CluHTM: Exploiting Semantic Word Clustering Representation for Non-Probabilistic Hierarchical Topic Modeling

Anonymous Author(s)*

ABSTRACT

Hierarchical Topic modeling (HTM) exploits latent topics and relationships among them as a powerful tool for data analysis and exploration. Despite its advantages over traditional topic modeling, HTM poses its own challenges, such as (1) incoherence of topics; (2) unreasonable structure; and (3) definition of the “ideal” number of topics and depth of the hierarchy. In this paper, we advance the state-of-the-art of HTM by means of the design, development and evaluation of CluHTM, a novel non-probabilistic hierarchical matrix factorization aimed at solving the specific issues of HTM. ClhuHTM’s novel contributions include: (i) the exploration of a richer text representation that preserves, embeds and takes advantage of both, *global (dataset level)* and semantic information, helping to solve the*topic coherence* problem as well as issues related to the *unreasonable structure*, which plague alternatives that only exploit *local information (upper level layers of the hierarchy)*; (ii) the exploitation of a stability analysis metric in a cross-level fashion for defining the number of topics and ultimately the ‘shape’ the structure of the hierarchy. In our experimental evaluation, considering 12 datasets, 7 state-of-the-art baselines, and three classes of topics quality metrics: (i) coherence; (ii) mutual information and (iii) semantic representation, CluHTM outperformed the baselines in the vast majority of the cases, with gains of around 500% over the strongest state-of-the-art baselines. Finally, we provide qualitative and quantitative statistical analyses of why our solution works so well.

KEYWORDS

Hierarchical Topic Modeling, Non-Probabilistic Topic Modeling, Data Representation, Word Embedding

ACM Reference Format:
Anonymous Author(s). 2019. CluHTM: Exploiting Semantic Word Clustering Representation for Non-Probabilistic Hierarchical Topic Modeling. In *WSDM '20, February 3 - 7, 2020, Houston, Texas.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/XXXXXXX.XXXXXX

1 INTRODUCTION

Topic Modeling (TM) is the task of automatically extracting latent topics (e.g., a concept or a theme) from a collection of textual documents. Such topics are usually defined as a probability distribution over a fixed vocabulary (a set of words) that refers to some subject and defines the latent topic as a whole. Topics might be related to each other and if they are defined at different semantic granularity levels (more general or more specific), this naturally induces a hierarchical structure. Although traditional TM strategies are of great importance to extract latent topics, the relationships among them are also paramount for data analysis and exploration. This is what Hierarchical Topic Modeling (HTM) aims to achieve. HTM is a machine learning technique whose goal is to induce latent topics from text data while preserving the inherent hierarchical structure [27]. Important scenarios have been shown to enjoy the usefulness of HTM, such as (i) hierarchical categorization of Web pages [17], (ii) extracting aspects hierarchies in reviews [7] and (iii) discovering research topics hierarchies in academic repositories [20].

Despite the practical importance of HTM and their advantages over traditional TM, HTM poses its own challenges to guarantee its proper application, the main ones being: (i) *Incoherence of topics* and (ii) *Unreasonable structure. Incoherence of topics* has to do with the need to learn meaningful topics. That is, the top words that represents a topic have to be semantically consistent with each other. *Unreasonable structure* is related to the extracted hierarchical topic structure. Topics near the root should be more general, while topics close to the leaves are more specific. Furthermore, child topics must be coherent with their corresponding parent topics, guaranteeing reasonable hierarchical structure. Finally, (iii) the number of topics in each hierarchy level is usually unknown and cannot be previously set to a predefined value since it directly depends on the latent topical distribution of the data.

Both supervised and unsupervised approaches have been applied for HTM. Supervised approaches use prior knowledge to build the hierarchical tree structure, such as labeled data or linking relationship among documents [30]. Those strategies are unfeasible when there is no explicit taxonomy or hierarchical scheme to associate with documents or when such an association (a.k.a., labeling) is very cumbersome or costly to obtain. Unsupervised HTM (uHTM) deals with such limitations. uHTM methods do not rely on previous knowledge (such as taxonomies or labeled hierarchies), having the additional challenge of discovering the hierarchy of topics based solely on the data at hand. State-of-the-art uHTM strategies explore some type of semantic knowledge (such as semantic correlation among words) [31] usually based on word co-occurrences. The derivation of the hierarchy takes advantage of sets of words that are likely to belong to the same topic at a certain level of the hierarchy.

HTM solutions can be roughly grouped into non-probabilistic and probabilistic models. In probabilistic modeling strategies, textual data is considered to be “ruled” by an unknown probability distribution that governs the relationships between documents and topics, in a hierarchical fashion. The major drawback in this type of approach has to do with the number of parameters in the model, which rapidly grows with the number of documents. This leads to learning inefficiencies and proneness to over-fitting. While approaches such as hLDA and its variants have been shown to excel on HTM tasks, they still suffer from the aforementioned learning inefficiencies due

WSDM '20, February 3 - 7, 2020, Houston, Texas, USA

ACM ISBN 978-1-4503-6317-4/19/07.
https://doi.org/10.1145/XXXXXXX.XXXXXX

to the large number of parameters to be learned and poor handling of short text data, as theoretically demonstrated in [26].

To overcome these drawbacks, non-probabilistic models aim at extracting hierarchical topic models by means of matrix factorization techniques instead of learning probability distributions. Such strategies also pose their own challenges. For instance, they are usually limited to just local information (i.e., data limitation) as they go deeper into the hierarchy structure when extracting the latent topics. That is, as one moves deeper in the hierarchical structure representing the latent topics, the available data rapidly reduces in size, directly impacting the quality of extracted topics (in terms of both, coherence and structure reasonableness). This phenomenon is mitigated by probabilistic models as they rely on global information when handling the probability distributions[31].

Although matrix factorization models are more learning efficient and less prone to over-fitting than their probabilistic counterparts in traditional TM tasks, they lack the desired (even necessary) coherence and reasonableness properties enjoyed by probabilistic models in the HTM context. This is especially true at deeper levels in the hierarchy structure. Because of that, the current main HTM methods are built based on probabilistic methods [4, 16].

In this paper, we aim at exploring the best properties of both non-probabilistic and probabilistic strategies while, at the same time, mitigating their main drawbacks. Up to our knowledge, the only work to explore this research venue is [11]. In that work, the authors explore NMF for solving HTM tasks by enforcing three optimization constraints during matrix factorization: global independence, local independence, and information consistency. Those constraints allow their strategy, named HSOC, to come up with hierarchical topics that somehow preserve topic coherence and reasonable hierarchical structures. However, as we shall see in our experimental evaluation, HSOC is still not capable of extracting coherent topics when applied to short text data, which is currently prominent of the Web, especially on social network environments.

We here propose a totally distinct approach, taking a data engineering perspective, instead of focusing on the optimization process. More specifically, we explore a matrix factorization solution properly designed to explore global information (akin to probabilistic models) when learning hierarchical topics while ensuring proper topic coherence and structure reasonableness. This allows us to build a learning efficient HTM strategy, less prone to over-fitting that also enjoys the desired properties of topic coherence and reasonable (hierarchical) structure. We do so by applying a matrix factorization method over a richer text representation that preserves and embeds both, global and semantic information when extracting the hierarchical topics.

Recent non-probabilistic methods [25, 28] have produced top-notch results on traditional TM tasks by taking advantage of semantic similarities obtained from the distances between words and their positioning within an embedding space [14, 21]. Our key insight for HTM was to note that the richer (semantic) representation offered by word embeddings can be readily explored as a global[1] source of information in deeper levels of the hierarchical structure of topics. This gives us a key building block to overcome the challenges of matrix factorization strategies for HTM without the need for additional optimization constraints.

In [28], the authors exploit the nearest words of a given "pre-trained" word embedding to generate "meta-words" able of enhancing the document representation, in terms of syntactic and semantic information, named CluWords. Such an enhanced representation is capable of mitigating the drawbacks of using the projected space of word embeddings as well as extracting cohesive topics when applying non-negative matrix factorization for topic modeling.

Motivated by this finding, we here advance the state-of-the-art in HTM, by means of the design, development and evaluation of an unsupervised non-probabilistic HTM method that exploits the CluWords as their key properties for TM when capturing the latent hierarchical structure of topics. As we shall detail later, we focus on the NMF method for uncovering the latent hierarchy as it is the most effective matrix factorization method for our purposes. Finally, the last aspect needed to be addressed for successful use of NMF for HTM is the definition of the appropriate number of topics k to be extracted. Choosing too few topics will produce overly broad results while choosing too many will result in over-clustering the data into many redundant, highly-similar topics. Thus, our proposed method uses a stability analysis concept to automatically select the best number of topics for each level of the hierarchy.

As we shall see, our approach outperforms HSOC and HLDA (current state-of-the-art) for both short and large text datasets, often by large margins. To summarize, our main contributions are:

- a novel non-probabilistic HTM strategy, based on NMF and CluWords that excels on HTM tasks (in both short and large text data) while ensuring topic coherence and reasonable topic hierarchies.
- the exploitation in an original way of a cross-level stability analysis metric for defining the number of topics and ultimately 'the shape' of the hierarchical structure; as far as we know this metric has never been applied with this goal.
- an extensive empirical analysis of our proposal considering 12 datasets and 7 state-of-the-art baselines. In our experimental evaluation, CluHTM outperformed the baselines in the vast majority of the cases (In case of NPMI, in **all** cases), with gains of 500% when compared to hLDA and 549% when compared to HSOC, some of the strongest baselines.
- A qualitative and quantitative statistical analyses of the individual components of our solution.

2 RELATED WORK

Hierarchical Topic Modeling (HTM) can be roughly grouped into supervised and unsupervised methods. Considering the supervised HTM strategies, we here highlight some relevant supervised extensions to the traditional Latent Dirichlet Allocation (LDA) [2], a widely used strategy for topic modeling (TM). Recall that **LDA** assumes a Dirichlet probability distribution over textual data to estimate as probabilities of words for each topic. In [13] the authors propose the **SLDA**, a supervised extension of LDA, for traditional topic modeling, that provides a statistical model for labeled documents that allows connecting each document to a regression variable in order to find latent topics that will best predict the response variables for future unlabeled documents. Based on **SLDA**,

[1] Distances in the embeddings space are global as they do consider the whole vocabulary and interactions among words in specific contexts.

the Hierarchical Supervised LDA (HSLDA) [22] incorporates the hierarchy of multi-label and pre-labeled data into a single model thus providing extended prediction capabilities w.r.t. the latent hierarchical topics. Such a strategy also explores the out-of-sample label in order to enhance prediction performance. The Supervised Nested LDA (SNLDA) [23], also based on **SLDA**, implements a generative probabilistic strategy where topics are sampled from a probability distribution. **SNLDA** extends the SLDA assuming that the topics are organized into a tree structure. Although our focus is on unsupervised solutions, we include **SLDA**, **HSLDA** and **SNLDA** as baselines in our experimental evaluation.

We now turn our attention on the unsupervised HTM strategies, in which a hierarchical structure is learned during topic extraction. In [16] the authors propose the Hierarchical Pachinko Allocation Model (hPAM), an extension of the TM technique known as *Pachinko Allocation* (PAM) [10]. In PAM, documents are a mix of distributions over an individual topic set, using a directed acyclic graph to represent the co-occurrences of topics. Each node in such a graph represents a *Dirichlet* distribution. At the highest level of PAM there is only a single node, where the lowest levels represent a distribution between nodes of the next higher level. In hPAM, each node is associated with a distribution over the vocabulary of documents. In [4], the authors propose the **hLDA** algorithm, which is also an expansion of LDA, being considered state-of-the-art in HTM. In hLDA, in addition to using the text Dirichlet distribution, the nested Chinese Restaurant Process (nCRP) is used to generate a hierarchical tree. NCRP needs two parameters: the tree level and a γ parameter. At each node of the tree, a document can belong to a path or create a new tree path with probability controlled by γ. More recently, in [31] the authors propose the unsupervised HTM strategy named knowledge-based hierarchical topic model (**KHTM**). This method is based on hLDA and, as such, models a generative process whose parameter estimation strategy is based on Gibbs sampling. KHTM is able to uncover prior knowledge (such as the semantic correlation among words) organizing them into a hierarchy, consisting of knowledge sets (k-sets). More specifically, the method first generates, through hLDA, an initial set of topics. After comparing pairs of topics, those topics with similarity higher than α (a.k.a., k-sets) are then filtered so that the first 20 words of each topic are kept and the remaining are just discarded. Those extracted k-sets are then used as an extra weight when extracting the final topics. All the previously discussed methods are used as baselines in our experimentation.

Probably the most similar work to ours is the HSOC strategy, proposed in [11]. As discussed in Section 1, in [11] the authors propose to use NMF for solving HTM tasks. In order to mitigate the main drawbacks of NMF in the HTM setting[2], HSOC relies on three optimization constraints to properly drive the matrix factorization operations when uncovering the hierarchical topic structure. Such constraints are global independence, local independence, and information consistency, and allow HSOC to come up with hierarchical topics that somehow preserve topic coherence and reasonable hierarchical structures. As we shall see in Section 5, HSOC is still not capable of extracting coherent topics when applied to short text data, which is currently prominent of the Web, especially on social network environments. Yet, we use it as one of our baselines.

As we can observe, almost all aforementioned models, supervised or unsupervised, are based on LDA. As discussed in Section 1, though matrix factorization strategies normally present better results than Dirichlet strategies in TM tasks, for HTM the situation is quite different. In fact, matrix factorization methods face difficult challenges in HTM, mainly regarding data size as ones goes deeper into the hierarchy. More specifically, at every hierarchical level, a matrix factorization needs to be applied to increasingly smaller data sets, ultimately leading to insufficient data at lower hierarchy levels. These approaches also do not exploit semantics nor any type of external enrichment, relying only on the statistical information extracted from the dataset. Contrarily, here we propose a new HTM approach, called **CluHTM**, which exploits externally built *word embedding* models to incorporate semantic information into the hierarchical topic tree creation. This brings some important advantages to our proposal in terms of effectiveness, topic coherence and hierarchy reasonableness altogether. Finally, unlike HSOC, the only baseline to explore NMF for HTM, we take a different perspective in our solution, focusing on data engineering aspects of the problem instead of focusing on the optimization process.

3 BACKGROUND

For a better understanding of our proposal, in this section, we briefly describe the CluWords and we present the details regarding the stability measure used to select the number of topics for the HTM task.

3.1 CluWords Representation

We start by describing the strategy that combines the traditional Bag of Words (BoW) representation with semantic information related to the terms present in the documents. The semantic context is obtained employing a "pre-trained" word representation, such as Fasttext [15]. The process of transforming each original words in a cluster of words (a.k.a., CluWord) is composed of two phases: (i) Generation of CluWords and (ii) TFIDF weights, explained below.

3.1.1 Cluwords Generation. Let $\mathcal{V}$ be the vocabulary of terms present in the set of documents $\mathcal{D}$. Also, let $\mathcal{W}$ be the set of vectors representing each term in $\mathcal{V}$, according to the "pre-trained" word embedding representation (e.g., Fasttext). Each term $t \in \mathcal{V}$ has a corresponding vector $u \in \mathcal{W}$, with l dimensions corresponding to the the word vector space. The CluWords representation is defined as a matrix $C \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$, where each index $C_{t,t'}$ is computed according to Eq. 1, as follows:

$$C_{t,t'} = \begin{cases} \omega(t,t') & \text{if } \omega(t,t') \geq \alpha \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where $\omega(t,t')$ is the cosine similarity defined in Eq. 2 and α is a similarity threshold that acts as a regularizer for such a representation, where larger values lead to more sparse representations. In this notation each CluWord is represented as a row C_t and each column $t' \in \mathcal{V}$ may correspond to a component in C_t if the cosine similarity between the vectors for t and t' in the word vector space is greater than or equal to α. Otherwise, the column t' is equal to zero.

[2] Namely, the incoherence of topics and unreasonable hierarchical structure caused by the lack of a learned probability distribution that governs the document/topics relationships

$$\omega(u,v) = \frac{\sum_i^l u_i \cdot v_i}{\sqrt{\sum_i^l u_i^2} \cdot \sqrt{\sum_i^l v_i^2}} \tag{2}$$

The CluWord C_t for term t relates to t' through its closest words, limiting this relationship with the cutoff value α that filters out unrelated words (i.e., words that do not have a significant relationship with t) from the CluWord. Since threshold α is a cosine similarity value, it is contained within the interval $[0, 1]$. If $\alpha = 0$ the similarities of every term in $\mathcal{V}_T$ are included in C_t, otherwise, if $\alpha = 1$ only the similarity of t to itself is included in C_t.

Notice that the term relationships contained in the CluWords are global, since it considers a vector space formed by the whole vocabulary of the dataset and contextual information shared by them in all documents. This is a key aspect of our solution.

3.1.2 TFIDF Weight for CluWords. The idea behind the TFIDF weight is to combine the two aspects of the conventional TFIDF metric [24] with the semantic information of a CluWord. The TFIDF weighting for the CluWords representation is defined in Eq. 3.

$$C_{TF-IDF} = C_{TF} \times idf(C) \tag{3}$$

First, the conventional term frequency (TF) can be represented as a matrix $T \in \mathbb{R}^{|\mathcal{D}| \times |\mathcal{V}|}$, where each position $T_{d,t}$ relates to the frequency of a term t in document d. Thus, the C_{TF} can be measured as a product of matrices, according to Eq. 4. Each value of $C_{TF_{d,t}}$ corresponds to the sum of the products of frequencies $T_{d,t'}$ of each $t' \in C_{t,t'} \neq 0$ occurring in document d.

$$C_{TF} = T \times C \tag{4}$$

To compute the IDF for a CluWord C_t, we first compute the average weights of terms occurring in the vocabulary $\mathcal{V}_{d,C_t}$, as described in Eq. 5. The vocabulary is composed by all the terms in d with weights greater than zero in the corresponding CluWord C_t. More formally, $\mathcal{V}_{d,C_t} = \{t' \in d | C_{t,t'} \neq 0 \ in \ C_t\}$. Finally, we compute the IDF C_t as defined in Eq. 6, where $\mathcal{D}$ is the data set.

$$\mu_{C_t,d} = \frac{1}{\left|\mathcal{V}_{d,C_t}\right|} \cdot \sum_{t \in \mathcal{V}_{d,C_t}} w_t \tag{5}$$

$$idf(C_t) = log\left(\frac{|\mathcal{D}|}{\sum_{1 \le d \le |\mathcal{D}|} \mu_{C_t,d}}\right) \tag{6}$$

3.2 Stability Measure

In this section, we describe the stability measure [3] explored here for automatic selection of the number of latent topics to be extracted by CluHTM. Such a measure is motivated by the term-centring approach generally taken in topic modeling strategies, where precedence is generally given to the term-topic output and topics are summarized using a truncated set of top terms.

The stability measure is illustrated in Algorithm 1. For each value of K in the $[\mathcal{K}_{min}, \mathcal{K}_{max}]$ range, the algorithm proceeds as follows. First, it learns a topic model considering the complete data set representation $\mathcal{T}$ (line 2), which will be used as a reference point for analyzing the stability afforded by the K topics. We represent this reference set as $\mathcal{W}$. Note that each topic is represented by the p top words. Subsequently, $\mathcal{R}$ samples of the data are randomly drawn from $\mathcal{T}$ without replacement, forming a subset of β% documents (line 5). Thus, β denotes the sampling rate. Then, the algorithm generates $\mathcal{R}$ topic models, one for each sub-sampling (line 6). Again, all the extracted topics are represented by the p top words as well.

$$AJ(W_i, W_j) = \frac{1}{t} \sum_{d=1}^{t} \gamma_d(W_i, W_j) \tag{7}$$

where

$$\gamma_d(W_i, W_j) = \frac{W_{i,d} \cap W_{j,d}}{W_{i,d} \cup W_{j,d}} \tag{8}$$

Algorithm 1: Stability

```
Input: $\mathcal{K}_{min}$ - Number of minimum topics;
       $\mathcal{K}_{max}$ - Number of maximum topics;
       $\mathcal{R}$ - Number of runs for compute stability;
       $\mathcal{T}$ - Data representation;
Output: $K$ - Selected number of topics.
1  foreach $K \in [\mathcal{K}_{min}, \mathcal{K}_{max}]$ do
2      $\mathcal{W} \leftarrow NMF(\mathcal{T}, K)$;
3      $scores \leftarrow \emptyset$;
4      foreach $r \in \mathcal{R}$ do
5          $\mathcal{T}_r \leftarrow RandomSample(\mathcal{T})$;
6          $\mathcal{W}_r \leftarrow NMF(\mathcal{T}_r, K)$;
7          $scores \leftarrow scores \cup Agree(\mathcal{W}, \mathcal{W}_r)$
8      $score_K \leftarrow \overline{scores}$;
9  $K \leftarrow K \in max(score_K)$;
10 return $K$
```

To measure the overall stability at K, the algorithm calculates the mean agreement between the reference ranking set and all other ranking sets using Eq. 8. The agreement is measured between two different topic models $W_x = R_{x1}, \cdots, R_{xp}$ and $W_y = R_{y1}, \cdots, R_{yp}$, represented by p ranked words. Thus, the similarity matrix $M_{p \times p}$ is built, where the position $M_{i,j}$ indicates the agreement among R_{xi} and R_{yj}. Such a similarity matrix is computed using the Average Jaccard coefficient (Eq. 7. The goal is to find the best match between the rows and columns of M. The optimal permutation ρ may be found in $\mathbb{O}(p^3)$ time by solving the minimal weight bipartite matching problem using the Hungarian method [8]. Thus, the agreement measure can be formalized as:

$$agree(W_x, W_y) = \frac{1}{p} \sum_{i=1}^{p} AJ(R_{xi}, \rho(R_{xi})) \tag{9}$$

4 PROPOSED SOLUTION

In this section, we present our non-probabilistic hierarchical topic model method, named CluHTM, which is inspired by the methods described in Section 3. CluHTM uses six inputs, as shown in Algorithm ??. $\mathcal{D}_{max}$ corresponds to the depth down to which we want to extract the hierarchical structure. $\mathcal{K}_{min}$ and $\mathcal{K}_{max}$ control the range of topics we want to explore for the stability measure. Such a range will be used in all levels of the hierarchy. $\mathcal{R}$ is the number of random samples to be exploited by the stability measure. $\mathcal{T}$ is the input text data. And, finally, $\mathcal{W}$ is the "pre-trained" word embedding vector space used for the CluWords generation. The output of the method is the hierarchical structure of p top words in each topic.

The method starts by getting the root topic (line 2 of Algorithm **??**), as well as, the input text data. It is an iterative method, which works with a queue schema to build the hierarchical structure. Throughout the iterations, all topics that need to be extracted in the hierarchical structure are added to this queue. Thus, at each iteration (line 3), the algorithm extracts the current depth, the text data to be used by the factorization and stability method, as well as, the parent structure (i.e., parent topics of the respective one). The text data and the "pre-trained" embedding are used to build the CluWords representation (line 5). Then, the stability measure, discussed in the Section 3 (Algorithm 1) is calculated for choosing the number of topics (i.e., parameter K). After choosing K, the topic model method is executed using the CluWords representation. In the Algorithm **??**, we use the NMF [1] method as the topic modeling strategy (line 7). The top p words are selected in line 8 for each extracted topic, and then, for each topic, the new parent configuration is stored in $\mathcal{H}$, as well as in the queue, if it does not exceed the maximum depth (line 12).

Summarizing, our solution exploits *global* semantic information (preserved by CluWords) within *local* factorizations, limited by a stability criterion that defines the 'shape' of the hierarchical structure. Though simple (and original), the combination of these ideas is extremely powerful for solving the HTM task and the main responsible for our remarkable experimental results, as we shall see.

5 EXPERIMENTAL RESULTS

5.1 Experimental Setup

5.1.1 Datasets. The primary goal of our solution is to effectively do hierarchical topic modeling so that more coherent topics can be extracted. To evaluate topic model coherence, we consider 12 real-world datasets as a reference. All of them were obtained from previous works in the literature. For all datasets, we performed stopwords removal (using the standard SMART list) and removed words such as adverbs, using the VADER lexicon dictionary [5], as the vast majority of the important words for identifying topics are nouns and verbs. These procedures improved both the efficiency and effectiveness of all analyzed strategies. Table 1 provides a brief summary of the reference datasets, reporting the number of features (words) and documents, as well as the mean number of words per document (density) and the corresponding references.

Table 1: Dataset characteristics

dataset	#Feat	#Doc	Density
Angrybirds [12]	1,903	1,428	7.135
Dropbox [12]	2,430	1,909	9.501
Evernote [12]	6,307	8,273	11.002
InfoVis-Vast [3]	6,104	909	86.215
Pinterest [12]	2,174	3,168	4.478
TripAdvisor [12]	3,152	2,816	8.532
Tweets [9]	8,029	12,030	4.450
WhatsApp [12]	1,777	2,956	3.103
20NewsGroup [4]	29,842	15,411	76.408
ACM [29]	16,811	22,384	30.428
Uber [28]	5,517	11,541	7.868
Facebook [28]	5,168	12,297	6.427

5.1.2 Evaluation, Algorithms and Procedures. We compare the HTM strategies using representative topic quality metrics in the literature [18, 19]. There are three classes of topic quality metrics based on three criteria: (a) coherence, (b) mutual information and (c) semantic representation. In this paper, we focus on these three criteria since they are the most used metrics in the literature [25]. We consider three topic lengths (5, 10 and 20 words) for each metric in our evaluation, since different lengths may bring different challenges.

Regarding the metrics, *coherence* captures easiness of interpretation by co-occurrence. Words that co-occur frequently in similar contexts in a corpus are easier to correlate since they usually define a more well-defined "concept" or "topic". We employ an improved version of regular coherence [18], called Coherence, defined as

$$c(t, W_t) = \sum_{w_1, w_2 \in W_t} log \frac{d(w_1, w_2) + \varepsilon}{d(w_1)}, \tag{10}$$

where $d(w1)$ denotes the number of occurrences of $w1$, $d(w1, w2)$ is the number of documents that contain both $w1$ and $w2$ together, and ε is a smoothing factor used for preventing $log(0)$.

Another class of topic quality metrics is based on the notion of *pairwise pointwise mutual information (PMI)* between the top words in a topic. It captures how much one "gains" in the information given the occurrence of the other word, taking dependencies between words into consideration. Following a recent work [18], we here compute a *normalized version of PMI* (NPMI) where, for a given ordered set of top words $W_t = (w_1, ..., w_N)$ in a topic:

$$NPMI_t = \sum_{i<j} \frac{log \frac{p(w_i, w_j)}{p(w_i)p(w_j)}}{-log\ p(w_i, w_j)}. \tag{11}$$

Finally, the third class of metrics is based on the distributed word representations introduced in [18]. The intuition is that, in a well-defined topic, the words should be semantically similar, or at least related, to be easily interpreted by humans. Hence, in a d-dimensional vector space model in which every vocabulary word $w \in W$ has been assigned to a vector $v_w \in R^d$, the vectors corresponding to the top words in a topic should be close to each other. In [18], the authors define topic quality as the average distance between the top words in the topic, as follows:

$$W2V - L1 = \frac{1}{|W_t|(|W_t| - 1)} \sum_{w_1 \neq w_2 \in W_t} d_{cos}(v_{w_1}, v_{w_2}). \tag{12}$$

Generally speaking, let $d(w_1, w_2)$ be a distance function in R^d. In this case, larger $d(w_1, w_2)$ corresponds to worse topics (with words not as localized as in topics with smaller average distances). In [18], the authors suggest four different distance metrics, with cosine distance achieving the best results. We here also employ the cosine distance, defined as $d_{cos}(x, y) = 1 - x^T y$.

Next, we compare our proposed data representation described in Section 4, as well as the best configuration with seven hierarchical topic model strategies recently proposed, marked in bold in Section 2. For the input parameters of CluHTM (Algorithm **??**), we set $\mathcal{K}_{min} = 5$, $\mathcal{K}_{max}$=25, $\mathcal{R} = 10$ and $\mathcal{D}_{max} = 3$. For the baseline methods, we adopt the parameters suggested by their respective works. We assess the statistical significance of our results by means

[3] https://www.cc.gatech.edu/gvu/ii/jigsaw/datafiles.html

[4] http://qwone.com/~jason/20Newsgroups/

of a paired t-test with 95% confidence and Holm-Bonferroni correction to account for multiple tests. In the next section, we present the results of experiments conducted to evaluate the effectiveness of the CluHTM using three metrics of evaluation, followed by a quantitative analysis to explain our gains.

Figure 1: Comparing the results achieved by each uHTM strategy for Coherence Scores.

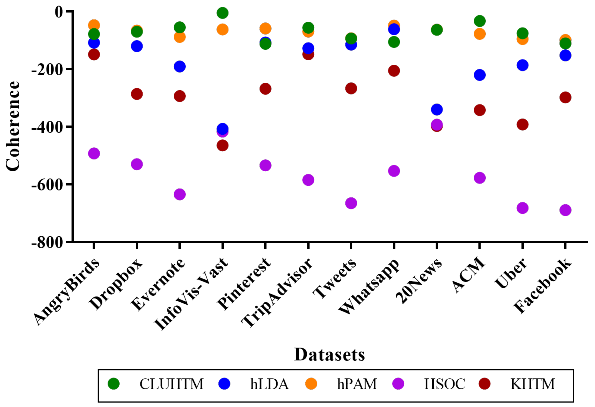

Figure 2: Comparing the results achieved by each uHTM strategy for W2V-L1.

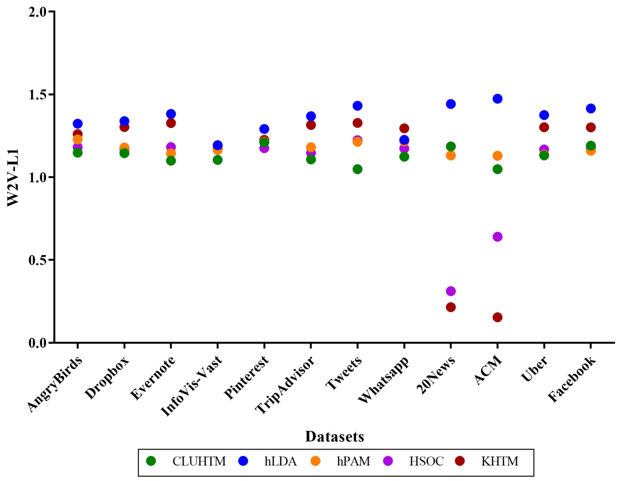

Table 2: Comparing the results achieved by each supervised HTM strategy for Coherence, W2V-L1 and NPMI.

Dataset	CluHTM	SLDA	SNLDA	HSLDA
		Coherence		
20News	−62.6898 ± 21.0606 ▲	−403.3413 ± 90.2313	−410.0020 ± 71.2366	−309.9041 ± 132.5511
ACM	−32.3371 ± 29.5853 ▲	−539.6660 ± 115.2125	−507.4476 ± 108.6966	−486.4835 ± 104.9369
		W2V-L1		
20News	1.1863 ± 0.1176 ▼	0.3093 ± 0.2006	0.3456 ± 0.2051	0.0952 ± 0.1094
ACM	1.0489 ± 0.6506 ●	0.6347 ± 0.2617	0.6803 ± 0.2243	0.2816 ± 0.1567
		NPMI		
20News	0.9351 ± 0.0365 ▲	0.2714 ± 0.1157	0.2205 ± 0.0752	0.4383 ± 0.2162
ACM	0.9641 ± 0.0416 ▲	0.2071 ± 0.0579	0.2064 ± 0.0529	0.2761 ± 0.0978

5.2 Experimental Results

We start by comparing CluHTM against four state-of-the-art uHTM baselines considering the twelve reference datasets. Three hierarchical levels for each strategy are used in this comparison. In Figures 1, 2 and 3 we contrast the results of our proposed CluHTM and the reference strategies, considering Coherence, W2V-L1, NPMI metrics. Note that, each strategy extracted different number of topics in its respective hierarchical structure. Considering the Coherence scores (Figure 1), our strategy achieves the single best results in 2 out of 12 datasets, with gains up to 58% and 92% against the strongest baseline (hPAM), tying in 8 out 12 and loosing 2 times for hLDA and hPAM. Similar results can be observed for W2V-L1 metric – CluHTM ties 10 out 12 results, 1 win and 1 loss for KHTM strategy. In case of NPMI, as seen in Figure 3, our strategy excels – we outperformed all baselines, with gains over 500% against the strongest ones. As we will see in the next results, even with few losses, our method proves to be more consistent than the baselines.

We turn now our attention to the effectiveness of our proposal when compared to the supervised HTM strategies. We consider the 20News and ACM datasets for which have a ground truth for supervised strategies. Table 2 presents the results considering Coherence, W2V-L1 and NPMI. The statistical significance tests ensure that the best results, marked in ▲, are superior to others. The statistically equivalent results are marked in ● while statistically losses are marked in ▼. Once again, in Table 2, our proposed strategy achieves the best results in 4 out 6 cases, tying with SNLDA and HSLDA in ACM and loosing for SLDA in 20News, both considering the W2V-L1 metric. However, it is important to notice that our method does not use privileged class information to build the hierarchical structure nor to extract topics. CluHTM shows consistent results in 2 out 3 effectiveness metrics.

We provide a comparative table with all experimental results[5], including the results for each extracted level of the hierarchical structure. We summarize our findings regarding the behavior of all analyzed strategies in the 12 datasets in Table 3. It counts the number of times each strategy figured out as a top performer[6]. As we can see, our proposal is in large advantage over the other explored baselines, being the strategy of choice in the vast majority of cases. Overall, considering a universe of 36 experimental results (combination of 3 evaluation metrics over 12 datasets) we obtained the best results (33 best performances), with the strongest baseline – hPAM – coming far away, with just 17 top performances. Another interesting observation is that, in terms of NPMI, CluHTM wins in **all** cases.

5.3 Impact of the Factors

One important open question remains to be answered: what impact do HTM strategies and types of text data have on the effectiveness of building cohesive topics? In order to answer this question, we provide a quantitative analysis regarding the hierarchical topic modeling effectiveness, measured by the NPMI score.

We start our analysis by quantifying the effects of various factors (parameters of interest) that might affect the performance of the

[5] https://drive.google.com/file/d/1k78Hl72_cyM6k2ty0VVJsMht9G3WPgmQ/view
[6] If two strategies are statistically tied as top performers in the same dataset, both will be counted.

Figure 3: Comparing the results achieved by each uHTM strategy for NPMI.

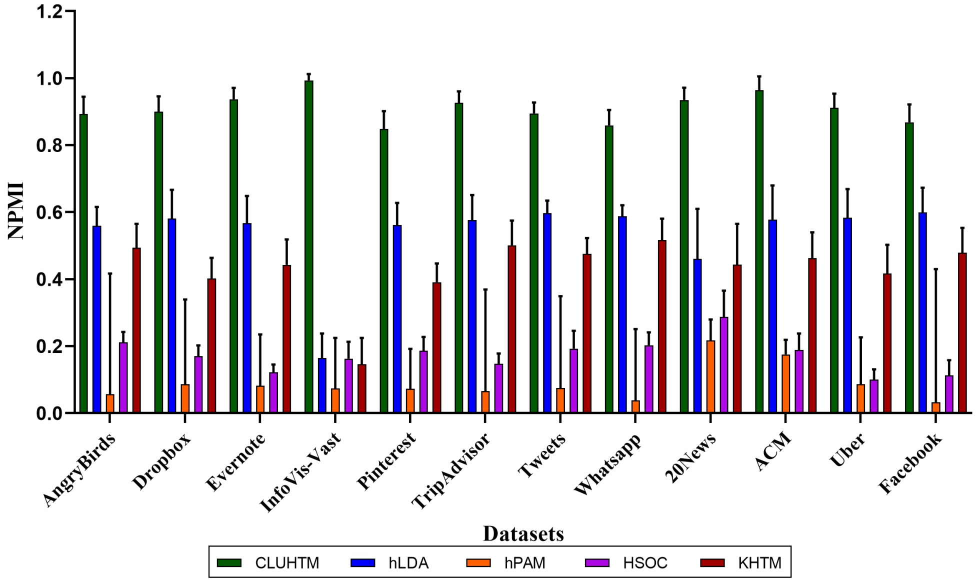

Table 3: Number of times each strategy was the top performer. CluHTM is the choice in the vast majority of cases.

Method	Metric			∑
	NPMI	W2V-L1	Coherence	(Sum)
CluHTM	**12**	**11**	**10**	**33**
hLDA	0	2	9	11
hPAM	0	9	8	17
HSOC	0	9	0	9
KHTM	0	6	2	8
SLDA	0	1	0	1
SNLDA	0	2	0	2
HSLDA	0	2	0	2

system under study, while also determining whether the observed variations are due to significant effects or simply due to random variations (e.g., measurement errors, inherent variability of the process being analyzed [6]). To this end, we adopt a 2^k factorial design, since it allows us to quantify the effect of k factors, under two possible levels (lower and upper), on the performance of the system under study (measured by a given response variable). A 2^k design consists of 2^k experiments defined by all combinations of factor levels. Such design is appropriate to provide some initial insights into the relative importance of different factors on the response variable. This importance is expressed as the fraction of the total variation observed in the measurements that can be explained by the changes in the levels of each factor: the more important factors explain a larger fraction of the total variation. The fraction of variation that cannot be explained is credited to the experimental errors.

We here apply a 2^k design to analyze the impact of the main characteristics of our proposed strategy. The ultimate goal is to better understand the main factors that contribute to the generation of more cohesive topics. As previously mentioned, the response variable in our design is the average NPMI score. We evaluate the two most impacting factors ($k = 2$) affecting the results of the proposed approach. The first one is the HTM strategy exploited to extract the topics and the hierarchical structure as well. The second factor is the intrinsic properties of text data, including noise and ambiguity.

Results are shown in Table 4. In the Table 4, we highlight the average NPMI score and the effects of each algorithm and datasets on that average score. From the effects, we can see that the CluHTM impact in the NPMI value is 99.38% higher than the overall average. We also see that the hLDA has around 18.67% higher NPMI score than the average overall score and the NMF based baseline HSOC has an NPMI score approximately 64.44% *smaller* than the overall NMPI average. Looking at the effects corresponding to the datasets, the factorial design experiment tells us that they have a small variation concerning the obtained average NPMI scores. We see that the dataset with the most variation in relation to the average NPMI score is InfoVis-Vast, with score of about 29.97% smaller than the overall NPMI.

We perform the ANOVA test to statistically assess whether the studied factors are indeed significant. As detailed in Table 5, we can conclude, with 99% confidence, according to the F-test, that the choice of algorithm (factor A) explains approximately 90% of the obtained NPMI values. Furthermore, we can also conclude that text data (factor B), as well as the experimental errors, have small influence over the experimental results.

Summarizing, the major factor explaining the results is the choice of the algorithm, with CluHTM as a clear winner, and these results are consistent across all tested datasets.

6 CONCLUSION

We advance the state-of-the-art in hierarchical topic modeling by means of the design, implementation and evaluation of a novel unsupervised non-probabilistic HTM method – CluHTM. Our new method exploits a richer (global) semantic data representation based

Table 4: Overview of the factorial desgin

Dataset-Algorithm	CluHTM	hLDA	hPAM	HSOC	KHTM	Row Sum	Row Mean	Row Effect
Angry Birds	0.8934	0.5593	0.3604	0.2120	0.4940	2.5191	0.5038	0.0507
Dropbox	0.9002	0.5806	0.2529	0.1703	0.4022	2.3062	0.4612	0.0082
Evernote	0.9374	0.5668	0.1534	0.1222	0.4426	2.2224	0.4445	-0.0086
Facebook	0.8686	0.5998	0.1517	0.1128	0.4791	2.2120	0.4424	-0.0107
InfoVis-Vast	0.9935	0.1650	0.1191	0.1632	0.1459	1.5867	0.3173	-0.1357
Pinterest	0.8482	0.5614	0.3028	0.1865	0.3912	2.2901	0.4580	0.0049
Trip Advisor	0.9265	0.5769	0.2745	0.1477	0.5007	2.4263	0.4853	0.0322
Tweets	0.8950	0.5966	0.2130	0.1928	0.4759	2.3733	0.4747	0.0216
Uber	0.9116	0.5829	0.1403	0.1006	0.4168	2.1522	0.4304	-0.0226
Whatsapp	0.8594	0.5881	0.3976	0.2031	0.5172	2.5654	0.5131	0.0600
Col Sum	9.0338	5.3774	2.3657	1.6112	4.2656	22.6537	-	-
Col Mean	0.9034	0.5377	0.2366	0.1611	0.4266	-	0.4531	-
Col effect	0.4503	0.0847	-0.2165	-0.2920	-0.0265	-	-	-

Table 5: ANOVA Test with 99% confidence to measure the impact of each factor.

Component	Sum of Squares	% Variation	Degrees of Freedom	Mean Square	F-Computed	F-Table (0.99)
y	14.0670	-	50	-	-	-
$y_{..}$	10.2638	-	1	-	-	-
$y - y_{..}$	3.8032	100.00%	49	-	-	-
A	3.4276	90.12%	4	0.8569	127.9197	3.8903
B	0.1345	3.92%	9	0.0149	2.2303	2.9461
e	0.2412	6.34%	36	0.0067	-	-

on word embeddings – CluWords – when capturing the latent hierarchical structure of topics as well as an original application of a stability measure to define the "shape" of the hierarchy. Both solutions are unheard in the HTM literature. Such strategy enables the the application of non-negative matrix factorization for uncovering latent topics, allowing us to overcome some important drawbacks faced by probabilistic HTM models such as learning inefficiencies, proneness to overfitting and noneffective handling of short texts. It also allows us to guarantee the desired (even necessary) coherence and reasonableness properties enjoyed by probabilistic models in the HTM context.

We assess the effectiveness of our proposal, both quantitatively and qualitatively, considering 12 real-world datasets and 7 baselines, including standard and popular as well as recently proposed methods, covering probabilistic and non-probabilistic, supervised and non-supervised strategies. Overall our experimental results, establishes CluHTM as the current state-of-the-art for HTM tasks, outperforming all explored baselines in the vast majority of cases – it is the top performer in all cases considering the NPMI metric; in 11 out of 12 cases considering the W2V-L1 metric; and in and 10 out of 12 cases considering the coherence metric, a remarkable result. A factorial experiment also shows that CluHTM results are consistent across most datasets, independently of the data characteristics and idiosyncrasies.

As future work, we plan to exploit the rich CluWord representations in probabilistic models for HTM as well as in the own stability metric, which currently does not exploit this information. Finally, we will also apply CluHTM in other representative applications on the Web such as hierarchical classification, by devising a supervised version of CluHTM. The idea is that, if available, we could take advantage of some supervision (labeling) in our inner mechanisms and representations.

REFERENCES

[1] Michael W Berry, Murray Browne, Amy N Langville, V Paul Pauca, and Robert J Plemmons. 2007. Algorithms and applications for approximate nonnegative matrix factorization. *Computational statistics & data analysis* (2007).

[2] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[3] Derek Greene, Derek O'Callaghan, and Pádraig Cunningham. 2014. How Many Topics? Stability Analysis for Topic Models. *CoRR* (2014).

[4] Thomas L Griffiths, Michael I Jordan, Joshua B Tenenbaum, and David M Blei. 2004. Hierarchical topic models and the nested Chinese restaurant process. In *Advances in neural information processing systems*. 17–24.

[5] Clayton J. Hutto and Eric Gilbert. 2014. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. In *ICWSM'14*.

[6] Raj Jain. 1991. *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*.

[7] Suin Kim, Jianwen Zhang, Zheng Chen, Alice Oh, and Shixia Liu. 2013. A hierarchical aspect-sentiment model for online reviews. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*.

[8] Harold W. Kuhn. 2010. The Hungarian Method for the Assignment Problem. In *50 Years of Integer Programming*.

[9] Quanzhi Li, Sameena Shah, Xiaomo Liu, Armineh Nourbakhsh, and Rui Fang. [n. d.]. TweetSift: Tweet Topic Classification Based on Entity Knowledge Base and Topic Enhanced Word Embedding. In *CIKM'16*.

[10] Wei Li and Andrew McCallum. 2006. Pachinko allocation: DAG-structured mixture models of topic correlations. In *Proceedings of the 23rd international conference on Machine learning*. ACM, 577–584.

[11] Rui Liu, Xingguang Wang, Deqing Wang, Yuan Zuo, He Zhang, and Xianzhu Zheng. 2018. Topic splitting: a hierarchical topic model based on non-negative matrix factorization. *Journal of Systems Science and Systems Engineering* 27, 4 (2018), 479–496.

[12] Washington Luiz, Felipe Viegas, Rafael Alencar, Fernando Mourão, Thiago Salles, Dárlinton Carvalho, Marcos Andre Gonçalves, and Leonardo Rocha. 2018. A Feature-Oriented Sentiment Rating for Mobile App Reviews *(WWW '18)*.

[13] Jon D Mcauliffe and David M Blei. 2008. Supervised topic models. In *Advances in neural information processing systems*. 121–128.

[14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. (2013).

[15] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in Pre-Training Distributed Word Representations. In *LREC'18*.

[16] David Mimno, Wei Li, and Andrew McCallum. 2007. Mixtures of hierarchical topics with pachinko allocation. In *Proceedings of the 24th international conference on Machine learning*. ACM, 633–640.

[17] Zhao-Yan Ming, Kai Wang, and Tat-Seng Chua. 2010. Prototype hierarchy based clustering for the categorization and navigation of web collections. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2–9.

[18] Sergey I Nikolenko. 2016. Topic Quality Metrics Based on Distributed Word Representations. In *SIGIR'16*.

[19] Sergey I Nikolenko, Sergei Koltcov, and Olessia Koltsova. 2017. Topic modelling for qualitative studies. *Journal of Information Science* (2017).

[20] John Paisley, Chong Wang, David M Blei, and Michael I Jordan. 2014. Nested hierarchical Dirichlet processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37, 2 (2014), 256–270.

[21] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation.. In *EMNLP*.

[22] Adler J Perotte, Frank Wood, Noemie Elhadad, and Nicholas Bartlett. 2011. Hierarchically supervised latent Dirichlet allocation. In *Advances in neural information processing systems*. 2609–2617.

[23] Philip Resnik, William Armstrong, Leonardo Claudino, Thang Nguyen, Viet-An Nguyen, and Jordan Boyd-Graber. 2015. Beyond LDA: exploring supervised topic modeling for depression-related language in Twitter. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*. 99–107.

[24] Gerard Salton and Christopher Buckley. 1988. Term-weighting Approaches in Automatic Text Retrieval. *Inf. Process. Manage.* 24, 5 (1988), 513–523.

[25] Tian Shi, Kyeongpil Kang, Jaegul Choo, and Chandan K. Reddy. 2018. Short-Text Topic Modeling via Non-negative Matrix Factorization Enriched with Local Word-Context Correlations. In *WWW '18*. 1105–1114.

[26] Jian Tang, Zhaoshi Meng, XuanLong Nguyen, Qiaozhu Mei, and Ming Zhang. 2014. Understanding the Limiting Factors of Topic Modeling via Posterior Contraction Analysis. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32 (ICML'14)*. JMLR.org, I–190–I–198. http://dl.acm.org/citation.cfm?id=3044805.3044828

[27] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. Hierarchical Dirichlet Processes. *J. Amer. Statist. Assoc.* 101, 476 (2006), 1566–1581. https://doi.org/10.1198/016214506000000302

[28] Felipe Viegas, Sérgio Canuto, Christian Gomes, Washington Luiz, Thierson Rosa, Sabir Ribas, Leonardo Rocha, and Marcos André Gonçalves. 2019. CluWords: Exploiting Semantic Word Clustering Representation for Enhanced Topic Modeling. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19)*. ACM, New York, NY, USA, 753–761. https://doi.org/10.1145/3289600.3291032

[29] Felipe Viegas, Marcos Gonçalves, Wellington Martins, and Leonardo Rocha. 2015. Parallel Lazy Semi-Naive Bayes Strategies for Effective and Efficient Document Classification. In *CIKM*.

[30] Chi Wang, Jialu Liu, Nihit Desai, Marina Danilevsky, and Jiawei Han. 2015. Constructing topical hierarchies in heterogeneous information networks. *Knowledge and Information Systems* 44, 3 (2015), 529–558.

[31] Yueshen Xu, Jianwei Yin, Jianbin Huang, and Yuyu Yin. 2018. Hierarchical topic modeling with automatic knowledge mining. *Expert Systems with Applications* 103 (2018), 106–117.