Topic Discovery via Convex Polytopic Model: A Case Study with Small Corpora

King Keung Wu[1], *Member IEEE,* Helen Meng[1], *Fellow IEEE,* and Yeung Yam[2], *Senior Member IEEE,*

***Abstract*—Topic discovery is an important problem in text processing. Topic modeling approaches such as latent Dirichlet allocation (LDA) has been applied quite successfully in extracting topics. However, there still exists several directions for further improvement. Short texts (e.g. tweets and news titles) present the problem of data sparsity for LDA. Second, there needs to be greater transparency in the process of topic discovery in order to enhance interpretability for humans. Third, the robustness of the model needs to be further enhanced to avoid sensitivity to the choice of hyper-parameters. In this paper, we propose a novel geometric approach based on convex polytopic model (CPM) which can discover representative and interpretable topical features from the given corpus. By embedding all documents into a low-dimensional affine subspace, we show that the *topics* can be obtained geometrically as the vertices of a compact polytope which encloses all the embedded documents. We further interpret the features acquired as *topics* and use them to obtain a convex polytopic document representation for every document. We studied the properties of CPM by two small corpora of short texts. Results reveal that the proposed CPM can discover interpretable topics even for short texts. We also discover that the geometric nature of CPM enhances model transparency and topic interpretability, as well as robustness to hyper-parameter selection.**

***Index Terms*—Topic discovery, document categorization, text representation, convex polytope.**

I. Introduction

DISCOVERING topics from a collection of documents is a challenging task in information extraction. The goal of this task is to automatically discover the most representative semantic features as *topics* which capture the gist of the documents such that they are interpretable for humans.

Various topic discovery algorithms have been proposed in recent decades. For example, Latent Semantic Analysis (LSA) extracts the concepts as semantic features of the documents by projecting them onto a low-dimensional latent semantic space using singular value decomposition (SVD) of the term-document matrix [1]. Probabilistic topic models such as Latent Dirichlet Allocation (LDA) [2] are generative models which extract topics by assuming that each document is generated by a mixture of topics where the topics are represented by the distribution of words.

However, the existing approaches have several limitations. For example, the dimensions of the latent space in LSA may be difficult to interpret, as they represent statistical features rather than human-interpretable features. The performance of LDA is easily affected by data sparsity in short documents [3, 4]. Besides, the model is not transparent as the inference process does not explicitly show how the topics are discovered. Furthermore, it is sensitive to the choices of hyper-parameters and the algorithm may give different results in several runs, which implies a need to enhance robustness.

In this paper, we propose a novel topic discovery approach based on convex polytopic model (CPM) which can break through the limitations of the existing methods. CPM transforms the task into a geometric problem by embedding the documents in the corpus into a low-dimensional affine subspace, and then generating a compact convex polytope to enclose the embedded documents. We define the vertices of the polytope as *topics* of the corpus and introduce two new interpretations of the *topics* based on the geometric properties of the vertices. First, a *topic* can be represented by the coordinates of the vertices where every dimension represents the relative strength of the associating word within the topic. If the vertex corresponds to one of the embedded documents, the *topic* can further be defined by the document text which can be directly understood by humans. We refer such document as the *constituent document*. This interpretation can serve as a supplement to the first one and thus improve topic interpretability. In addition, CPM allows every document in the corpus to be represented by the convex combination of these *topics*. We begin our investigation by studying CPM with a small corpus, which enables us to easily visualize the process of topic discovery, providing a transparent topic model. We also show that CPM outperforms the existing methods such as LSA and LDA in terms of model robustness and transparency, as well as topic interpretability.

The contribution of this paper is three-fold: first, we propose a novel topic discovery approach that is applicable in short texts and small corpora where topic models such as LDA tends to fall short. Second, we provide a new perspective to topic interpretability in addition to the conventional vector representation by interpreting the text of the *constituent document* as the representative of a *topic*. Third, we develop a new geometric analysis method for topic models using small corpora which illustrates how the model transparency can be improved to allow users to understand the topic discovery process via visualization.

The rest of the paper is organized as follows. Section II gives the detailed formulation of the proposed CPM. Section III demonstrates the efficacy of CPM by testing on corpora with and without clear categorization. Section IV concludes the paper and discusses some possible future work.

[1] King Keung Wu and Helen Meng are with the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong, China `kkwu@mae.cuhk.edu.hk`, `hmmeng@se.cuhk.edu.hk`

[2] Yeung Yam is with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong, China `yyam@mae.cuhk.edu.hk`

II. Convex Polytopic Model

This section presents the detailed formulation of the proposed CPM and discusses how to interpret the semantic meaning from its geometric formulation.

A. Geometric Formulation

The CPM consists of two steps. First, all documents are embedded in a low-dimensional affine subspace using principal component analysis (PCA) and each document is represented by a point. Second, a compact convex polytope is generated that encloses all the embedded document as points.

Step 1: Embedding documents into low-dimensional affine subspace

Given a corpus of N documents, preprocess it by removing the stopwords and build a vocabulary. Let the vocabulary size be M. The corpus can then be transformed into a sum-normalized term-document matrix $\mathbf{X} \in \mathbb{R}^{M \times N}$ where the entry $\mathbf{x}_{ij}$ is the i-th term count divided by the total counts of all terms in document the j-th document. Let $\bar{\mathbf{X}} \in \mathbb{R}^{M \times N}$ be the matrix with each column equal the average of all the columns of $\mathbf{X}$.

Then, we aim to find the closest low-dimensional subspace for the documents represented by $(\mathbf{X}-\bar{\mathbf{X}})$, which can be done using SVD. Assume that the dimension of the subspace is R, where an orthonormal basis $\mathbf{U} \in \mathbb{R}^{M \times R}$ can be obtained by keeping the first R eigenvectors of $(\mathbf{X}-\bar{\mathbf{X}})(\mathbf{X}-\bar{\mathbf{X}})^{\mathrm{T}}$. The documents can then be projected onto the R-dimensional *document subspace* spanned by the basis $\mathbf{U}$. Let the projected documents be the *document point set* $\{\mathbf{p}_i\}_{i=1}^N$, as the columns of $\mathbf{P} \in \mathbb{R}^{M \times N}$ represented by:

$$\mathbf{P} = \bar{\mathbf{X}} + \mathbf{U}\mathbf{U}^{\mathrm{T}}(\mathbf{X}-\bar{\mathbf{X}}). \tag{1}$$

Note that the columns of $\mathbf{P}$ are still *sum-to-one* and hence the *document points* are on the *sum-to-one hyperplane*.

The above procedure is equivalent to PCA, where the original document points in $\mathbf{X}$ are fitted into an R-dimensional ellipsoid, where the axes of the ellipsoid are the principal components. Note that the principal component corresponds to the direction where the data has the largest variance. Hence, the projected points keep the most distinguishable features. Although LSA also utilizes SVD for finding the latent semantic space, it is fundamentally different from the proposed approach. In traditional LSA, the term-document matrix is not sum-normalized nor subtracted by the average, and hence it is not finding the *affine document subspace*. The key difference between the *latent semantic space* and the *affine document subspace* is that the former measures the difference of documents by the included angle of their document vectors, while the latter measures with their spatial distance. We will show that such property of affine document subspace is useful for finding meaningful features for the corpus of documents.

Step 2: Enclose the embedded document points by a compact polytope

The second step of CPM is to extract meaningful features as topics. As the distance of points in document subspace measures the difference between the documents, we assume that the extremes of the document point set have the least similarity and hence can represent topical features for the corpus. Finding the extreme points is equivalent to finding the convex polytope with the minimum volume that encloses all the points, where the vertices are the extreme points. We define the vertices as the representative *topics* for the corpus. Note that all document points can thus be represented by the convex combinations of the vertices, and the minimum volume enclosing convex polytope always exists and is unique under the permutation of vertex labels for a finite point set. We refer this type of polytope as the normal (NO)-type, where all the vertices are within the point sets [5]. The documents associated with the vertices are thus referred as *constituent documents*. Hence, the features represented by the vertices are interpretable as human can understand the text of the consitituent documents directly. This is different from the existing topic discovery approaches where the topics are represented either by a vector or a distribution of words. The NO-type polytope defines the topics by documents directly.

There are several algorithms that can generate NO-type polytope, under the class of convex hull algorithms. For example, Quickhull [6] is a popular divide-and-conquer-based convex hull algorithm. For more details about the algorithm, please refer to [6].

For the NO-type polytope, the number of vertices may be as large as the number of documents N. However, in some circumstances, we prefer to extract a fixed number of features as topics, denoted by K, which is less than N. The problem can thus be formulated as finding the K-vertex enclosing convex polytope with the minimum volume. Assuming that the dimension of the document subspace is $R = K-1$, it is equivalent to finding the minimum volume simplex (MVS) that contains the document point set. We refer this as the MVS-type polytope. Note that the vertices may no longer be in the document point set, hence a new method is needed to interpret the vertices and will be discussed in the coming subsection.

We define a special type of MVS-type polytope called close-to-normal (CNO)-type, where at least one of the vertices are in the document point set. Those vertices can then be interpreted just like in the NO-type.

B. Convex Polytopic Representation

The *convex polytopic representation* of the *document point* $\mathbf{p}_i$ is defined as the convex combinations of $\mathbf{v}_j$:

$$\mathbf{p}_i = \sum_{j=1}^{K} w_{ij}\mathbf{v}_j, \tag{2}$$

where $w_{ij} \geq 0$ and $\sum_{j=1}^{K} w_{ij} = 1$, for $i = 1, \ldots, N$. The constraint non-negative and sum-to-one is the convexity condition. Equation (2) represents that all points $\{\mathbf{p}_i\}_{i=1}^N$ are enclosed in a convex polytope with vertices $\{\mathbf{v}_j \in \mathbb{R}^M\}_{j=1}^K$. (2) can be written in matrix form:

$$\mathbf{P} = \mathbf{V}\mathbf{A}, \tag{3}$$

where $\{\mathbf{v}_j\}_{j=1}^K$ form the columns of $\mathbf{V} \in \mathbb{R}^{M \times K}$ and $\mathbf{A} \in \mathbb{R}^{K \times N}$ with entries $a_{ij} = w_{ji}$. Note that all columns of $\mathbf{A}$, denoted by $\mathbf{a}_1, \ldots, \mathbf{a}_N$, are sum-to-one and non-negative.

C. Minimum Volume Simplex Analysis

Consider $R = K - 1$ and the matrix $\mathbf{P}$ is known, we apply the algorithm originally proposed by Li et al. in [7, 8] to find the MVS. To formulate the MVS problem into an optimization problem, we first rewrite (3) into the following form:

$$\mathbf{P} = \mathbf{VA} \quad s.t. \quad \mathbf{A} \succeq 0 \quad and \quad \mathbf{1}_K^{\mathrm{T}}\mathbf{A} = \mathbf{1}_N^{\mathrm{T}}. \tag{4}$$

It is known that the volume of simplex defined by the origin and the columns of $\tilde{\mathbf{V}}$ is $|\det(\tilde{\mathbf{V}})|$, where $\det(\tilde{\mathbf{V}})$ is the determinant of $\tilde{\mathbf{V}}$. As the distance from the origin to the *sum-to-one hyperplane* is fixed, minimizing the volume of simplex that contains all *document points* is equivalent to minimizing the volume of simplex defined by the origin and the columns of $\tilde{\mathbf{V}}$, the minimal volume simplex problem can be formulated as follows:

$$\begin{aligned} \tilde{\mathbf{V}}^* &= \arg\min_{\tilde{\mathbf{V}}} |\det(\tilde{\mathbf{V}})| \\ s.t. \quad \tilde{\mathbf{V}}^{-1}\tilde{\mathbf{P}} &\succeq 0 \quad and \quad \mathbf{1}_K^{\mathrm{T}}\tilde{\mathbf{V}}^{-1}\tilde{\mathbf{P}} = \mathbf{1}_N^{\mathrm{T}}, \end{aligned} \tag{5}$$

where $\tilde{\mathbf{P}}$ and $\tilde{\mathbf{V}}$ are the projected version of $\mathbf{P}$ and $\mathbf{V}$, respectively.

Li et al. [7] proposed a sequential quadratic programming (SQP) algorithm to solve (5). Then, the vertices $\mathbf{V}$ can be obtained. As each vertex is also on the *sum-to-one hyperplane*, its entry value can be interpreted as the strength of the corresponding term in the topic. Hence, this gives an alternative interpretation of the vertices when they are not within the document point set.

III. Experiments

In this section, we investigate the properties of the proposed CPM using two different corpora. Both corpora are chosen to be small so that it is possible to easily illustrate how the model works by direct examination, as we analyze the characteristics of every step in the method.

We implemented the algorithms of LSA and the proposed CPM in Matlab. The *Qhull* algorithm [6] is applied to generate the NO-type polytope. The *TP Tool* [9] is applied for generating the CNO-type polytope. The algorithm for finding the MVS-type polytope is based on the minimum volume simplex analysis proposed by Li et al. [7, 8]. The results of LDA is computed by a Python toolbox named Gensim [10].

A. Experiment 1 – Small Corpus with Clear Categorization

The first experiment uses a corpus that contains 10 short documents which are the titles of papers categorized by two research fields: *human-computer interface* and *graphs*, respectively (see Table I). We have borrowed the examples from [1] and further extended them slightly, by adding a document (D10 in Table I) about *graphs*. Note that the two categories are expected to have almost no intersection, i.e., the documents corresponding to different categories should have no common keywords. We aim to discover the topical features that characterized the two categories from the unlabeled corpus using the CPM and show that the features extracted by CPM is more interpretable than that of LSA.

Table I: The first corpus used in Experiment 1. Words occurring in more than one document are italicized. D4 and D7 are in bold as they are automatically chosen by the NO-type polytope to be the endpoints, which signify key *constituent documents*.

IDs	Documents
D1	*Human machine interface* for Lab ABC *computer* applications
D2	A *survey* of *user* opinion of *computer system response time*
D3	The *EPS user interface* management *system*
D4	***System* and *human system* engineering testing of *EPS***
D5	Relation of *user*-perceived *response time* to error measurement
D6	The *generation* of *random*, binary, unordered *trees*
D7	**The intersection *graph* of paths in *trees***
D8	*Graph minors* IV: Widths of *trees* and well-quasi-ordering
D9	*Graph minors*: A *survey*
D10	Fast *random graph generation*

Table I shows the 10 documents in the corpus of Experiment 1. D1-D5 belong to *humans-computer interface*, and D6-D10 belong to the *graphs*.

We generated the term-document matrix as in [1], where the stopwords such as *the*, *and* are removed, and only the terms occurring in more than one document are considered. Then, we sum-normalized each column of the term-document matrix representing the input of the documents and applied PCA to project the columns to a 1-D subspace as the number of expected topics is 2.

Figure 1 shows the NO-type convex polytope, which is a line segment. Note that for the 1-D subspace, the NO-type polytope always exists and has the minimal volume, i.e., is also of MVS-type. V1 and V2 are defined here as the endpoints of the line. As shown in Figure 1, the points D4 and D7 coincide with V1 and V2, respectively. Hence, the two documents, D4 and D7, are the *constituent documents* which can serve to illustrate the topic features to enhance interpretability. By observing the text of D4, the words *system* and *human* are the most common words in among D1-D5, and D4 has the largest proportion of these two words. It shows that the NO-type polytope tends to select the document with the largest proportion of the most common words as the vertex. Similar phenomenon is observed for D7, where the words *graphs* and *trees* are the most common words among D6-D10 and D7 has the largest proportion of these words.

Another way to interpret the topics is to use the coordinates of the vertices with respect to the original space, represented by the set of vectors $\{\mathbf{v}_j\}_{j=1}^K$ in Equation (2). The i-th entry of the vector $\mathbf{v}_j$ corresponds to the relative strength of the i-th word in the vocabulary to the j-th topic (represented by the j-vertex). Table II shows the values of $\mathbf{v}_1$ and $\mathbf{v}_2$ in descending order, where the top words of $\mathbf{v}_1$ and $\mathbf{v}_1$ (words in bold)

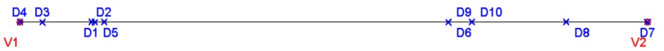

Figure 1: The NO-type convex polytope in one dimension (and hence a line segment) of Experiment 1. Each blue cross is a point that represents a document. V1 and V2 are the endpoints represented by the red circles.

Table II: The values of $\mathbf{v}_1$ and $\mathbf{v}_2$ from Equation (2) in descending order of word strength for NO-type in Experiment 1. Words in bold are the top words that can help interpret the topics as related to *human-computer interaction* and *graphs*.

$\mathbf{v}_1$		$\mathbf{v}_2$	
system	0.21027	**graph**	0.40153
user	0.15735	**trees**	0.32878
human	0.12796	minors	0.16125
interface	0.12507	generation	0.14036
eps	0.11731	random	0.14036
computer	0.10072	survey	0.06329
response	0.10014	response	-0.01487
time	0.10014	time	-0.01487
survey	0.03973	computer	-0.01563
generation	0.00971	interface	-0.02801
random	0.00971	user	-0.03155
minors	-0.00643	human	-0.03175
graph	-0.0444	eps	-0.03709
trees	-0.04727	system	-0.06179

are highly related to the topics of *human-computer interaction* and *graphs*, respectively. This provides a keyword-based approach to interpret the topics. The constituent documents can be viewed as the representative instances of $\mathbf{v}_1$ and $\mathbf{v}_2$ as seen in D4 where the key words of $\mathbf{v}_1$ is in the largest proportion among the documents. Therefore, CPM provides two perspectives to interpret the vertices as topical features.

The combination coefficients for every document which are the columns of matrix $\mathbf{A}$ in Equation (3) are shown in Figure 2 in the form of a stacked bar chart. It is found that D1-D5 have higher proportions in topic associated to V1, and the remaining documents are closer to V2. This is consistent to our expectation where the documents can be respectively categorized into two classes.

We provide the results of LSA for comparisons. LSA approximates the term-document matrix with a low-rank matrix $\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}}$. Here we consider the rank of $\mathbf{M}$ to be 2 because the corpus consists of documents from two categories. The columns of matrix $\mathbf{U}$ are orthonormal and can be interpreted as the vector representation of the *underlying concepts*. The values of the vector indicate the strength of association of the terms with the corresponding concept. Table III displays the concept vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ in descending order of strength. Note that unlike in CPM, the term *human*, *computer* and *interface* have relative lower strengths in $\mathbf{u}_1$ which is more difficult to interpret the concept as related to *human-computer interaction* comparing with CPM.

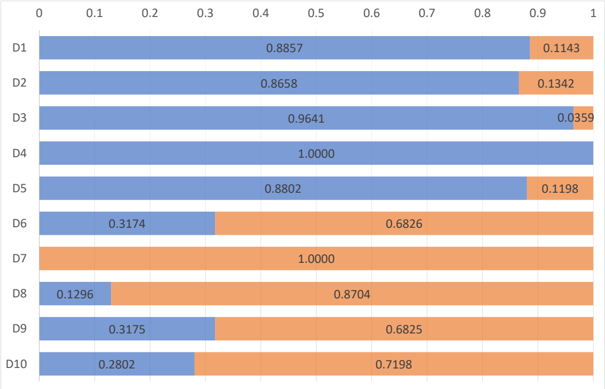

Figure 2: The blue and orange bars represent the combination coefficients as the columns of $\mathbf{A}$ in Equation (3) for the documents with respect to V1 and V2, respectively.

Table III: The LSA concept vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ in descending order of word strength in Experiment 1. We make the same set of words to bold as in Table II

$\mathbf{u}_1$		$\mathbf{u}_2$	
system	0.64381	**graph**	0.65189
user	0.40329	**trees**	0.47957
eps	0.30043	minors	0.3411
response	0.26491	generation	0.30548
time	0.26491	random	0.30548
computer	0.24034	survey	0.16222
human	0.22106	response	0.01464
survey	0.20709	time	0.01464
interface	0.19739	computer	0.00154
graph	0.04597	user	-0.01102
minors	0.03456	interface	-0.03638
trees	0.01767	human	-0.04779
generation	0.00888	eps	-0.06272
random	0.00888	system	-0.08751

B. Experiment 2 – Small Corpus without Clear Categorization

The second corpus consists of 12 short documents among three research topics with no clear categorization, i.e., some documents contain more than one topic (see Table IV). The three topics include *machine translation*, *speech recognition* and *neural networks*. It is well-known that *machine translation* and *speech recognition* are related respectively to *natural language processing* and *speech processing*, which are two different fields. However, there are some recent publications about applying *neural networks* to the both fields. We chose the titles of some of those publications in the corpus, and added some documents that are purely related to *neural networks*. We attempt to show that CPM can extract the topics which are more interpretable than the conventional methods such as LSA and LDA.

Table IV shows the 12 documents in the corpus of Experiment 2. D1-D4 are related to *speech recognition*, and D5-D8 are related to *machine translation*. D9-D12 are about *neural networks* but not *speech recognition* or *machine translation*. Note that D1, D2, D5 and D6 are also related to *neural networks*. Thus, they cannot be purely classified to a specific category.

We generated the term-document matrix like that in Experiment 1. Then, we sum-normalized each column of the term-document matrix and applied PCA to project the columns to a 2-D subspace. Figure 3 displays the NO-type convex polytope in 2-D subspace which has 5 vertices V1-V5 corresponding to D3, D8, D5, D9 and D10, respectively. Observing from the text of the documents, it is obvious that V1 (D3) corresponds to the pure topic of *speech recognition*. Similarly, V2 (D8) is associated to pure topic of *machine translation*. Note that V5 (D10) is a document related to *neural networks* without the term *neural* but with *convolutional*. It is well-known that

Table IV: Corpus of Experiment 2. Words occurring in more than one documents are italicized. The words *neural* and *networks* are in bold.

IDs	Documents
D1	*Speech recognition* with *deep* recurrent ***neural networks***
D2	*Deep* ***neural networks*** for acoustic *modeling* in *speech recognition*
D3	Fundamentals of *speech recognition*
D4	Self-organized language *modeling* for *speech recognition*
D5	***Neural*** *machine translation* by jointly *learning* to align and translate
D6	Effective *approach* to attention-based ***neural*** *machine translation*
D7	A *statistical approach* to *machine translation*
D8	Minimum error rate training in *statistical machine translation*
D9	Sequence to sequence *learning* with ***neural networks***
D10	Visualizing and understanding *convolutional* ***networks***
D11	*Deep learning* in ***neural networks***: an overview
D12	*Learning* semantic representations using *convolutional* ***neural networks*** for web search

convolutional neural networks is a specific type of *neural networks*. As *convolutional* appears in two documents (D10 and D12), it is reasonable for CPM to consider D10 as a topical feature, where the proportion of *convolutional* is the largest. V3 (D5) is related to the *machine translation* based on *neural networks* approach. V4 (D9) is about pure *neural networks* without the term *convolutional*. V3 and V4 are less distinct cf. V1, V2 and V5, and hence can be considered as *sub-topics*.

Figure 4 displays the MVS-type convex polytope. In 2-D subspace, its simplex must have three vertices only. Note that the generated MVS-type has a vertex V1 locating exactly at D3, which also makes it a CNO-type polytope. Comparing with the NO-type, V2 in MVS-type is an extension along the (D3-D9)-direction that eliminates the vertex V3 in the NO-type. Similarly, V3 in MVS-type extends along the (D3-D10)-direction to eliminate the vertex V4 in the NO-type. Figure 5 shows the convex combination coefficients in $\mathbf{A}$ of Equation (3) of MVS-type polytope. The topics represented by V1, V2 and V3 dominate in documents D1-D4, D5-D8 and D9-D12, respectively.

Table V shows the the coordinates of V1-V3, respectively represented by $\mathbf{v}_1$, $\mathbf{v}_2$ and $\mathbf{v}_3$, in descending order of values.

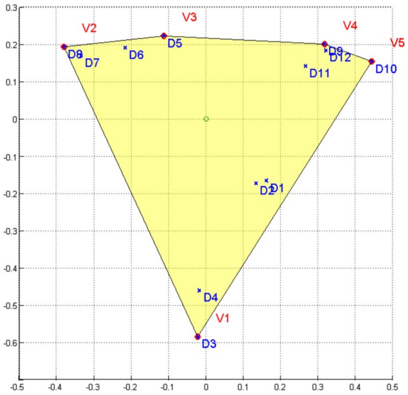

Figure 3: The NO-type convex polytope in 2-D subspace of Experiment 2. V1-V5 are the vertices represented by the red circles.

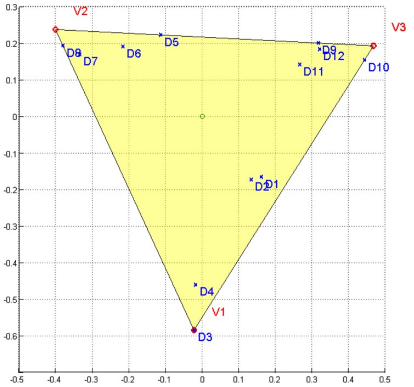

Figure 4: The MVS-type convex polytope in 2-D subspace of Experiment 2. V1-V3 are the vertices represented by the red circles.

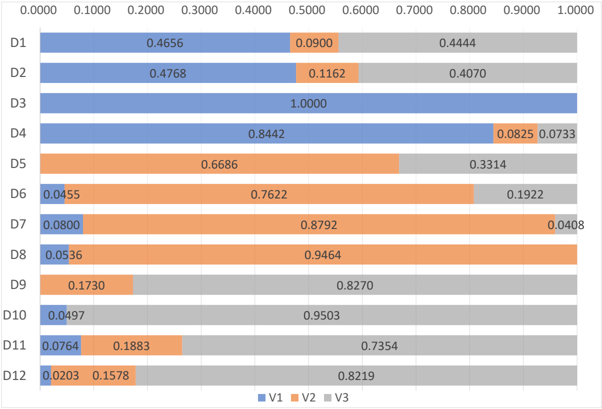

Figure 5: The combination coefficients in $\mathbf{A}$ of MVS-type polytope for Experiment 2.

The top two words indicates the corresponding topics that is consistent with human expectation.

We compared the results of CPM with LSA. Table VI presents the LSA concept vector C1-C3 in descending order of absolute word strength. Note that the word *convolutional* is a top word of the topic related to *neural networks* (V3) in MVS-type CPM but not in that of LSA (C1). This serves to illustrate how CPM can discover more interpretable topics than LSA.

Next, we compared CPM with LDA, where topics are represented as word distributions. We set the number to topics to be 3. Note that unlike in LSA and CPM, we included the terms occurring only once in the corpus as the input. Table VII presents the top 10 words for the 3 obtained topics T1-T3. It is obvious that T1 and T3 are about *machine translation* and *neural networks*, respectively. However, both *machine translation* and *speech recognition* share similar weights, that makes T2 hard to interpret.

Table VIII shows the topic proportions for all documents. As T1 and T2 have overlapping on *machine translation*, this causes D4-D5 and D6-D7 have different prominent topics. D10 is expected to belong to pure *neural networks*, but T2 dominates in D10 because the terms *networks, visualizing* and

Table V: The CPM representations of $\mathbf{v}_1$, $\mathbf{v}_2$ and $\mathbf{v}_3$ in descending order of word strength in Experiment 2.

$\mathbf{v}_1$		$\mathbf{v}_2$		$\mathbf{v}_3$	
speech	**0.4473**	**machine**	**0.3340**	**networks**	**0.4090**
recognition	**0.4473**	**translation**	**0.3340**	**neural**	**0.2558**
modeling	0.1666	statistical	0.2075	**convolutional**	**0.2270**
deep	0.0638	approach	0.1539	learning	0.2230
neural	0.0088	neural	0.1079	deep	0.0933
networks	0.0006	learning	0.0455	modeling	-0.0012
statistical	-0.0004	modeling	-0.0031	speech	-0.0188
approach	-0.0031	deep	-0.0078	recognition	-0.0188
machine	-0.0171	speech	-0.0250	approach	-0.0240
translation	-0.0171	recognition	-0.0250	machine	-0.0458
convolutional	-0.0398	convolutional	-0.0544	translation	-0.0458
learning	-0.0570	networks	-0.0676	statistical	-0.0536

Table VI: The LSA concept vectors of $\mathbf{u}_1$, $\mathbf{u}_2$ and $\mathbf{u}_3$ in descending order of word strength in Experiment 2.

$\mathbf{u}_1$		$\mathbf{u}_2$		$\mathbf{u}_3$	
neural	**0.5706**	**machine**	**0.5425**	**speech**	**0.4650**
networks	**0.4875**	**translation**	**0.5425**	**recognition**	**0.4650**
learning	0.3048	approach	0.2926	modeling	0.2692
deep	0.3036	statistical	0.2813	machine	0.2067
speech	0.2830	learning	0.1256	translation	0.2067
recognition	0.2830	neural	0.1101	statistical	0.1839
machine	0.1572	convolutional	-0.0114	approach	0.1529
translation	0.1572	modeling	-0.1470	deep	0.0590
modeling	0.1495	deep	-0.1626	neural	-0.1958
convolutional	**0.1109**	networks	-0.1677	convolutional	-0.2240
approach	0.0744	speech	-0.2667	networks	-0.2886
statistical	0.0422	recognition	-0.2667	learning	-0.4208

Table VII: The top 10 words in LDA topics of T1, T2 and T3 in Experiment 2.

T1		T2		T3	
machine	**0.0873**	**translation**	**0.0662**	**neural**	**0.1249**
translation	**0.0872**	**machine**	**0.0661**	**networks**	**0.1108**
statistical	0.0582	**recognition**	**0.0655**	learning	0.0745
training	0.0569	**speech**	**0.0635**	deep	0.0692
rate	0.0568	networks	0.0446	speech	0.0562
error	0.0566	approach	0.0425	recognition	0.0549
minimum	0.0564	visualizing	0.0421	sequence	0.0486
attention-based	0.0468	statistical	0.0420	modeling	0.0348
approach	0.0462	understanding	0.0416	convolutional	0.0283
neural	0.0444	fundamentals	0.0416	acoustic	0.0280

Table VIII: The LDA topic proportions for the documents in Experiment 2.

	T1	T2	T3
D1	0.0483	0.0515	**0.9000**
D2	0.0422	0.0454	**0.9120**
D3	0.0842	**0.8220**	0.0939
D4	0.0564	**0.8790**	0.0650
D5	0.0464	**0.8760**	0.0780
D6	**0.8900**	0.0528	0.0575
D7	**0.8520**	0.0797	0.0679
D8	**0.9140**	0.0439	0.0420
D9	0.0562	0.0571	**0.8870**
D10	0.0677	**0.8460**	0.0868
D11	0.0563	0.0571	**0.8870**
D12	0.0377	0.0390	**0.9230**

understanding are in the top words of T2. The reason for the unsatisfactory performance of LDA may also deal to the small size of the corpus. It is well-known that LDA does not perform well in short texts [3,4]. In the experiment above, we assumed the topic proportions follows a symmetric Dirichlet distribution, which is implicitly defined in the selected hyper-parameters. However, this presumption affects the topic proportions of the document. For example, one would expect D1 has similar ratio between *speech recognition* and *neural networks*. However, T3 dominates D1 with proportion about 90% in the LDA, while in CPM the ratio between *speech recognition* and *neural networks* is about 1 : 1 (see Figure 5). Hence, CPM is more robust than CPM in terms of selection of hyper-parameters.

IV. Conclusions

This paper presents a novel topic discovery approach that embeds documents in low-dimensional affine subspace and generates a compact convex polytope to enclose all the embedded documents. We show that the resulting topics as the vertices of the polytope are more representative and interpretable than the existing methods on two small corpora with short texts. The proposed model has several advantages. First, it is transparent as the process can be visualized and understood in terms of geometry. This enhances the interpretability of the model which is useful to justify the results. Second, it is robust as the NO-type polytope always exists and is unique, avoiding the selection of hyper-parameters. Third, the resulting topics are consistent with human expectation even in corpus without clear categorization, which enhance the topic interpretability. Thus, this model can be conceivably applied to complex corpora where the topics are highly overlapped. In future work, we will test the scalability of CPM with large real-world corpora. Moreover, we will investigate the possibility to incorporate tensor product to CPM as inspired by the tensor product model transformation in polytopic model-based control [5].

References

[1] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, pp. 391–407, 1990.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *Journal of machine learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[3] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proceedings of the first workshop on social media analytics*. ACM, 2010, pp. 80–88.

[4] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *International conference on World Wide Web*. ACM, 2013, pp. 1445–1456.

[5] P. Baranyi, Y. Yam, and P. Várlaki, *Tensor product model transformation in polytopic model-based control*. CRC Press, 2013.

[6] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The Quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software (TOMS)*, vol. 22, no. 4, pp. 469–483, 1996.

[7] J. Li and J. M. Bioucas-Dias, "Minimum volume simplex analysis: A fast algorithm to unmix hyperspectral data," in *IEEE International Geoscience and Remote Sensing Symposium*, vol. 3. IEEE, 2008, pp. III–250.

[8] J. Li, A. Agathos, D. Zaharie, J. M. Bioucas-Dias, A. Plaza, and X. Li, "Minimum volume simplex analysis: A fast algorithm for linear hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 9, pp. 5067–5082, 2015.

[9] S. Nagy, Z. Petres, and P. Baranyi, "TP Tool-a Matlab toolbox for TP model transformation," in *International Symposium of Hungarian Researchers on Computational Intelligence and Informatics*, 2007, pp. 483–495.

[10] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in *LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer, 2010.