Temporal Multinomial Mixture for Instance-oriented Evolutionary Clustering

Young-Min Kim[†], Julien Velcin[‡], Stéphane Bonnevay[‡], and Marian-Andrei Rizoiu[‡]

[†]Korea Institute of Science and Technology Information, South Korea
[‡]ERIC Lab., University of Lyon 2, France
ymkim@kisti.re.kr
{julien.velcin,stephane.bonnevay,marian-andrei.rizoiu}@univ-lyon2.fr

Abstract. Evolutionary clustering aims at capturing the temporal evolution of clusters. This issue is particularly important in the context of social media data that are naturally temporally driven. In this paper, we propose a new probabilistic model-based evolutionary clustering technique. The Temporal Multinomial Mixture (TMM) is an extension of classical mixture model that optimizes feature co-occurrences in the trade-off with temporal smoothness. Our model is evaluated for two recent case studies on opinion aggregation over time. We compare four different probabilistic clustering models and we show the superiority of our proposal in the task of instance-oriented clustering.

Keywords: Evolutionary clustering, mixture model, temporal analysis.

1 Introduction

Clustering is a popular way to preprocess large amount of unstructured data. It can be used in several ways, such as data summarization for decision making or representation learning for classification purpose. Recently, evolutionary clustering aiming at capturing temporal evolution of clusters in data streams begun to make a mark. In contrast to incremental clustering, evolutionary clustering methods optimize a different measure [1–3]. They build a clustering model at time $t+1$ by taking into account of the model at time t, in a retrospective manner. Applications range from clustering photo tags in `flickr.com` to document clustering in textual corpora.

The existing methods fall into two different categories. *Instance-oriented* evolutionary clustering mostly aims at primarily regrouping objects and *topic-oriented* evolutionary clustering aims at estimating distributions over components (*e.g.*, words). While the former extracts tightest clusters in the feature space, the latter improves the smoothness of temporally consecutive clusters. In this work, we focus on developing a new temporal-driven model of the first category, motivated by two case studies.

We propose a new probabilistic evolutionary clustering method aiming at finding dynamic instance clusters. Our model, Temporal Mixture Model (TMM),

is an extension of classical mixture model to categorical data streams. The main novelty is not to use Dirichlet prior in order to relax smoothness constraint. While our model can further be improved in terms of more advanced properties, such as learning the number of clusters as in non-parametric models [4, 5], in this work we mainly focus on realizing our basic idea and studying the performance of the model. Using internal evaluation measures, we demonstrate that TMM outperforms a typical *topic-oriented* dynamic model and achieves similar compactness results with two static models. This result is achieved at the slight expense of cluster smoothing ability through temporal epochs.

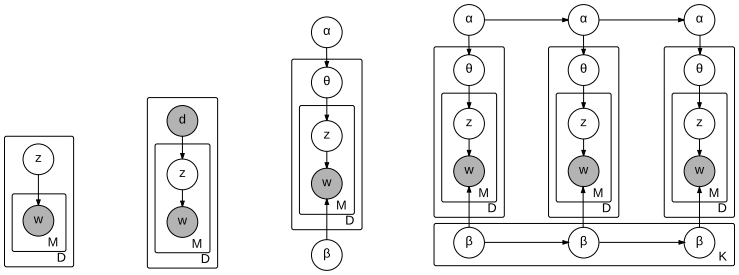
In the following sections, we first motivate and present in detail the proposed TMM model. Then we present the experimental results of TMM as well as three other methods of the literature, showing the superiority of our method with new type of datasets in opinion mining. Finally we conclude with some perspectives and future works.

2 Motivation and related work

2.1 Motivation

Document clustering and topic extraction are sometimes considered as equivalent problems, and the methods desired to address each problem are used interchangeably [6]. However, there is a fundamental difference in terms of clustering objective between them and this draws a clear algorithmic difference. Even though this issue has not been actively mentioned in the clustering literature, it is indirectly confirmed by the fact that topic modeling is not recommended to be used directly for document clustering in general. [7] have empirically shown that even simple mixture models outperform Dirichlet distribution-based topic models for document clustering, when directly using model parameters. A recent work [8] is dealing with this issue by proposing an integrated graphical model for both document clustering and topic modeling. However, the great success of topic models in unsupervised learning has often led researchers to use them as instance clustering in practice. This observation remains valid for evolutionary clustering, for which one hardly finds an alternative to topic models using Dirichlet smoothing. The situation is identical when dealing with more classical categorical data, which is the case of our work. This paper starts from this significant issue in evolutionary clustering.

To the best of our knowledge, this is the first attempt to use a non-Dirichlet mixture model for temporal analysis of data steams. The reason why we abandon Dirichlet prior reflects our (maybe peculiar) point of view towards the Dirichlet distribution. That is, the power of topic models mainly comes from their ability to smoothen distributions via the Dirichlet prior. It is effective for extracting representative topics or for making prediction on new data. However, in case of clustering instances, a hasty smoothing of the distributions risks to mix data samples with no common feature. In this paper, target datasets are not necessarily textual; therefore the clustering process can be more sensitive to this effect than when dealing with a large feature space (such as a vocabulary of words). In



(a) (b) (c) (d)

Fig. 1: Graphical representation of (a) MM, (b) PLSA, (c) LDA, and (d) DTM.

our case, each feature becomes more important, thus special attention must be given to the actual matching between the cluster distribution and the observed feature co-occurrences. This is the reason why we decide to build our method on top of a simple mixture model expecting to minimize the discussed risk.

2.2 Related work

Our new evolutionary clustering model, *Temporal Multinomial Mixture* (TMM), has been designed with the assumption that regrouping non co-occurring features is highly prejudicial. TMM is a temporal extension of the *Multinomial Mixture* (MM), a simple probabilistic generative model for clustering. More complex mixture models such as *Probabilistic Latent Semantic Analysis* (PLSA) [9] or *Latent Dirichlet Allocation* (LDA) [10] seem less suitable for analyzing non-textual data as mentioned in Section 2.1. In these models, non co-occurring features are often mixed together in a same cluster because of additional hidden layers, realized as instance-topic distributions (PLSA) or Dirichlet prior (LDA). The graphical representation of these models are given in Fig. 1(a)-(c).

Despite the obvious difference between our purpose and dynamic topic models, since the temporal approaches in unsupervised learning usually stand on the basis of topic models, it is inevitable to introduce the state-of-the arts of topic models. Most of the current techniques in clustering introducing a temporal dimension are topic models taking Dirichlet distribution [11, 12] since the development of *Dynamic Topic Model* (DTM, Fig. 1(d)) [13], a simple extension of LDA. This kind of dynamic topic analysis has been the object of numerous studies over recent years and more complex models such as DMM [11] or MDTM [12] have been developed. In comparison, TMM is much simpler and we experimentally show the power of simple modeling by comparing three clustering methods, MM, PLSA and DTM with ours.

On the other hand, some pioneer works were designed for data points basically last during more than two time periods. These stand on various theoretical bases such as k-means, agglomerative hierarchical method, spectral clustering, and even generative model [1, 2, 14]. However, the underlined property of data points is contrary to the case of data stream, which is our concern.

Whatsoever, most applications in temporal analysis are intended for text document analysis. Being designed for text hinders the "out-of-the-box" application of these methods to unfamiliar data such as image, gene, market, network data etc. In comparison, TMM is an evolutionary clustering dedicated to general categorical datasets.

3 Temporal Multinomial Mixture

We propose Temporal Multinomial Mixture (TMM) for instance-oriented clustering over time. TMM is a temporal extension of MM and the relation between TMM and MM is analogous to that between DTM and LDA. While the majority of existing temporal topic analysis tend to complicate the modeling process, TMM rather goes against this trend. We assume that complicated distributional structures confuse the instance-oriented clustering. Therefore our method assumes the form of a simple mixture model. As in many other evolutionary clusterings and temporal topic analysis, data instances are associated with a time epoch. A time epoch indicates a time period between two adjacent moments. Dataset is generally divided into subsets by epoch. Instances are assumed to be described by features weighted with a frequency[1].

3.1 Graphical model

The graphical representation of TMM is given in Fig. 2. The extension from MM is realized by encoding the temporal dependency into the relation between data components w of the current epoch and the clusters z of the previous epoch. The generation process of an instance $d^t = i$ at the epoch t is as follows:

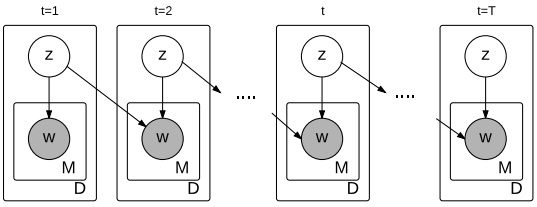


Fig. 2: Graphical representation of a temporal multinomial mixture model

- Choose a cluster z_i^{t-1} with probability $p(z_i^{t-1})$.
- Choose a cluster z_i^t with probability $p(z_i^t)$.
- Generate an instance $d^t = i$ with probability $p(d^t = i|z_i^{t-1}, z_i^t)$ when $t > 1$ or with $p(d^1 = i|z_i^1)$ when $t = 1$.

[1] For the sake of understanding, the reader can see a feature as a unique word over a vocabulary and a data component as a word occurrence in a document even if an instance is not a document here.

Table 1: Notations

Symbol	Description
d^t	instance d at epoch t
w_{im}^t	mth component in the instance $d^t{=}i$ at epoch t
z_i^t	assigned cluster for instance $d^t{=}i$ at epoch t
D^t	sequence of instances at epoch t
Z^t	sequence of cluster assignments for D^t
$\mathbf{D}$	sequence of all instances, $\mathbf{D} = (D^1, D^2, ..., D^T)$
$\mathbf{Z}$	sequence of cluster assignments for $\mathbf{D}$, $\mathbf{Z} = (Z^1, Z^2, ..., Z^T)$
T	number of epochs
$\|D^t\|$	number of instances at epoch t
M_d^t	number of components in instance d at epoch t
V	number of unique components (number of features)
K	number of clusters
$\boldsymbol{\phi}_k^t$	multinomial distribution of cluster k over components at epoch t
π_k^t	prior probability of cluster k at epoch t
α	weight for the component generation from the clusters of previous epoch, $0 < \alpha < 1$

The last step is realized by repeatedly generating the components $w_{im}^t, \forall m$, sequential features in the instance $d^t = i$, as illustrated in the graphical representation. Unlike most temporal graphical models, it is a connected network considering the correlation of all topics of t and $t-1$. The notations used in TMM are shown in Table 1. We mostly referred the notations in [15] and [16]. Because of the variable dependency between different time epochs, we need sequential expression of features. Therefore we can not use the simple notation of MM, that does not require sequential component representations.

3.2 Parameter estimation via approximate development

The objective function to be maximized is the expectation of log-likelihood [17]:

$$\mathbb{E}(\tilde{\mathcal{L}}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{D}, \Theta_{old}) \cdot \log\left(p(\mathbf{D}, \mathbf{Z}|\Theta)\right) \tag{1}$$

Because of the dependency between the variables z^t and z^{t-1}, the log-likelihood cannot be simplified using marginalized latent variables as in MM or PLSA. Instead, we start with the joint distribution of instances and assigned clusters (latent variables):

$$p(\mathbf{D}, \mathbf{Z}) = \left\{ \prod_{d=1}^{|D^1|} p(z_d^1) \cdot p(d^1|z_d^1) \right\} \left\{ \prod_{t=2}^{T} \prod_{d=1}^{|D^t|} p(z_d^t) \cdot p(d^t|z_d^t, z_d^{t-1}) \right\} \tag{2}$$

Eq. 1 can be simplified by taking only the valid latent variables per term:

$$\begin{aligned} \mathbb{E}(\tilde{\mathcal{L}}) = & \sum_{i=1}^{|D^1|} \sum_{k=1}^{K} p(z_i^1 = k|d^1 = i) \log\{p(z_i^1 = k) p(d^1 = i|z_i^1 = k)\} \\ & + \sum_{t=2}^{T} \sum_{i=1}^{|D^t|} \sum_{k=1}^{K} \sum_{k'=1}^{K} p(z_i^t{=}k, z_i^{t-1}{=}k'|d^t{=}i) \log\{p(z_i^t{=}k) p(d^t{=}i|z_i^t{=}k, z_i^{t-1}{=}k')\} \end{aligned} \tag{3}$$

At epoch 1, $p(d^1{=}i|z_i^1{=}k)$ can be rewritten using ϕ_k^1 and $n_{i,j}^1$, the frequency of unique component j included in instance i, such as $\prod_{j=1}^{V}(\phi_{k,j}^1)^{n_{i,j}^1}$. On the other hand, the instance generation at epoch t, $\forall t \geq 2$ is dependent also on the cluster of the previous epoch. Thus the conditional probability of an instance i given current and previous clusters k and k', is inferred as follows with Bayes Rules:

$$p(d^t{=}i|z_i^t{=}k,z_i^{t-1}{=}k') = \prod_{m=1}^{M_i^t} \frac{p(z_i^t{=}k|w_{im}^t,z_i^{t-1}{=}k')p(z_i^{t-1}{=}k'|w_{im}^t)p(w_{im}^t)}{p(z_i^t{=}k,z_i^{t-1}{=}k')} \quad (4)$$

Under the assumptions of graphical model, the analytical calculation of $p(z_i^t|w_{im}^t, z_i^{t-1})$ is so complicated because the latent variables are related by the explaining away effect. To tackle this issue, we make an important hypothesis that $p(z_i^t|w_{im}^t, z_i^{t-1})$ can be `approximated` by $p(z_i^t|w_{im}^t)$. Consequently, Eq. 4 is rewritten using $p(w_{im}^t{=}j|z_i^t{=}k)$ as well as $p(w_{im}^t{=}j|z_i^{t-1}{=}k')$, which is equivalent to the previous epoch's parameter $\phi_{k',j}^{t-1}$. Penalizing the influence rate of the previous cluster with α, a weighted parameter value $(\phi_{k',j}^{t-1})^{\alpha}, 0 < \alpha < 1$ is used instead of $\phi_{k',j}^{t-1}$. Letting the constant $\prod_{m=1}^{M_i^t} 1/p(w_{im}^t)$ be C_i^t, we obtain the following equation.

$$p(d^t = i|z_i^t = k, z_i^{t-1} = k') = C_i^t \cdot \prod_{j=1}^{V} (\phi_{k,j}^t)^{n_{i,j}^t} (\phi_{k',j}^{t-1})^{\alpha \cdot n_{i,j}^t} \quad (5)$$

Using the parameters Θ, the $\mathbb{E}(\tilde{\mathcal{L}})$ becomes:

$$\mathbb{E}(\tilde{\mathcal{L}}){=}\sum_{i=1}^{|D^1|}\sum_{k=1}^{K} p(z_i^1{=}k|d^1{=}i) \cdot \left\{ \log \pi_k^1 + \sum_{j=1}^{V} n_{i,j}^1 \cdot \log \phi_{k,j}^1 \right\}$$
$$+\sum_{t=2}^{T}\sum_{i=1}^{|D^t|}\sum_{k=1}^{K}\sum_{k'=1}^{K} p(z_i^t{=}k,z_i^{t-1}{=}k'|d^t{=}i)\cdot\left\{ \log\pi_k^t{+}\log C_i^t{+}\sum_{j=1}^{V} n_{i,j}^t\cdot\big(\log\phi_{k,j}^t{+}\alpha\cdot\log\phi_{k',j}^{t-1}\big) \right\}$$

3.3 EM algorithm

We solve the following optimization problem to obtain the parameter values.

$$\arg\max_{\Theta} \ \mathbb{E}(\tilde{\mathcal{L}}), \quad \text{subject to} \sum_{j=1}^{V} \phi_{k,j}^t = 1, \ \forall t, k \quad \text{and} \quad \sum_{k=1}^{K} \pi_k^t = 1, \ \forall t.$$

The EM algorithm is updated as follows.

Initialization
Randomly initialize parameters $\Theta = \{\boldsymbol{\phi}_k^t, \pi_k^t \mid \forall t, k\}$

$$\text{subject to} \quad \sum_{j=1}^{V} \phi_{k,j}^t = 1, \quad \forall t, k \quad \text{and} \quad \sum_{k=1}^{K} \pi_k^t = 1, \quad \forall t.$$

E-step
Compute the expectation of posteriors as follows.

$$p(z_i^t{=}k, z_i^{t-1}{=}k'|d^t{=}i){=}\frac{\prod_{j=1}^{V}(\phi_{k,j}^t)^{n_{i,j}^t}(\phi_{k',j}^{t-1})^{\alpha\cdot n_{i,j}^t}\cdot\pi_k^t\cdot\pi_{k'}^{t-1}}{\sum_{a=1}^{K}\sum_{a'=1}^{K}\prod_{j=1}^{V}(\phi_{a,j}^t)^{n_{i,j}^t}(\phi_{a',j}^{t-1})^{\alpha\cdot n_{i,j}^t}\cdot\pi_a^t\cdot\pi_{a'}^{t-1}}, 2{\le}t{\le}T, \forall k, k', i. \tag{6}$$

$p(z_i^1 = k|d^1 = i)$ is similarly calculated by eliminating the variables of $t-1$.

M-step
Update the parameters maximizing the objective function.

$$\phi_{k,j}^t = \frac{\sum_{i=1}^{|D^t|}\sum_{k'=1}^{K} n_{i,j}^t{\cdot}p(z^t{=}k,z^{t-1}{=}k'|d^t{=}i)+\sum_{i=1}^{|D^{t+1}|}\sum_{k'=1}^{K}\alpha{\cdot}n_{i,j}^{t+1}{\cdot}p(z_i^{t+1}{=}k',z_i^t{=}k|d^{t+1}{=}i)}{\sum_{i=1}^{|D^t|}\sum_{j'=1}^{V}\sum_{k'=1}^{K} n_{i,j'}^t{\cdot}p(z^t{=}k,z^{t-1}{=}k'|d^t{=}i)+\sum_{i=1}^{|D^{t+1}|}\sum_{j'=1}^{V}\sum_{k'=1}^{K}\alpha{\cdot}n_{i,j'}^{t+1}{\cdot}p(z_i^{t+1}{=}k',z_i^t{=}k|d^{t+1}{=}i)},$$
$$2 \le t \le T-1, \quad \forall j, k. \tag{7}$$

$\phi_{k,j}^1$ is calculated by eliminating the variables of $t-1$ from the above formula and $\phi_{k,j}^T$ is done by eliminating both variables and terms of $t+1$.

$$\pi_k^t{=}\frac{\sum_{i=1}^{|D^t|}\sum_{k'=1}^{K} p(z^t{=}k,z^{t-1}{=}k'|d^t{=}i)+\sum_{i=1}^{|D^{t+1}|}\sum_{k'=1}^{K} p(z_i^{t+1}{=}k',z_i^t{=}k|d^{t+1}{=}i)}{\sum_{i=1}^{|D^t|}\sum_{a=1}^{K}\sum_{k'=1}^{K} p(z^t{=}a,z^{t-1}{=}k'|d^t{=}i)+\sum_{i=1}^{|D^{t+1}|}\sum_{k'=1}^{K}\sum_{a=1}^{K} p(z_i^{t+1}{=}k',z_i^t{=}a|d^{t+1}{=}i)},$$
$$2 \le t \le T-1, \quad \forall k \tag{8}$$

π_k^1 and π_k^T are calculated as in $\phi_{k,j}^1$ and $\phi_{k,j}^T$.

3.4 Instance assignment and cluster evolution

The assignment of each instance is eventually obtained from the estimated distributions. For $t = 1$, we assign to the instance i the cluster that maximizes the posterior probability $p(z_i^1{=}k|d^1{=}i)$. For the instances in the other epochs, we integrate out z_i^{t-1} to obtain the instance cluster such that $p(z_i^t{=}k|d^t{=}i){=}\sum_{k'=1}^{K} p(z_i^t{=}k, z_i^{t-1}{=}k'|d^t{=}i)$.

TMM being a connected network, all the clusters in the epoch $t-1$ can contribute to the clusters in the epoch t. Please note that the same cluster index in different epochs does not mean that the corresponding clusters are identical over time. That is why we need to find which cluster of the previous epoch contributes most to the specific cluster k of the current epoch. The dynamic correlation between clusters of the adjacent epochs is fully encoded in the distribution $p(z_i^t{=}k, z_i^{t-1}{=}k'|d^t{=}i)$. By integrating out z_i^t instead of z_i^{t-1}

from $p(z_i^t{=}k, z_i^{t-1}{=}k'|d^t{=}i)$, we can deduce the most likely cluster at the previous epoch for the instance $d^t{=}i$. We call it the origin of the instance. Given the specific cluster $z^t = k$, we have the classified instances and their origins. By counting the most frequent origin, we can finally relate the most influential cluster of the previous epoch to $z^t = k$. Since this is a surjective function from t to $t-1$, the division of a cluster over time is traceable. Conversely, the merge of multiple clusters can also be caught if we choose not only the most likely cluster but also the second or the third likely one.

4 Experiments

We compare four different generative models in order to evaluate the performance of TMM. DTM is selected as a Dirichlet-based model; MM and PLSA are used as static baselines for highlighting the effect of introducing a temporal dimension. Finally, we show that TMM outperforms the other models on two datasets of opinion mining, by finding a trade-off between compactness and temporal smoothing.

4.1 Datasets

ImagiWeb political opinion dataset[2] The first dataset is comprised of a set of about 7000 unique tweets related to two politicians (each politician is analyzed separately). The manual annotation process has been supervised by domain experts of public opinion analysis and it has followed a detailed procedure with the design of 9 aspects (*e.g.*, project, ethic or political line) targeted by 6 possible opinion polarities (-2=very negative, -1=negative, 0=neutral, +1=positive, +2=very positive, NULL=ambiguous). For instance, the tweet "RT @anonym: P's project is just hot air" can be described by the pair `(project,-2)` attached to the politician P. Each pair corresponds to a feature w whose value is the occurrence of the corresponding opinion for describing the studied entity. The full procedure and dataset are described in [18]. Because of the length limit of a tweet as well as for clustering purpose, we decide to combine the annotations by author for each time epoch.

RepLab 2013 Corpus This corpus has been used for the RepLab 2013, second evaluation campaign on Online Reputation Management. It consists of a collection of tweets referring to 61 entities from four domains. We select two dominant domains out of four, automative and music, where the number of entities is 20 respectively. The clustering is done for each *domain* separately this time instead of entity. Tweets are annotated with three polarities, positive, negative and neutral. We let the features be the *entity-polarity* pairs instead of aspect-polarity pairs, so that the opinion aggregation is based on co-occurring entities. It means that the opinion groups are constructed by users, who are interested in same entities with similar polarities. Tab. 2 sums up basic statistics on the two datasets.

[2] It will be distributed to the public in Spring 2015 on the ImagiWeb official website, http://mediamining.univ-lyon2.fr/velcin/imagiweb.

Table 2: Statistics of datasets and features we define.

	ImagiWeb opinion dataset	RepLab 2013
source	Political opinion tweets	English & Spanish opinion tweets
annotation size	11527 tweets (7283 unique)	26709 tweets (all unique)
subsets	Entity (politician P, politician Q)	Domain (automative, music)
feature space	Aspect-polarity pair	Entity-polarity pair
	9 aspects, 6 polarities	20 entities per domain, 3 polarities

4.2 Evaluation Measures

The ground truth is hardly available when evaluating clustering output for evolutionary clustering. We instead develop the following three quantitative measures with the object of well detecting clustering quality.

Co-occurrence level. Our main interest lies in detecting compact clusters, which means that the number of observed co-occurring features actually match the estimated distribution. This measure counts the real number of co-occurring feature couples in each sample among the non-zero features grouped in a cluster.

Unsmoothness. This catches the dissimilarity between corresponding clusters through different time epochs using Kullback-Leibler (KL) divergence. If a temporal clustering method detects well the evolution of clusters, the cluster signatures having same identity would be similar to each other. Therefore we develop 'unsmoothness' to measure how suddenly a cluster changes over time.

Homogeneity. This measures the degree of unanimity of grouped tweets in a cluster in terms of polarity. Opposite opinions hardly co-occur because an author usually keep his opinion stance in a sufficiently short time. By ignoring the degree of polarity, the homogeneity of a cluster is simply defined as follows[3]:

$$\text{Homogeneity} = (|\#(\text{positive}) - \#(\text{negative})|)/(\#(\text{positive}) + \#(\text{negative}))$$

This is intuitive and easy to be visually represented but is an indirect evaluation.

4.3 Result

Clustering is conducted at subset level. For a given clustering method and subset, experiments are repeated 10 times by changing initialization to get the statistical significance. Since MM and PLSA are time-independent, temporal clusters are obtained via two stages: normal clustering per epoch and heuristic matching between clusters of two adjacent epochs judged by their distributional form.

The first sub-table of Table 3 shows the experimental results of four methods on ImagiWeb dataset. Once clustering is done per subset, we merge the results to analyze together the reputation of two competitors. The number of epochs is fixed at two by splitting data by an actual important political event date. Each value is the averaged result of 10 experiments as well as the standard deviation in brackets. Bold number indicates the best result among four methods and the underlined one is the second best. The gray background of bold number means the result statistically outperforms the second best and the light-gray means it does not outperform the second best, but does the third one. The

[3] #(polarity) is the number of tweets annotated with this polarity.

Table 3: Evaluation of temporal clustering for four methods on ImagiWeb opinion dataset(left) and RepLab 2013 for automative(middle) and music(right).

	ImagiWeb opinion dataset				RepLab(Auto)				RepLab(Music)			
	TMM	DTM	MM	PLSA	TMM	DTM	MM	PLSA	TMM	DTM	MM	PLSA
Avg. Homogen.	**0.86**	0.70	**0.86**	0.67	**0.76**	0.67	0.73	0.70	**0.77**	0.75	0.75	0.76
(stand. deviation)	(0.02)	(0.06)	(0.02)	(0.05)	(0.02)	(0.05)	(0.03)	(0.04)	(0.03)	(0.05)	(0.02)	(0.03)
Co-occurr. level	**123**	113	122	111	**40**	34	**40**	33	**26**	22	25	22
(stand. deviation)	(1.98)	(1.02)	(0.88)	(1.48)	(1.21)	(1.18)	(0.58)	(1.52)	(0.74)	(0.80)	(0.40)	(0.35)
Avg. Unsmooth.	2.27	**1.57**	3.16	3.61	4.30	**1.37**	6.35	6.91	4.5	**2.54**	6.12	7.75
(stand. deviation)	(0.23)	(0.10)	(0.33)	(0.21)	(0.90)	(0.12)	(0.82)	(0.69)	(0.90)	(0.51)	(0.87)	(1.11)

value of α in TMM has been set to 0.7 after several pre-experiments judged by visual representation of clusters (as shown in Fig. 3) as well as balance among cluster sizes. We manually choose the value by varying α from 0.5 to 1. Larger value increases distributional similarity whereas decreases separation of opposite opinions. The hyper parameters of DTM have also been set to best ones after several experiments.

Globally, TMM outperforms the others in terms of two measures except unsmoothness. Then DTM and MM are in the second place. PLSA produces the worst result for all measures. Since homogeneity is a direct basis to evaluate if the tested method detects well the difference between negative and positive opinion groups, it becomes more important when the mix of opposite opinions is a crucial error. Co-occurrence level also directly shows if the captured clusters are really based on the co-occurring features. Given that both measures evaluate cluster quality of a specific time epoch, it is encouraging that TMM provides identical or even slightly better result than MM because TMM can be thought of as a relaxed version of MM in the point of view of data adjustment over time. The result therefore demonstrates that TMM successfully makes use of the generative advantage of MM. For homogeneity, TMM and MM obtaining 0.86 both perfectly outperform the second best DTM in terms of Mann-Whitney test with the p-value of 0.00001. Meanwhile, for unsmoothness the best one is DTM with a clearly better result, 1.57 than the others. DTM concentrates on the distribution adjustment over time at the expense of well grouping opinions that is the principal objective in the task. The second best TMM also perfectly outperforms MM with the p-value of 0.0002. It proves the time dependency encoded in TMM successfully enhances MM for capturing cluster evolution.

Besides the quantitative evaluation, we visualize a TMM clustering result in Fig. 3. It is the evolution of two clusters with five different time epochs on politician P subset. The zoomed figure shows a negative group about P at epoch 1 especially on the aspects "political line" and "project". TMM captures the dynamics of the cluster over time as shown in the figure. As time goes by, opinions about "project" disappear (at t=5) but the other negative opinions about "ethic" appear in the cluster. The cluster in the second line groups mainly positive and neutral opinions about various aspects at epoch 1, but some aspects gradually disappear with time.

The experimental results on RepLab 2013 corpus are given in the middle and right sub-tables in Table 3. Number of epochs is also fixed at two and the

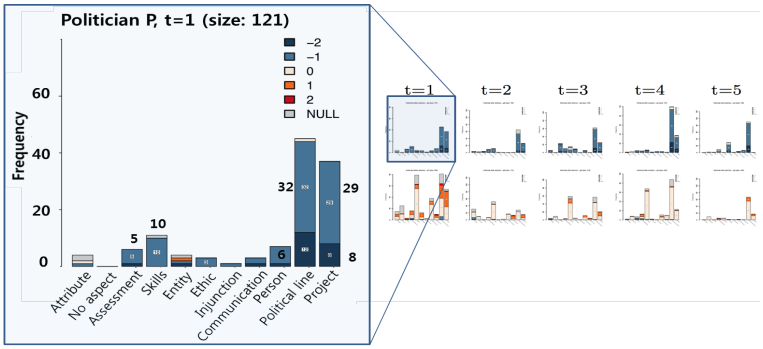


Fig. 3: Visualization of the evolution of two clusters extracted from a TMM clustering result with five different time epochs on politician P subset.

data is split by the median date. This corpus is not originally constructed for opinion aggregation, therefore we do not have sufficient feature co-occurrences. The proportion of instances having at least two components is only 5.2% for automative and 2.9% for music. Despite the handicap, we rather expect that we would emphasize the characteristics of each model via experiments with this restrictive dataset. The α value has been set to 1 to make maximum use of the effect of previous clusters regarding lack of co-occurrences.

Two outstanding methods are TMM and DTM but there is an evident difference between their results. TMM gives a better performance in terms of local clustering quality such as homogeneity and co-occurrence whereas DTM outperforms the others in temporal view. Homogeneity does not seem really meaningful here because the opposite opinions about different entities can be naturally mixed in an opinion group. However, from the fact that co-occurring features are rarely observed and, moreover, only 10% of total opinions are negative in the corpus, negative and positive opinions seldom co-occur. Therefore, the high homogeneity can be a significant measure here also. As in ImagiWeb dataset, the co-occurrence level of TMM is clearly better than that of DTM. On the other hand, even though DTM gives a perfectly better result for unsmoothness, the captured distributions are not really based on the real co-occurrences when we manually verify the result. Nevertheless, when the dataset is extremely sparse as in this case, smoothing distribution would anyway provide the opportunity not to ignore rarely co-occurring features.

5 Conclusions

The proposed TMM model succeeds in effectively extending MM, by taking into consideration the temporal factor for clustering. Our method captures the dynamics of clusters much better than the heuristic matching of single clustering results using MM or PLSA, without losing clustering quality at local time epoch. TMM clearly outperforms DTM in terms of local cluster quality. DTM tends to produce well-smoothed distributions over time, but as shown through its low performance with the other measures, high smoothness does not always signify that the cluster evolution is well detected.

An inherent hypothesis in TMM is that clusters evolve progressively over time and it has enabled the modeling of direct dependency between two adjacent epochs. However if abrupt changes arrive, the distributions found for each cluster can be incoherent. A future developmental direction is taking such changes into account. A possible way could be to establish an automatic adjustment of the dependency rate α. Another interesting direction is to develop means to infer more exactly the conditional probability $p(z_i^t|w_{im}^t, z_i^{t-1})$.

Acknowledgments. This work was partially funded by the project ImagiWeb ANR-2012-CORD-002-01.

References

1. Chakrabarti, D., Kumar, R., Tomkins, A.: Evolutionary clustering. In: KDD '06, ACM (2006) 554–560
2. Chi, Y., Song, X., Zhou, D., Hino, K., Tseng, B.L.: Evolutionary spectral clustering by incorporating temporal smoothness. In: KDD '07, ACM (2007) 153–162
3. Xu, T., Zhang, Z.M., Yu, P.S., Long, B.: Dirichlet process based evolutionary clustering. In: ICDM '08, IEEE Computer Society (2008) 648–657
4. Teh, Y., M. Jordan, M.B., Blei, D.: Hierarchical Dirichlet processes. Journal of the American Statistical Association **101**(476) (2006) 1566–1581
5. Ahmed, A., Xing, E.: Dynamic non-parametric mixture models and the recurrent chinese restaurant process : with applications to evolutionary clustering. In: SIAM International Conference on Data Mining. (2008)
6. Zhang, J., Song, Y., Zhang, C., Liu, S.: Evolutionary hierarchical Dirichlet processes for multiple correlated time-varying corpora. In: KDD '10, ACM 1079–1088
7. Pessiot, J.F., Kim, Y.M., Amini, M.R., Gallinari, P.: Improving document clustering in a learned concept space. Inform. Process. & Manag. **46**(2) (2010) 180–192
8. Xie, P., Xing, E.P.: Integrating document clustering and topic modeling. In: UAI. (2013)
9. Hofmann, T.: Probabilistic latent semantic analysis. In: UAI '99. (1999) 289–296
10. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3** (2003) 993–1022
11. Wei, X., Sun, J., Wang, X.: Dynamic mixture models for multiple time series. In: IJCAI'07, Morgan Kaufmann Publishers Inc. (2007) 2909–2914
12. Iwata, T., Yamada, T., Sakurai, Y., Ueda, N.: Online multiscale dynamic topic models. In: KDD '10, ACM (2010) 663–672
13. Blei, D.M., Lafferty, J.D.: Dynamic topic models. In: International conference on Machine learning. ICML '06, ACM (2006) 113–120
14. Lin, Y.R., Chi, Y., Zhu, S., Sundaram, H., Tseng, B.L.: Facetnet: a framework for analyzing communities and their evolutions in dynamic networks. In: WWW '08, ACM (2008) 685–694
15. AlSumait, L., Barbará, D., Domeniconi, C.: On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In: ICDM '08, IEEE Computer Society (2008) 3–12
16. He, Y., Lin, C., Gao, W., Wong, K.F.: Dynamic joint sentiment-topic model. ACM Transactions on Intelligent Systems and Technology (2013)
17. Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA (2006)
18. Velcin, J., et al.: Investigating the image of entities in social media: Dataset design and first results. In: LREC '14, ELRA (2014)