 3: fax gas bolletta ascolto savoir
NMF 4: bolletta puro bonifico luce rimborso
NMF 5: dettare fattura fax risulta bolletta
Neural-LDA 1: scuola contratto rotto spelling mora
Neural-LDA 2: appuntamento attivazione mattina ufficiare energia
Neural-LDA 3: comunicazione acqua lavoro inizia documenta
Neural-LDA 4: fornitore conto interesse gas IBAN
Neural-LDA 5: nullo fossa metano indirizzo ascolta

Table 3. Top ten words for five topics, using $k = 25$.

LDA 1: fax dettare fattura bolletta luce
LDA 2: bolletta emettere contrarre fax gas
LDA 3: fattura fax bollettino autolettura ringrazia
LDA 4: bolletta emettere dettare fax gas
LDA 5: fax emettere bolletta gas IBAN
CTM 1: registra consumo randomica rateizzazione gas
CTM 2: raccomandata tangere ascolto fattura luce
CTM 3: nostro raccomandata disdire prevede fax
CTM 4: patente salute Vodafone giornale bolletta
CTM 5: luce fax giornale bolletta gas
NMF 1: voltura accorda fattura contare bollettino
NMF 2: ricorda dettare voltura fax costo
NMF 3: servire indirizzo fax attivazione scadenza
NMF 4: dettare gas fax appuntamento mètre
NMF 5: autolettura gas energia consumare fattura
Neural-LDA 1: intestatario tesare IBAN occorre traduzione
Neural-LDA 2: affatto perfetta battere contante Romagna supporto
Neural-LDA 3: fax tangere consumo autolettura emettere
Neural-LDA 4: quarantott affatto risorsa contante IBAN
Neural-LDA 5: accordare funzionare facile risorsa consegna

Table 4. Topic Diversity results.

	k = 10	**k = 20**	**k = 25**
LDA	0.2	0.185	0.188
CTM	0.84	0.875	0.84
NMF	0.38	0.435	0.364
Neural-LDA	0.895	0.81	0.884

The CTM model shows small values of similarity, which contrasts the too-high degree of diversification. Consequently, the topics identified will be excessively different from each other. The Neural-LDA model has similarity values close to 0 while guaranteeing a high diversification. The Topic Coherence, on the other hand, shows lower values than the

LDA method, underlining the lower reliability of the method in the case study. Finally, the Classification shows values similar to those relating to the CTM method, albeit not close to an optimal level. The Neural-LDA is the least-performing method for the dataset used in all metrics, showing unreliable results, especially concerning Topic Coherence.

Table 5. Similarity results.

	k = 10	**k = 20**	**k = 25**
LDA	0.47	0.52	0.494
CTM	0.012	0.01	0.010
NMF	0.146	0.14	0.130
Neural-LDA	0.007	0.017	0.006

Table 6. Topic Coherence results.

	k = 10	**k = 20**	**k = 25**
LDA	0.47	0.63	0.56
CTM	0.37	0.53	0.48
NMF	0.41	0.59	0.57
Neural-LDA	0.08	0.18	0.10

Table 7. Classification results.

	k = 10	**k = 20**	**k = 25**
LDA	0.24	0.39	0.29
CTM	0.21	0.37	0.26
NMF	0.18	0.34	0.26
Neural-LDA	0.05	0.22	0.17

The NMF model has acceptable diversity values in the $k = 25$ configuration; however, at the same time, the degree of similarity is too low to be able to define the outputs as performing. Concerning the Coherence and Classification metrics, we can observe a better performing configuration with values of $k = 20$. Compared with the CTM model, the Coherence is even higher, guaranteeing a greater degree of reliability. The resulting trend for each metric is depicted in Figure 4 for diversity, Figure 5 for similarity, Figure 6 for coherence, and Figure 7 for classification.

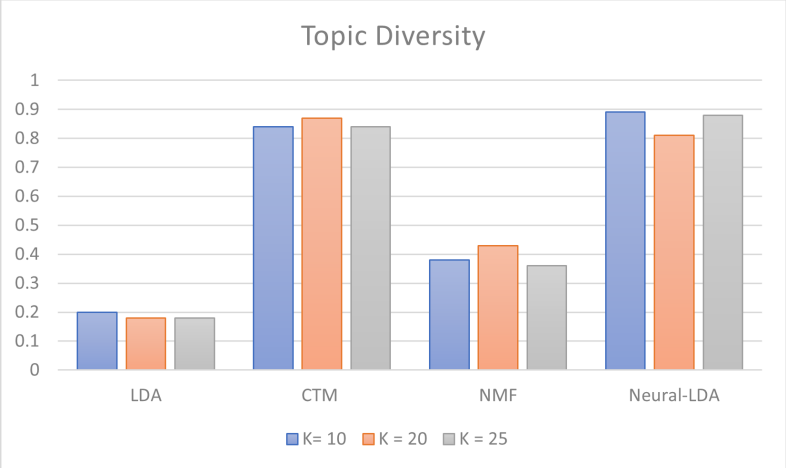

Figure 4. Topic Diversity Bar graph.

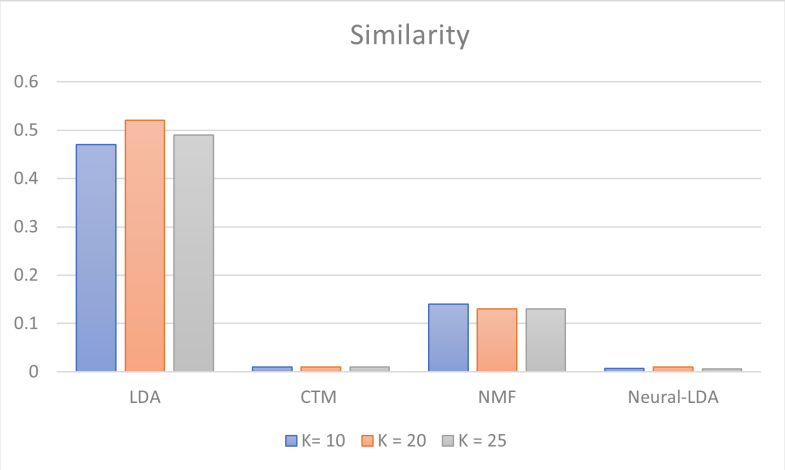

Figure 5. Similarity Bar graph.

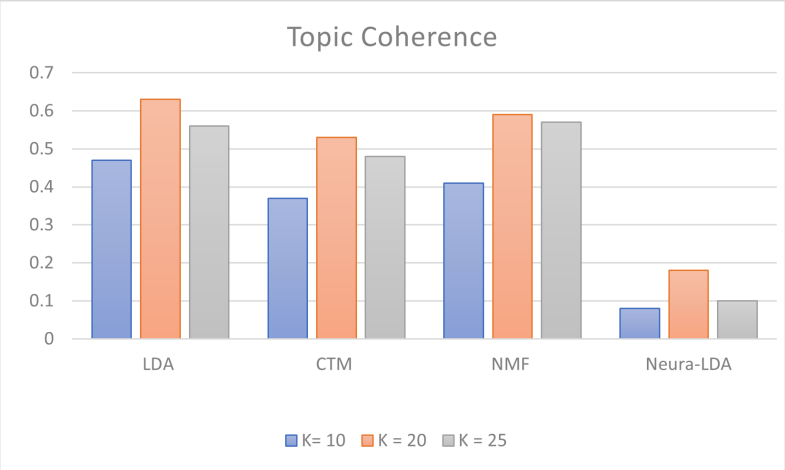

Figure 6. Topic Coherence Bar graph.

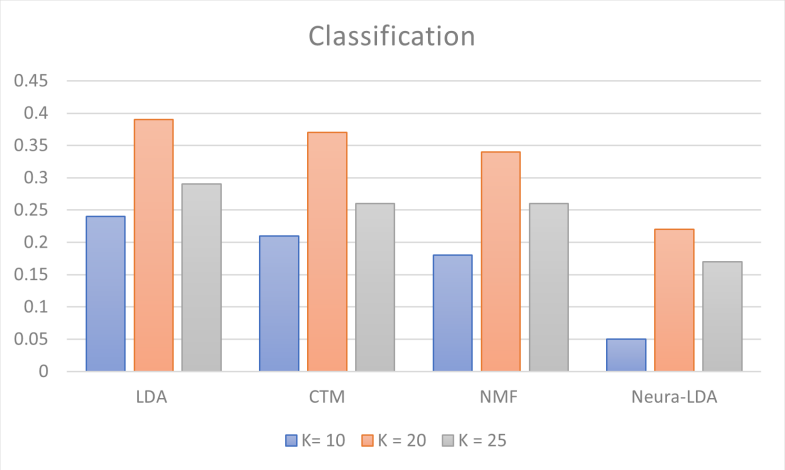

Figure 7. Classification Bar graph.

The process–work schema in Figure 8 presents an overview of the work performed to arrive at the desired post-processing outputs. Given the collection of 937 transcriptions in the Italian language, the dataset was subjected to a series of pre-processing procedures to optimize the input necessary for the TM. In particular, compared with the previous paper, the process has been optimized through the use of lemmatization instead of the stemming

phase. This has made it possible to obtain result words that have a complete meaning and are no longer truncated, consequently also improving the results of post-processing. The processing of data in Italian required the specific use of the Simplemma Python library [32].

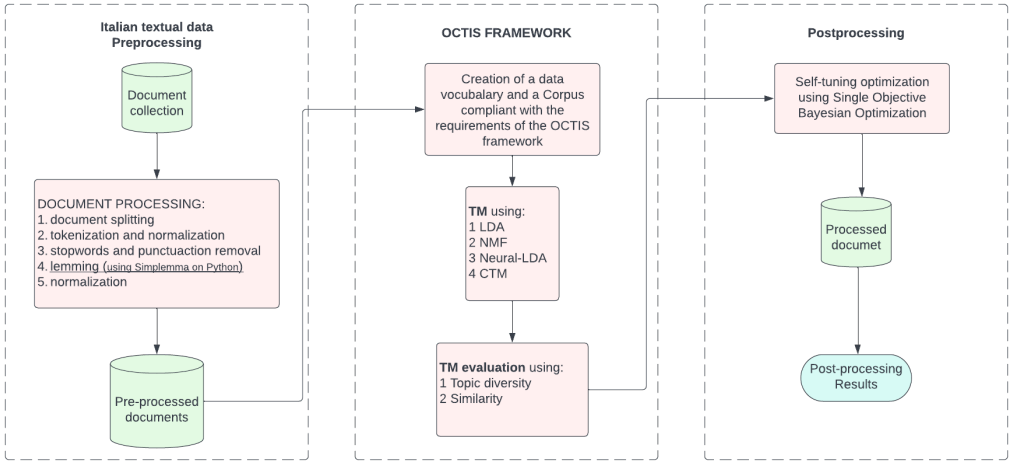

Figure 8. Process–work schema of 3 steps (from left to right): (1) pre-processing of the data in Italian natural language; (2) OCTIS framework; (3) post-processing for TM.

Simplemma is a Python library that allows the user to perform lemmatization through a simplified and multilingual approach (48 languages including Italian), allowing them to create a raw series of tokens without the need to have morphosyntactic information. This library was born from the algorithms and studies presented in [33–35]. Although not optimized for all 48 languages available, it is a functional and fast method for most cases, including Italian.

Once the pre-processed files were obtained, a further step was implemented to use the OCTIS framework through the Python library. In this step, the dataset of 937 transcriptions was partitioned into three subsets: training, validation, and testing data. In particular, 60% of data were used for training, 20% for validation, and 20% for testing.

Furthermore, the OCTIS framework required the creation of .txt files corresponding to a vocabulary in the Italian natural language, which is useful for processing the TM. Then, the processing phase started via the Python library. The first step involved the definition of the Topic Models to be used. The choice fell on the LDA, CTM, NMF, and Natural-LDA methods because, following the tests performed in the method evaluation phase, they resulted to be the most suitable to operate in our real case study, which involved using real conversations in the Italian natural language. The optimization took place through the standard method proposed by the library used or through the Single Objective Bayesian Optimization. The evaluation metrics of the implemented models were then defined. Having chosen the use of the Perplexity metric in the previous work, in this case, we chose to use the Topic Diversity and Similarity metrics.

At this point, the data processing took place, which returned the various outputs presented and discussed in the following section. In general, the TM returns two main outputs:

- Top k-words topic words: i.e., the identification of the most representative words for each of the topics discovered.
- Topic document distributions: the possibility that every real Italian conversation, extrapolated from the dataset, belongs to a specific topic among those identified.

4.3. Post-Processing and Results Discussion

The results of data post-processing fall within the Topic Visualization, i.e., all those useful tools to better understand the results of the TM visually. Once the results of the first tests were obtained, it was evident that it was necessary to optimize the results and to carry

out further steps to improve the dataset. For example, all the words with a percentage of use that was too high or too small were removed. In addition, all non-text characters—i.e., numbers or symbols—and some words out of context, not very useful for the final result, were also removed—such as the Italian term "daccapo", which in English corresponds to "again".

From the results obtained in the three tests relating to the use of RBO metrics, the configuration with $k = 20$ appears to be the most stable both in terms of diversity and similarity (RBO and IRBO, or Inverse Rank Biased Overlap). For this reason, the results of the post-processing with k equal to $k = 20$ are discussed in the following section.

Figure 9 shows the results of the Topic–document probability; the topic that most likely (39.49%) represents one of the 937 conversations contained in the dataset is represented.

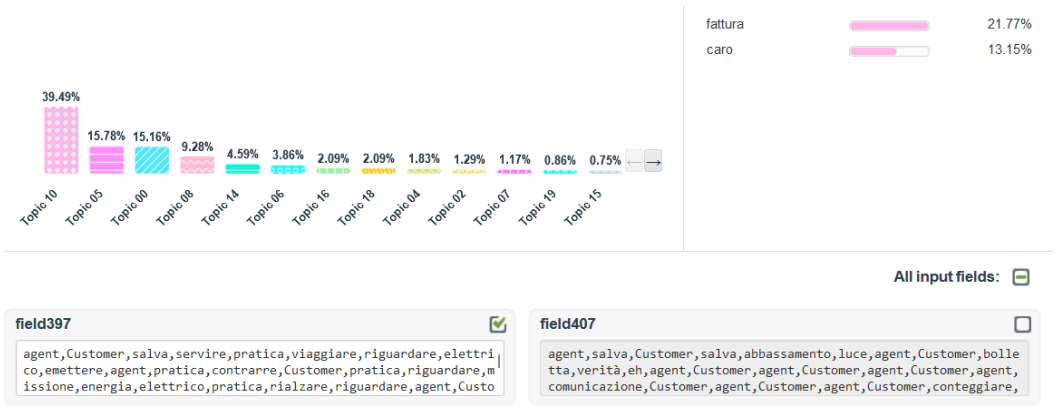

Figure 9. Topic Distribution for transcription no. 397, with $k = 20$.

The figure shows the probability of a specific conversation (transcription no. 397) belonging to a topic, showing through a bar graph (top left) the percentage probability in descending order of the topic distribution. Further, by clicking on a specific topic, it is possible to obtain the top two k-words relating to that topic. It is also important to underline that words such as "Customer" and "Agent" were not considered for the topic document distribution step, as they have too high a percentage of use within the dataset.

Topic modeling quality can be graphically assessed by using the multidimensional scaling (MDS) method [36] of the following Figure 10. From Figure 10, it appears that, as a result of the 20-topic model, the topics spread through unit spaces but also cluster in small groups of two or three. From our empirical study, we observed that this result mainly depends on the specific case study of customer care service, in which groups of 2 or 3 topics may share technical keywords. One example is given by topics 8 and 10, and similarly by topics 19 and 14. In a few cases, topics do not cluster with the others, such as for topics 6 and 12.

The word cloud graph for topic numbers 10 and 8 are depicted in Figures 11 and 12, respectively. The word cloud is a Topic Visualization tool in which the words that best represent a specific topic are graphically emphasized, through the use of more striking colors and higher textual characters. From Figures 11 and 12, it can be observed that the two topics share different top words, such as "bolletta" (the Italian for "bill") or "gas". The MSD method also showed that topics 10 and 8 cluster together. The word cloud of topic 10 represents the topic with the highest possibility of being relating to the transcriptions in the case study of this work.

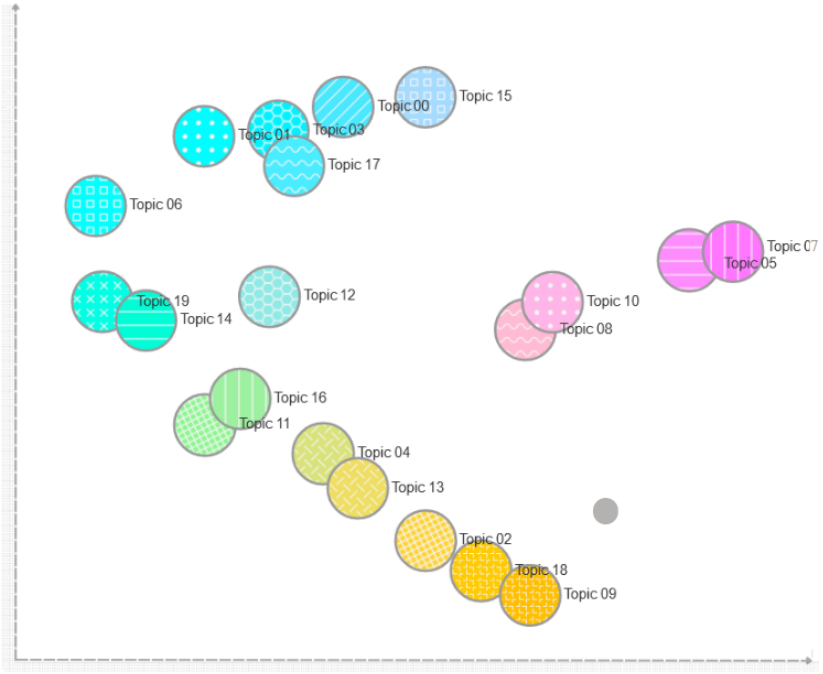

Figure 10. Inter-topic distances in a two-dimensional space by the multidimensional scaling (MDS) method.

Figure 11. World Cloud of topic 10.

Figure 12. World Cloud of topic 8.

Furthermore, topic 19 and topic 14, which cluster together in the MDS representation but spread away from topic 8 and topic 10, are represented by a word cloud in Figures 13 and 14, respectively. It can be observed that these two topics differ from topic 8 and topic 10, in fact, they do not share different top words (such as "bolletta" or "gas").

Figure 13. World Cloud of topic 19.

From an application point of view, the use of this approach brings significant advantages to the user of a TM system. In fact, in the contact center environment, a manager employed in customer care management would have the possibility, through the analysis of the TM outputs, to obtain fundamental information regarding the different problems relating to different calls. Consequently, this would lead to an increase in the quality of the service as well as an improvement in company performance.

It is important to underline how the results for the post-processing of data relating to the TM of conversations in the Italian natural language were extrapolated through the application of the OCTIS framework and the use of Matplotlib Python libraries for the implementation of models from the Matlab environment.

Figure 14. World Cloud of topic 14.

Ultimately, the output of the TM returns useful data to evaluate the most suitable and efficient method to evaluate the most suitable metric, also allowing a comparison with the results obtained in the paper [7]. The results show that the LDA model is once again the most reliable in our specific case for TM operations with metrics other than Perplexity.

The lower diversity value and a higher degree of similarity suggest that, for the LDA model, the greatest degree of coherence is obtained between the terms that characterize the different topics identified—in particular, for the test relating to a value of $k = 20$, guaranteeing ideal performance for the case study treated. The values obtained show a reasonable degree of correspondence between the terms contained in the various topics. At the same time, similarity values that are not too large also confirm a suitable level of diversification.

5. Conclusions

In this study, we focused on the use of TM methods for the Italian language. A methodology based on pre-processing implemented partly in Matlab and partly in Python has been proposed. The pre-processing involved a lemmatization step, which is particularly complicated for the Italian language.

The implementation of four TM algorithms—namely, LDA, CTM, NMF, and NeuralLDA models—was performed using the OCTIS Python library, which allowed to conduct automatic optimization through the Single-Objective Bayesian Optimization. Performances of the TM algorithms were evaluated using two metrics based on the rank-biased overlap: Topic Diversity and Similarity. The goal was to obtain an optimal configuration that would guarantee good coherence between the k-words of the identified topics and, at the same time, guarantee an adequate degree of diversity. Three tests relating to the TM were performed, based on the identification of several topics (k) equal to 10, 20, and 25, from which it emerged that the optimal configuration was the one based on the LDA method and the use of a k equal to $k = 20$. The similarity value turned out to be optimal (with a value equal to 0.5) while guaranteeing an acceptable diversity value (about 0.2) at the same time. The results were presented graphically using Topic Visualization tools. In particular, the Topic Distribution, which highlights the percentage of belonging of a given conversation to a specific topic among those identified. Furthermore, word clouds were created to highlight the words with the greatest weight within a given topic.

The main limitation of the present study is related to the difficulty of generalizing findings from our case study to other settings. Case studies are performed in real-world settings and, consequently, they have a high degree of realism, mainly at the expense of the level of control. Despite the fact that the case study of the present work may not generate the same insights as controlled experiments do, our study may provide an understanding of the best-performing TM algorithm. The research result can be exploited by practitioners to find the best approach for topic modeling of human-to-human conversation transcriptions. Therefore, this work can be an asset for grounding applications of topic modeling and can be inspiring for similar case studies in the domain of customer care quality.

A possible extension of the current study could involve further investigation of the data analyzed. In particular, it would be useful to investigate methods that allow the exploitation of voice sources in real-time and integrated data from Web and Social sources.

Author Contributions: Conceptualization, M.P. (Massimo Pacella); methodology, M.P. (Massimiliano Perrone); software, M.P. (Massimiliano Perrone); validation, M.P. (Massimo Pacella) and G.P.; resources, G.P. and V.G.; writing—original draft preparation, G.P.; writing—review and editing, M.P. (Massimo Pacella), M.P. (Massimiliano Perrone), G.P.; supervision, G.P. and M.P. (Massimo Pacella); project administration, G.P. and V.G.; and funding acquisition, G.P. and V.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been funded by Puglia Region (Italy)—Project "VOice Intelligence for Customer Experience (VO.I.C.E. First)".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The actual transcriptions from the customer support center are not publicly available due to privacy concerns. The list of stopping words for algorithm implementation of the Italian natural language is available on request as a supplement to the present paper.

Acknowledgments: The authors are thankful to Teleperformance S.p.A. (Italy) for providing the dataset for the case study.

Conflicts of Interest: The authors declare no conflict of interest. The funder had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent Dirichlet allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
2. Lee, D.; Seung, H.S. Algorithms for Non-negative Matrix Factorization. In *Proceedings of the Advances in Neural Information Processing Systems*; Leen, T., Dietterich, T., Tresp, V., Eds.; MIT Press: Cambridge, MA, USA, 2000; Volume 13.
3. Srivastava, A.; Sutton, C. Autoencoding Variational Inference For Topic Models. *arXiv* **2017**, arXiv:1703.01488.
4. Bianchi, F.; Terragni, S.; Hovy, D.; Nozza, D.; Fersini, E. Cross-lingual Contextualized Topic Models with Zero-shot Learning. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, Online, 19–23 April 2021.
5. Dieng, A.B.; Ruiz, F.J.; Blei, D.M. The dynamic embedded topic model. *arXiv* **2019**, arXiv:1907.05545.
6. Webber, W.; Moffat, A.; Zobel, J. A similarity measure for indefinite rankings. *ACM Trans. Inf. Syst. (TOIS)* **2010**, *28*, 1–38. [CrossRef]
7. Papadia, G.; Pacella, M.; Giliberti, V. Topic Modeling for Automatic Analysis of Natural Language: A Case Study in an Italian Customer Support Center. *Algorithms* **2022**, *15*, 204. [CrossRef]
8. Churchill, R.; Singh, L. The evolution of topic modeling. *ACM Comput. Surv.* **2022**, *54*, 1–35. [CrossRef]
9. Nigam, K.; McCallum, A.K.; Thrun, S.; Mitchell, T. Text classification from labeled and unlabeled documents using EM. *Mach. Learn.* **2000**, *39*, 103–134. [CrossRef]
10. Blei, D.; Lafferty, J. Correlated topic models. In Proceedings of the NIPS'06, Vancouver, BC, Canada, 4–7 December 2006; Volume 18, p. 147.
11. Dieng, A.B.; Ruiz, F.J.; Blei, D.M. Topic modeling in embedding spaces. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 439–453. [CrossRef]
12. Bianchi, F.; Terragni, S.; Hovy, D. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence. *arXiv* **2020**, arXiv:2004.03974.
13. Lau, J.H.; Newman, D.; Baldwin, T. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In Proceedings of the EACL'14, Gothenburg, Sweden, 26–30 April 2014; pp. 530–539.
14. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 3111–3119.
15. Xia, L.; Luo, D.; Zhang, C.; Wu, Z. A survey of topic models in text classification. In Proceedings of the 2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD), Chengdu, China, 25–28 May 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 244–250.
16. Likhitha, S.; Harish, B.; Kumar, H.K. A detailed survey on topic modeling for document and short text data. *Int. J. Comput. Appl.* **2019**, *178*, 1–9. [CrossRef]
17. Abdelrazek, A.; Eid, Y.; Gawish, E.; Medhat, W.; Hassan, A. Topic modeling algorithms and applications: A survey. *Inf. Syst.* **2022**, *112*, 102131. [CrossRef]
18. Liu, Z.; Ng, A.; Lee, S.; Aw, A.T.; Chen, N.F. Topic-aware pointer-generator networks for summarizing spoken conversations. In Proceedings of the 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Singapore, 14–18 December 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 814–821.
19. Tur, G.; De Mori, R. *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*; John Wiley & Sons: Hoboken, NJ, USA, 2011.
20. Jelodar, H.; Wang, Y.; Yuan, C.; Feng, X.; Jiang, X.; Li, Y.; Zhao, L. Latent Dirichlet Allocation (LDA) and Topic modeling: Models, applications, a survey. *Multimed. Tools Appl.* **2019**, *78*, 15169–15211. [CrossRef]
21. Hazen, T.J. Chapter 12: Topic identification. In *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*; John Wiley & Sons: Hoboken, NJ, USA, 2011; Volume 12, pp. 319–356.
22. Zhao, G.; Zhao, J.; Li, Y.; Alt, C.; Schwarzenberg, R.; Hennig, L.; Schaffer, S.; Schmeier, S.; Hu, C.; Xu, F. MOLI: Smart conversation agent for mobile customer service. *Information* **2019**, *10*, 63. [CrossRef]
23. Pota, M.; Ventura, M.; Catelli, R.; Esposito, M. An effective BERT-based pipeline for Twitter sentiment analysis: A case study in Italian. *Sensors* **2020**, *21*, 133. [CrossRef] [PubMed]

24. Agostino, D.; Brambilla, M.; Pavanetto, S.; Riva, P. The contribution of online reviews for quality evaluation of cultural tourism offers: The experience of Italian museums. *Sustainability* **2021**, *13*, 13340. [CrossRef]
25. Aria, M.; Cuccurullo, C.; D'Aniello, L.; Misuraca, M.; Spano, M. Thematic analysis as a new culturomic tool: The social media coverage on COVID-19 pandemic in Italy. *Sustainability* **2022**, *14*, 3643. [CrossRef]
26. Murdock, J.; Allen, C. Visualization Techniques for Topic Model Checking. In Proceedings of the AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015; Volume 29.
27. Maier, D.; Waldherr, A.; Miltner, P.; Wiedemann, G.; Niekler, A.; Keinert, A.; Pfetsch, B.; Heyer, G.; Reber, U.; Häussler, T.; et al. Applying LDA topic modeling in communication research: Toward a valid and reliable methodology. *Commun. Methods Meas.* **2018**, *12*, 93–118. [CrossRef]
28. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes. *arXiv* **2013**, arXiv:1312.6114.
29. Terragni, S.; Fersini, E.; Galuzzi, B.G.; Tropeano, P.; Candelieri, A. Octis: Comparing and optimizing topic models is simple! In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Online, 19–23 April 2021; pp. 263–270.
30. Röder, M.; Both, A.; Hinneburg, A. Exploring the space of topic coherence measures. In Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, Shanghai, China, 2–6 February 2015; pp. 399–408.
31. Phan, X.H.; Nguyen, L.M.; Horiguchi, S. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In Proceedings of the 17th International Conference on World Wide Web, Beijing, China, 21–25 April 2008; pp. 91–100.
32. Simplemma: A Simple Multilingual Lemmatizer for Python [Computer Software]. Available online: https://github.com/adbar/simplemma (accessed on 11 December 2022).
33. Barbaresi, A.; Hein, K. Data-driven identification of German phrasal compounds. In Proceedings of the International Conference on Text, Speech, and Dialogue, Prague, Czech Republic, 27–31 August 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 192–200.
34. Barbaresi, A. An unsupervised morphological criterion for discriminating similar languages. In Proceedings of the 3rd Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2016), Osaka, Japan, 12 December 2016; pp. 212–220.
35. Barbaresi, A. Bootstrapped OCR error detection for a less-resourced language variant. In Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016), Bochum, Germany, 19–21 September 2016; pp. 21–26.
36. Guo, L.; Li, S.; Lu, R.; Yin, L.; Gorson-Deruel, A.; King, L. The research topic landscape in the literature of social class and inequality. *PLoS ONE* **2018**, *13*, e0199510. [CrossRef] [PubMed]