Front. Comput. Sci., 2012, 6(1): 88–101
DOI 10.1007/s11704-011-1167-7

RESEARCH ARTICLE

Enriching short text representation in microblog for clustering

Jiliang TANG (✉), Xufei WANG, Huiji GAO, Xia HU, Huan LIU

Computer Science & Engineering, Arizona State University, Tempe, AZ 85281, USA

Abstract Social media websites allow users to exchange short texts such as tweets via microblogs and user status in friendship networks. Their limited length, pervasive abbreviations, and coined acronyms and words exacerbate the problems of synonymy and polysemy, and bring about new challenges to data mining applications such as text clustering and classification. To address these issues, we dissect some potential causes and devise an efficient approach that enriches data representation by employing machine translation to increase the number of features from different languages. Then we propose a novel framework which performs multi-language knowledge integration and feature reduction simultaneously through matrix factorization techniques. The proposed approach is evaluated extensively in terms of effectiveness on two social media datasets from Facebook and Twitter. With its significant performance improvement, we further investigate potential factors that contribute to the improved performance.

Keywords short texts, text representation, multi-language knowledge, matrix factorization, social media

1 Introduction

Social media allows users to post short texts. Facebook status length is limited to 420 characters. Twitter limits the length of each Tweet to 140 characters. A personal status message on Windows Live Messenger is restricted to 128 characters[1]. Most categories in Yahoo! Answers has an average post length of less than 500 characters [1].

These short texts pose new challenges to traditional text mining tasks, e.g., clustering, classification, etc. First, short texts often do not provide sufficient statistical information for effective similarity measures (*short docs* problem). Second, abbreviations are widely used and new words are created incessantly (*rampant abbreviations* problem). These problems also exacerbate the problems of *synonymy* and *polysemy*. The former is the problem with distinct words of the same meaning, and the latter is the problem of the same word with different meanings depending on context. A basic representation of a document is *bag of words* in which a document is represented as a vector of words whose entries are non-zero if the corresponding terms appear in the document. Weighting schemes such as *term frequent-inverse document frequency* (tf-idf) are used in text mining to evaluate how important a word is to a document in a text corpus. It is a simple and efficient representation, however, it omits some contextual information such as phrases and sequential patterns.

Researchers made extensive efforts to enrich the short texts representation by exploiting external resources such as WordNet[2] [2], MeSH[3] [3], Wikipedia [4], and the Open Directory Project (ODP) [5]. These improvements involve sophisticated natural language processing for semantic and syntactic analysis with complex representations. Following the same spirit of employing accumulated knowledge, we question whether we could retain the simplicity of the representation while mitigating the four problems.

Received September 31, 2011; accepted November 7, 2011

E-mail: Jiliang.Tang@asu.edu

[1] http://reface.me/status-updates/whats-the-maximum-length-of-a-facebook-status-update/
[2] http://wordnet.princeton.edu
[3] http://www.nlm.nih.gov/mesh

The decades of research on machine translation produced powerful machine translators such as Google Translate[4] and Yahoo! Babel Fish[5]. The idea of utilizing multiple languages to enrich the representation of short texts is based on the following three observations: 1) Multiple words that are synonyms in one language may be translated into unique terms in another language. As shown in Table 1, English terms such as "firm" and "company" are mapped to "entreprise", "notebook" and "laptop" are translated into "ordinateur portable" in French. 2) Contextual information is utilized during the translation from one language to another. For instance, English terms "saw" and "notebook" that have multiple meanings under different contexts are correctly addressed by Google Translate. Thus, word sense disambiguation (WSD) based on context information is naturally involved when documents are translated. 3) Statistical machine translation based on large-scale corpus is capable of dealing with abbreviations and new words effectively to some extent. For example, English words "lab" and "laboratory", "Abbr" and "Abbreviation" are actually equivalent in French.

Table 1 Google translate: some illustrative examples

	English	French
Synonymy	firm	entreprise
	company	entreprise
	notebook	ordinateur portable
	laptop	ordinateur portable
Polysemy	I cut the wood with the **saw**	Je coupe le bois *à* **la scie**
	I **saw** my mother in the park	J'ai **vu** ma mère dans le parc
	I write some words in the **notebook**	Je vous *é*cris quelques mots dans le **carnet**
	My **notebook** is connected to the Internet	Mon **ordinateur portable** est connect *é à* Internet
Abbreviation	lab	laboratoire
	laboratory	laboratoire
	Abbr	Abr*é*viation
	Abbreviation	Abr*é*viation

In this paper, we mainly focus on whether integrating multi-language knowledge can improve the clustering performance for short texts in social media, then study how to integrate knowledge from multiple languages effectively. Our contributions are summarized below:

- Alleviating the four problems for short texts to some extent by taking advantage of the great success of statistical machine translation;
- Enriching the short texts representation with additional knowledge from other languages;
- Proposing an effective framework to integrate knowledge from multiple languages;
- Discovering key factors that contribute to performance improvements.

The rest of this paper is organized as follows. The research problem is formally stated in Section 2. The text enrichment via knowledge from multi-language is detailed in Section 3. Experimental designs and findings are presented in Section 4. Section 5 summarizes the relevant work, and Section 6 concludes the proposed work and our future work.

2 Problem statement

Our research aims to effectively represent short documents such as tweets for clustering by leveraging the power of machine translation while retaining the simple representation of *a bag of words*. In order to address the four problems (short docs, abbreviations, synonymy, and polysemy), we expand a short document by adding its translated counterparts. Machine translation often utilizes contextual information so it can help solve the last three problems to some extent. Actually, as well as the four problems mentioned above, social media data also has other problems such as misspelling and weird grammar. We leave these problems as future work.

Let $T = \{t_1, t_2, \ldots, t_m\}$ be the language set where m is the number of languages considered by the proposed approach. In this paper, we assume that t_1 is the original language while t_2 to t_m are the target languages. Let $D_1 = \{d_{(1,1)}, d_{(1,2)}, \ldots, d_{(1,n)}\}$ be the short text corpus in the original language where n is the number of texts in D_1. $W_1 = \{w_{(1,1)}, w_{(1,2)}, \ldots, w_{(1,m_1)}\}$ denotes the vocabulary of D_1, where m_1 is the number of unique words in D_1.

A short text $d_{(1,j)}$ ($j \in [1,n]$) from D_1 will be translated by the languages t_2 to t_m into $d_{(2,j)}$ to $d_{(m,j)}$. Let $D_i = \{d_{(i,1)}, d_{(i,2)}, \ldots, d_{(i,n)}\}$ be the corpus that is translated from D_1 by t_i. $W_i = \{w_{(i,1)}, w_{(i,2)}, \ldots, w_{(i,m_i)}\}$ is the vocabulary of the D_i where m_i is the number of unique words in D_i.

Let $L = \{L_1, L_2, \ldots, L_m\}$ be the set of term-document matrices where $L_i \in \mathbb{R}^{m_i \times n}$ is the term-document matrix defined on D_i and the vocabulary W_i. For dataset D_i, the weight of the kth word $w_{(i,k)}$ in the jth text $d_{(i,j)}$ is calculated by tf-idf as

[4] http://translate.google.com
[5] http://babelfish.yahoo.com

follows:

$$L_i(k, j) = tf_{d_{(i,j)}}(w_{(i,k)}) \times idf(w_{(i,k)}), \quad (1)$$

where $tf_{d_{(i,j)}}(w_{(i,k)})$ denotes the frequency of $w_{(i,k)}$ in $d_{(i,j)}$ and $idf(w_{(i,k)})$ presents the inverted document frequency of $w_{(i,k)}$ in D_i.

Using the notations and definitions defined above, our enriching short texts representation for clustering can be stated as follows:

Given the set of languages T and the original short document set D_1, we first attempt to construct the term-document matrix set L through machine translators (i.e., Google Translate). Then, we obtain an enriched text representation by integrating multi-language knowledge. The final clusters are identified by applying traditional clustering methods to the enriched representation.

3 Enriching text representation

The scheme of our proposed method for enriching text representation for clustering is demonstrated in Fig. 1. We first translate the original texts from t_1 into other languages t_i ($i \in [2, m]$) by machine translators. Thus, we obtain D_1, $D_2, \ldots, D_m$ in different languages. For each D_i, tf-idf weighting scheme is applied to obtain the corresponding term-document matrix L_i. Next we examine the potential problems of an over-simplified data integration approach.

Intuitively, adding vocabularies W_i ($i \in [2, m]$) from languages t_2 to t_m, we can expand the original vocabulary W_1 to $W = \{W_1, W_2, \ldots, W_m\}$. Thus, the term-document matrix L' becomes $(L_1, L_2, \ldots, L_m)$ and $L' \in \mathbb{R}^{(\sum_{i=1}^{m} m_i) \times n}$. However, there are two problems with this expansion: 1) The dimension of each short text is increased from m_1 to $\sum_{i=1}^{m} m_i$ which makes the expanded term-document matrix L' even more sparse. 2) Machine translation may introduce noise. To avoid these two problems, we propose an effective integration framework through matrix factorization techniques.

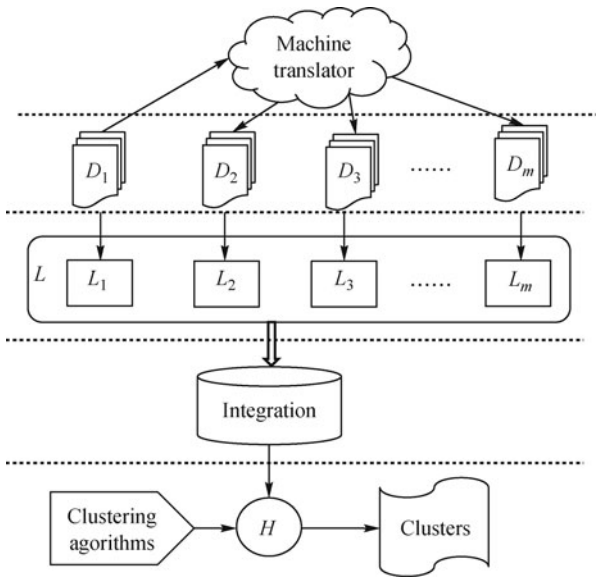


Fig. 1 Framework of integrating multi-language knowledge for short text clustering

3.1 Multi-language knowledge integration framework

Our integration framework is illustrated in Fig. 2. We assume that given the reduced representation H_i of L_i, L_i is independent on $\{L_1, L_2, \ldots, L_{i-1}, L_{i+1}, \ldots, L_m\}$. H_i can be obtained through matrix factorization techniques.

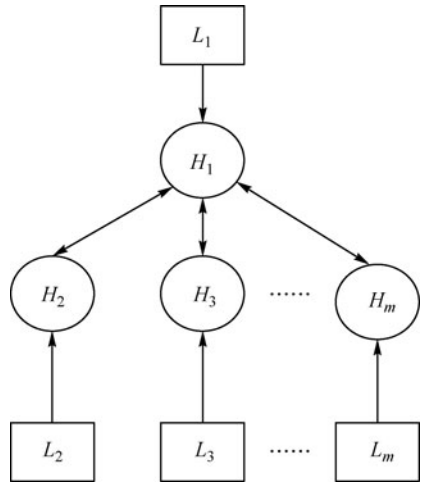


Fig. 2 The framework for integration multi-language knowledge

In our application, matrix factorization techniques map both terms and short texts to a joint latent factor space of dimensionality K. When ignoring coupling between H_i, it can be obtained by solving the following optimization problem.

$$\min_{U_i \geqslant 0, H_i \geqslant 0} \|L_i - U_i H_i\|_F^2, \quad (2)$$

where $\|\cdot\|_F$ denotes Frobenius norm of a matrix. Matrices $U_i \in \mathbb{R}^{m_i \times K}$ and $H_i \in \mathbb{R}^{K \times n}$ are the reduced representations for terms and documents respectively in the K dimension joint latent space. Due to many text clustering algorithms such as LDA [6], PLSI [7], and NMF [8] just accepting nonnegative matrices as their inputs, we further add the nonnegative constraints on U_i and H_i.

Since all the H_i ($i \in [1, m]$) are different views for the n short texts in the K dimensional latent space, we assume that the different views in the K dimension latent space, H_i from language L_i ($i \in [2, m]$), should be close to H_1 from the original language L_1. With this assumption, we minimize the dis-

tance between H_i ($i \in [2, m]$) and H_1 as shown in Eq. (3),

$$\min \sum_{i=2}^{m} \|H_i - H_1\|_F^2. \tag{3}$$

$\mathcal{F}(U_1, \ldots, U_m; H_1, \ldots, H_m)$ is defined as in Eq. (4), which is a combination of Eqs. (2) and (3).

$$\mathcal{F}(U_1, \ldots, U_m; H_1, \ldots, H_m) = \sum_{i=1}^{m} \|L_i - U_i H_i\|_F^2 + \sum_{i=2}^{m} \gamma_i \|H_i - H_1\|_F^2. \tag{4}$$

Then by solving the following optimization problem, we can obtain the reduced and enriched representation.

$$\begin{aligned} &\min \mathcal{F}(U_1, \ldots, U_m; H_1, \ldots, H_m), \\ &\text{s.t.} \quad U_i \geqslant 0, H_i \geqslant 0, i \in [1, m]. \end{aligned} \tag{5}$$

Our framework aims to integrate multi-language knowledge while removing noise introduced by machine translation. In $\mathcal{F}(U_1, \ldots, U_m; H_1, \ldots, H_m)$, $\gamma_i, i \in [2, m]$ is used to control the contributions of these two parts. When γ_i is small, a big weight will be put on multi-language integration; while a big value of γ_i indicates H_i should be very closed to H_1, i.e., removing noise.

3.2 Optimization method for the integration framework

The formulation in Eq. (5) performs integration of multi-language knowledge and dimension reduction simultaneously. There are $2m$ coupling components in $\mathcal{F}$, and $\mathcal{F}$ is not concave. Thus it is hard to find a global solution for the joint optimization problem. However, if we fix $2m - 1$ components in $\mathcal{F}$, the resulting optimization problem for the remaining component is concave. By computing these $2m$ components alternatively, we can find an optimal solution. Since in this schema, each component is optimized individually, the solution is a locally minimal solution for Eq. (5). The projected gradient method is adopted in our implementation. In the $(k+1)$th iteration, U_i^{k+1} and H_i^{k+1} are updated as follows:

$$\begin{aligned} U_i^{k+1} &= \max(0, U_i^k - \alpha_k \nabla_{U_i} \mathcal{F}), \\ H_i^{k+1} &= \max(0, H_i^k - \beta_k \nabla_{H_i} \mathcal{F}), \end{aligned} \tag{6}$$

where α_k and β_k are the step sizes. Variants of projected methods differ on selecting the step sizes and we consider a simple and effective one called the Goldstein conditions. For the function $f(x) : \mathbb{R}^n \to \mathbb{R}$, Algorithm 1 can be used to search α_k satisfied Goldstein conditions.

Algorithm 1 Searching α_k with Goldstein condition

Given $0 < c < \frac{1}{2}, \rho \in (0, 1)$
Choose $\hat{\alpha} > 0$, Set $\alpha \leftarrow \hat{\alpha}$
Set $p_k = x_{k+1} - x_k$
repeat
| $\alpha \leftarrow \rho\alpha$
until $f(x_k) + (1-c)\alpha \nabla f_k^{\mathrm{T}} p_k \leqslant f(x_{k+1}) \leqslant f(x_k) + c\alpha \nabla f_k^{\mathrm{T}} p_k$;
Set $\alpha_k = \alpha$

We define $f(U_i)$ from $\mathcal{F}$ by fixing all components except U_i and $g(H_i)$ from $\mathcal{F}$ by fixing all components except H_i.

$$\begin{aligned} f(U_i) &= \|L_i - U_i H_i\|_F^2 + C_1, \\ g(H_i) &= \begin{cases} \|L_i - U_i H_i\|_F^2 + \sum_{i=2}^{m} \gamma_i \|H_i - H_1\|_F^2 + C_2, & i = 1; \\ \|L_i - U_i H_i\|_F^2 + \gamma_i \|H_i - H_1\|_F^2 + C_3, & i \in [2, m], \end{cases} \end{aligned} \tag{7}$$

where C_1, C_2, and C_3 are constants. Then $\nabla_{U_i}(\mathcal{F})$ can be easily derived based on $f(U_i)$

$$\nabla_{U_i}(\mathcal{F}) = \nabla_f = -L_i H_i^{\mathrm{T}} + U_i H_i H_i^{\mathrm{T}}.$$

And ∇_{H_i} can be easily gotten from $g(H_i)$ as follows:

$$\begin{aligned} \nabla_{H_i}(\mathcal{F}) &= \nabla_g \\ &= \begin{cases} -U_i^{\mathrm{T}} L_i + U_i^{\mathrm{T}} U_i H_i + \sum_{i=2}^{m} (\gamma_i (H_1 - H_i)), i = 1; \\ -U_i^{\mathrm{T}} L_i + U_i^{\mathrm{T}} U_i H_i + \gamma_i (H_i - H_1), \ i \in [2, m]. \end{cases} \end{aligned} \tag{8}$$

Note that the solution of U_i and H_i, where $i \in [1, m]$, is not unique. Then we have the following theorem:

Theorem 1 Let $U_i, H_i, 1 \leqslant i \leqslant m$ be a valid solution of Eq. (5), then $\tilde{U}_i$ and $\tilde{H}_i$ as defined below is also a valid solution with the same objective value.

$$\begin{aligned} \tilde{U}_i &= U_i Q^{\mathrm{T}}, \\ \tilde{H}_i &= Q H_i, \end{aligned} \tag{9}$$

where $QQ^{\mathrm{T}} = Q^{\mathrm{T}} Q = I_K$ and $Q \in \mathbb{R}^{K \times K}$.

It suffices to show that for each U_i and H_i, two components in $\mathcal{F}(U_1, \ldots, U_m; H_1, \ldots, H_m)$, i.e., $\|L_i - U_i H_i\|_F^2$ and $\|H_i - H_1\|_F^2$, do not change. We can easily prove that the first component does not change by $\tilde{U}_i \tilde{H}_i = U_i Q^{\mathrm{T}} Q H_i = U_i H_i$. For the second component, set $\tilde{H} = H_i - H_1$, then,

$$\begin{aligned} \|QH_i - QH_1\|_F^2 &= \|Q\tilde{H}\|_F^2, \\ &= \mathrm{tr}(\tilde{H}^{\mathrm{T}} Q^{\mathrm{T}} Q \tilde{H}), \\ &= \mathrm{tr}(\tilde{H}^{\mathrm{T}} \tilde{H}), \\ &= \|H_i - H_1\|_F^2, \end{aligned} \tag{10}$$

which completes the proof.

H_1 is the reduced representation, which integrates the knowledge from L_1 to L_m. We seek a unique solution by applying a normalization to each column of H_1. Then our optimization algorithm for the integration framework is illustrated in Algorithm 2. Finally traditional clustering methods can be used to identify clusters based on H_1.

Algorithm 2 An optimization-based integration framework

Initialize U_i and H_i, $i \in [1, m]$
for $k = 1, 2, \ldots$ **do**
  Using algorithm 1 to find the search steps α_k and β_k
  for $i = 1 \rightarrow m$ **do**
    $U_i^{k+1} = \max(0, U_i^k - \alpha_k \nabla_{U_i} \mathcal{F})$
    $H_i^{k+1} = \max(0, H_i^k - \beta_k \nabla_{H_i} \mathcal{F})$
Normalize H_1

3.3 Time complexity analysis

When using Algorithm 2, we must maintain the gradient ∇_{U_i} and ∇_{H_i}. Following the discussion in [9], we should calculate ∇_{U_i} by $U_i(H_i H_i^{\mathrm{T}}) - L_i H_i^{\mathrm{T}}$ and then the time complexity is $O(\mu_i K + nK^2)$, where μ_i is the number of nonzero entities in L_i. Similarly, the time complexity of ∇_{H_i} is $O(\mu_i K + m_i K^2)$. For the short text dataset, the term-document matrix L_i is very sparse. Thus it is not difficult to verify that $\mu_i = O(n)$.

Another main computational task for iteration k is to find step sizes α_k and β_k such that the Goldstein conditions are satisfied. The major operation in Algorithm 1 is $\nabla f_k^{\mathrm{T}} p_k$. Following the above analysis, the computational cost is $O(tnm_iK)$ where t is the number of repetitions in Algorithm 1.

When considering the total number of languages m, the time complexity for integrating m languages is:

$$\sum_{i=1}^{m} O(nK + nK^2 + m_i K^2 + tnm_i K). \tag{11}$$

For the language L_i, the size of vocabulary m_i is almost constant as the number of short texts increases. Considering $K \ll n$, theoretically, the computation time is almost linear to the number of short texts n and the number of integrated languages m.

3.4 Connection to nonnegative matrix factorization

In this subsection, we show the close connection between the proposed formulation and nonegative matrix factorization (NMF). Then we have the following theorem:

Theorem 2 When the components $\{U_1, \ldots, U_{i-1}, U_{i+1}, \ldots, U_m, H_1, \ldots, H_{i-1}, H_{i+1}, \ldots, H_m\}$ in $\mathcal{F}$ are fixed, then the proposed formulation is equal to nonnegtive matrix factorization.

$$\min_{\mathcal{W} \geqslant 0, H \geqslant 0} \|\mathcal{L} - \mathcal{W}H\|_F^2. \tag{12}$$

It suffices to show that this theorem is approval in the next two cases, i.e., $\{U_1, H_1\}$ and $\{U_i, H_i\ (i \in [2, m])\}$.

When other components are fixed, U_1 and H_1 can be obtained by solving the following optimization problem:

$$\|L_1 - U_1 H_1\|_F^2 + \sum_{i=2}^{m} \gamma_i \|H_i - H_1\|_F^2. \tag{13}$$

In this case, set $\mathcal{L}$, $\mathcal{W}$, and H as follows:

$$\begin{aligned} \mathcal{L} &= (L_1^{\mathrm{T}}, \sqrt{\gamma_2} H_2^{\mathrm{T}}, \ldots, \sqrt{\gamma_m} H_m^{\mathrm{T}})^{\mathrm{T}}, \\ \mathcal{W} &= (U_1^{\mathrm{T}}, \sqrt{\gamma_2} I_K, \ldots, \sqrt{\gamma_m} I_K)^{\mathrm{T}}, \\ H &= H_1, \end{aligned} \tag{14}$$

then Eq. (13) can be converted into Eq. (12).

In the other case, U_i and H_i can be obtained through Eq. (15).

$$\|L_i - U_i H_i\|_F^2 + \gamma_i \|H_i - H_1\|_F^2. \tag{15}$$

By setting $\mathcal{L}$, $\mathcal{W}$, and H as in Eq. (16), Eq. (15) can be converted into Eq. (12),

$$\begin{aligned} \mathcal{L} &= (L_i^{\mathrm{T}}, \sqrt{\gamma_i} H_1^{\mathrm{T}})^{\mathrm{T}}, \\ \mathcal{W} &= (U_i^{\mathrm{T}}, \sqrt{\gamma_i} I_K)^{\mathrm{T}}, \\ H &= H_i, \end{aligned} \tag{16}$$

which completes the proof.

Through Theorem 2, we find another way to solve Eq. (5) based on NMF as shown in Algorithm 3.

Algorithm 3 Integration framework based on NMF

for $k = 1, 2, \ldots$ **do**
  Construct $\mathcal{L}$, $\mathcal{W}$, and H through Eq. (14)
  $(U_1, H_1) = NMF(\mathcal{L}, \mathcal{W}, H)$
  for $i = 2, \ldots, m$ **do**
    Construct $\mathcal{L}$, $\mathcal{W}$, and H through Eq. (16)
    $(U_i, H_i) = NMF(\mathcal{L}, \mathcal{W}, H)$
Normalize H_1

4 Empirical evaluation

Text clustering is an important research topic with many practical applications in information retrieval [10] and data mining [11]. Short texts pose new challenges for text clustering. In this section, we aim to answer two questions: 1) Can

multi-language translation help enrich the short texts and improve the performance of short text clustering? 2) Does integrating more languages help? Via varied experiments, we endeavor to figure out the potential causes for improved clustering performance. Finally we present the scalability results of our method.

4.1 Datasets

Two social media datasets (from Facebook and Twitter) and five widely used languages are used in the experiments. The original language of these data sets is English (L1) and then they are translated into four other languages: French (L2), Italian (L3), German (L4), and Spanish (L5). Next we briefly describe the datasets.

For both Facebook and Twitter datasets, we construct a ground truth by selecting 30 topics from Google Trends[6], and retrieve the most relevant personal status or tweets via their APIs.

The topics used to construct Facebook and Twitter datasets are selected from Google Trends. The most popular 30 topics in the last two years are selected and presented in Table 2. The topics are used as queries to Facebook and Twitter, respectively. The queries cover multiple categories including sports (e.g., NFL, New York Giants, Pro Bowl 2011), public figures (e.g., Victoria Beckham, Jerry Herman), movies (e.g., The Dark Night, Total Eclipse), events (e.g., Black Friday), etc. In the experimental evaluations, the selected topics are treated as class labels for the retrieved short text messages.

Table 2 The selected hot topics in two datasets

Topics		
NFL	Family Watch Dog	Victoria Beckham
Eyedea	New York Giants	Diddy Dirty Money
Green Bay	Sidney Poitier	The Dark Knight
Black Friday	Amazing Grace	Fox News Channel
Bloom Box	Aretha Franklin	Sugarloaf Mountain
Bill T Jones	Anjelah Johnson	Teddy Pendergrass
Total Eclipse	Russian National Anthem	
Merle Haggard	Giants Stadium Demolition	
Jared Allen	Sue Sylvester Vogue	
Herman Cain	National Economic Council	
Jerry Herman	Kennedy Center Honors	
Pro Bowl 2011	West Memphis Three	

Facebook[7] is a friendship network where user can interact with their friends. As of January 2011, Facebook has more than 600 million active users. It allows user to post status message (i.e., "What is on your mind?"). In total 3578 status updates are obtained and the number of clusters is set to 30 in the following experiments.

Twitter[8] is a microblogging website. It attracts 190 million visitors per month and generating 65 million tweets a day[9]. A tweet is a short message with a length limit of 140 characters. We retrieve the top 100 tweets for each topic, a total of 2430 tweets. The number of clusters is set to 30 in the following experiments.

The statistics of the datasets in English are presented in Table 3. Both datasets contain very short texts; each text has, on average, less than 25 words. For both datasets, we use Google Translate to obtain four other languages for the short texts.

Table 3 Statistics of the datasets

Dataset	Facebook	Twitter
Number of Docs	3578	2430
Number of Classes	30	30
Vocabulary size	10502	7680
Avg. terms	22.72	17.43

4.2 Clustering methods and evaluation metrics

The proposed integration of multi-language knowledge is independent of the concrete clustering methods. So any traditional clustering algorithms can be used in our experiments. In our work, k-means [12] and LDA [6] are adopted. Since both clustering algorithms often converge to a local minima, we repeat the experiments 10 times and report the average performance and standard deviations. The parameters of our model are determined through cross-validation.

In our work, the clustering quality is evaluated by two metrics, accuracy (ACC) and normalized mutual information (NMI). Denoting $l(c_i)$ as the label of cluster c_i, $l(d_j)$ as the predicted label of the jth document, the accuracy is defined as follows,

$$ACC = \frac{1}{n}\sum_{i=1}^{K}\sum_{j=1}^{n}\delta(l(c_i), l(d_j)), \tag{17}$$

where $\delta(x, y)$ is the delta function that its value is 1 if $x = y$ and 0 otherwise.

Given two clusterings C and C', the mutual information

[6] http://www.google.com/trends
[7] http://www.facebook.com
[8] http://twitter.com
[9] http://techcrunch.com/2010/06/08/twitter-190-million-users/

$MI(C, C')$ is defined as

$$MI(C, C') = \sum_{c_i \in C, c'_j \in C'} p(c_i, c'_j) \log_2 \frac{p(c_i, c'_j)}{p(c_i)p(c'_j)}, \quad (18)$$

and the NMI is defined by

$$NMI(C, C') = \frac{MI(C, C')}{\max(H(C), H(C'))}, \quad (19)$$

where $H(C)$ and $H(C')$ represent the entropies of clusterings C and C', respectively. Larger NMI values represent better clustering qualities.

4.3 Determining latent dimensions

For Facebook and Twitter datasets, we study the correlation between the dimension K of the joint latent space and the clustering performance when five languages are integrated. The results are presented in Figs. 3 and 4. Both figures show similar patterns that the clustering performance varies (improves, reaches its peak, and then deteriorates) with the increment of latent dimensions. When too few dimensions are selected, we lose too much intrinsic information and when too many dimensions are chosen, noise is retained. Thus, this behavior can be used to determine the number of dimensions. The latent dimensions for Facebook and Twitter datasets are approximately chosen at those points with the highest performance. Thus we choose $K = 80$ on Twitter dataset for both k-means and LDA and set $K = 100$ for kmeans and LDA on the Facebook dataset.

4.4 Effect of external knowledge

To answer the question of whether or not multi-language translation can help enrich the short texts and improve the performance of short text clustering, the text representation is enriched by adding another language knowledge to the studied English datasets and checking if they can improve clustering quality. WordNet is a lexical database for the English language [2]. The synonyms of the terms within the short text messages can be added as extra features. Wikipedia can also be utilized to enrich short text message representation with

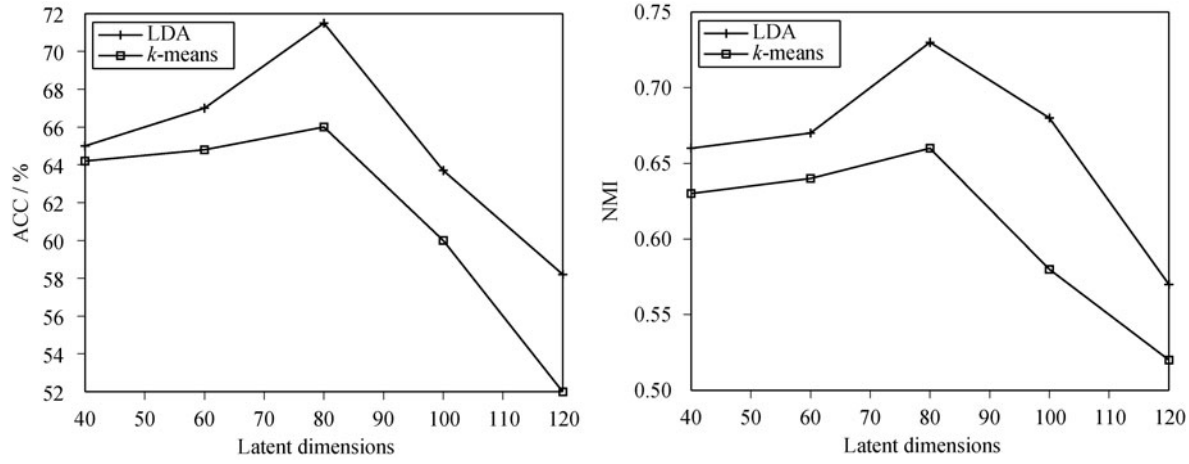


Fig. 3 Accuracy and NMI performance w.r.t. latent dimensions on Twitter

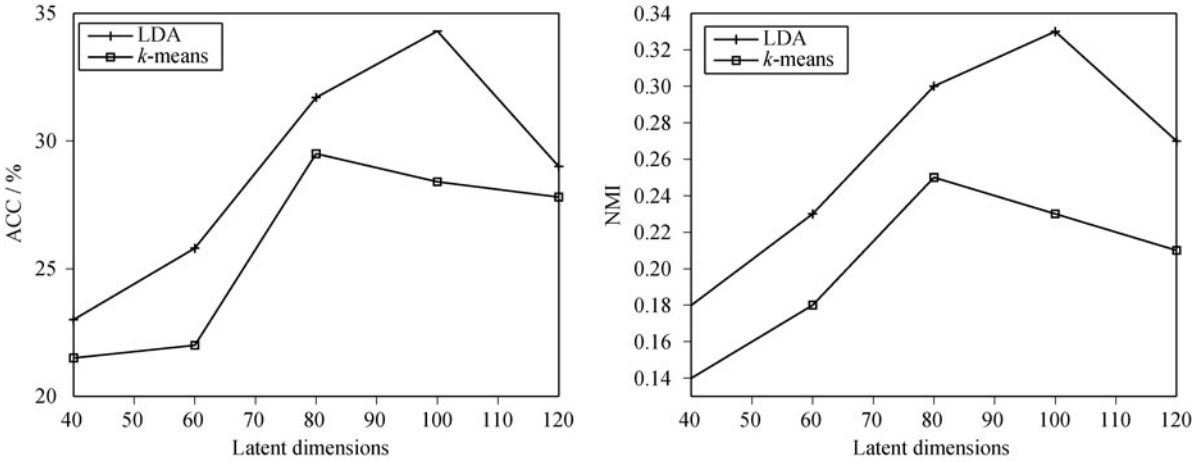


Fig. 4 Accuracy and NMI performance w.r.t. latent dimensions on Facebook

titles and key concepts [13]. Besides, Hu et al. [4] propose to exploit semantically related key concepts from both WordNet and Wikipedia. We also compare different ways of enrichment of short texts and show how they improve the clustering quality.

- L1: English corpus (L1) with tf-idf weighting.
- L1+FR: English corpus (L1) with feature reduction by NMF [14] before clustering.
- L1 + Li ($2 \leqslant i \leqslant 5$): Integrating English (L1) and another language (Li) using our proposed integration framework.
- L1+WN: English corpus (L1) with external knowledge from WordNet [2].
- L1+Wiki: English corpus (L1) with external knowledge from Wikipedia [13].
- L1+WWK: English corpus (L1) with external knowledge from both WordNet and Wikipedia [15].

The clustering performance of the translated languages are shown in Tables 4 and 5. The performance is always reduced after translation. Although translation can solve the four problems of *short doce, rampant abbreviations, synonymy* and *polysemy* to some extent, it also can introduce noise. In our proposed framework, feature reduction techniques are used to address this issue during integration. Also for different translated languages, the performance is different, which may be dependent on the translation quality of the machine translators.

Table 4 k-means performance on translated languages

Dataset	Facebook		Twitter	
	ACC/%	NMI	ACC/%	NMI
L1	15.87±0.53	0.1395±0.0004	45.17±2.05	0.4634±0.0018
L2	14.92±0.72	0.0970±0.0000	41.20±1.97	0.4275±0.0007
L3	15.01±0.86	0.1105±0.0001	41.58±2.31	0.4296±0.0004
L4	15.63±0.23	0.1201±0.0002	43.26±2.01	0.4421±0.0013
L5	15.04±0.93	0.1170±0.0002	39.69±1.89	0.4127±0.0011

Table 5 LDA performance on translated languages

Dataset	Facebook		Twitter	
	ACC/%	NMI	ACC/%	NMI
L1	20.11±1.13	0.2043±0.0004	50.65±2.26	0.5191±0.0024
L2	19.44±1.17	0.1993±0.0001	45.86±1.96	0.4609±0.0011
L3	18.63±1.11	0.1911±0.0001	47.58±2.21	0.4769±0.0018
L4	20.00±1.26	0.1927±0.0003	49.75±2.37	0.4995±0.0025
L5	19.10±1.22	0.1970±0.0004	47.55±1.95	0.4627±0.0014

The performance of LDA and k-means on the reduced representation enriched by two languages are presented in Tables 6 and 7 respectively. We observe that LDA always obtains better performance than kmeans while for standard deviations it seems that k-means is more stable than LDA. From the tables, we can see that performing feature reduction before clustering can significantly improve the performance. Thus it is necessary to perform feature reduction before clustering for social media data due to its noise and sparseness. Also after feature reduction, k-means and LDA are more stable.

Table 6 k-means performance w.r.t. external knowledge

Dataset	Facebook		Twitter	
	ACC/%	NMI	ACC/%	NMI
L1	15.87±0.53	0.1395±0.0004	45.17±2.05	0.4634±0.0018
L1+FR	24.58±0.47	0.1835±0.0007	56.72±1.84	0.5745±0.0010
L1+L2	31.35±0.71	0.2620±0.0002	66.52±1.03	0.6624±0.0007
L1+L3	30.55±0.28	0.2518±0.0001	62.46±0.76	0.6351±0.0010
L1+L4	26.09±0.16	0.2147±0.0001	64.69±1.68	0.6748±0.0015
L1+L5	38.34±0.51	0.3214±0.0001	64.77±1.24	0.6407±0.0012
L1+WN	17.12±0.93	0.1505±0.0003	45.39±1.96	0.4585±0.0018
L1+Wiki	18.65±0.72	0.1598±0.0001	46.83±1.77	0.4937±0.0013
L1+WWK	20.47±0.71	0.1725±0.0001	48.09±1.54	0.5149±0.0007

Table 7 LDA performance w.r.t. external knowledge

Dataset	Facebook		Twitter	
	ACC/%	NMI	ACC/%	NMI
L1	20.11±1.13	0.2043±0.0004	50.65±2.26	0.5191±0.0024
L1+FR	26.78±0.58	0.2762±0.0016	63.31±1.92	0.6478±0.0051
L1+L2	40.74±0.65	0.4295±0.0002	72.14±1.61	0.7835±0.0018
L1+L3	32.65±1.04	0.3470±0.0005	66.46±0.98	0.6991±0.0016
L1+L4	28.09±0.25	0.3086±0.0000	70.45±1.82	0.7671±0.0047
L1+L5	34.14±0.89	0.3592±0.0004	68.07±1.52	0.7359±0.0017
L1+WN	22.47±1.01	0.2054±0.0004	51.85±2.60	0.5249±0.0023
L1+Wiki	24.11±0.98	0.2182±0.0001	53.46±2.13	0.5453±0.0016
L1+WWK	25.37±0.85	0.2221±0.0001	55.66±1.92	0.5711±0.0013

We note that the clustering performance based on the new representation is also improved significantly. On average, we gain 60% and 35% relative improvement in terms of accuracy on Facebook and Twitter datasets respectively. We observe a similar improvement with respect to NMI. Factors that can influence the performance, e.g., the machine translation quality and our integration framework, will be studied in later sections. Another important observation is that different languages affect the performance with varying degrees. For instance, a language that performs well on one dataset will not necessarily perform well on others.

External knowledge such as WordNet and Wikipedia can help to improve the clustering performance to some extent. However, there are several limitations to these methods. First, a dictionary is limited by its vocabulary and phrases, thus, it is hard to deal with rampant abbreviations, acronyms, and coined words for short texts in social me-

dia. Second, involving external knowledge may introduce noise and increase the dimensionality, which can harm performance. Thus feature reduction is necessary when enriching the text representation by external knowledge which also increase the dimensionality and might introduce noises.

4.5 Effect of the number of languages

We attempt to answer the second question in this subsection: does integrating more languages help? Given the four other available datasets with different languages, we evaluate the performance with respect to different combinations with the dataset in the original language. The results by LDA are presented in Table 8 since similar results can be observed from *k*-means. We find that the peak performance is not achieved when all five languages are integrated. For example, we obtain the best performance with respect to accuracy when integrating L1, L3, and L4 on Facebook, and L1, L4, and L5 on Twitter. When all five languages are integrated, the performance drops.

By integrating multiple language knowledge, it may cause a negative impact on clustering performance as the feature space expands. First, the quality of the new generated features (terms) depends on the quality of machine translation. As will be shown later, the translation quality is one of the factors affecting the performance. Second, inconsistency between different languages may exist due to machine translation. Third, it leads to the curse of dimensionality. For example, when we add all five languages for Twitter dataset, the dimension is increased from 7 680 to 50 452. Although effective tools can be used to reduce dimension, it may impair the meaningful terms.

4.6 Key factors for improvement

The above empirical results suggest that our framework for enriching short text representation by integrating multiple languages helps improve the performance of text clustering. We attempt to further discover key factors that contribute to significantly improved performance and the results shown below are based on LDA. We construct three additional experiments.

- Using a dummy translator

We are curious if we can gain clustering performance by simply expanding the dimensionality of the data. So we construct a dummy translator that translates an English word to itself. Thus, through this translation, we do not add any more information into the original corpus, but only double the dimensionality. Integrating the two should not improve the clustering quality. The performance is presented in Fig. 5. "L1+L1" represents integrating two copies of L1 with our integration framework. "MaxL1+Li"and "MinL1+Li" represent the best and worst performance when two languages are integrated. As is expected, compared with "L1+FR", "L1+L1" does not improve the performance, as a matter of fact, it slightly reduces performance. One reason is the doubled dimensionality; this unnecessary doubling in features can do harm to clustering quality. Through our framework, "L1+L1" is reduced to a low dimension, on which the clustering algorithm obtains better performance than on the original representation. Since the impact of translation is fixed, we believe that the improvement is from the

Table 8 LDA performance when integrating multiple language knowledge

Dataset	Facebook		Twitter	
	ACC/%	NMI	ACC/%	NMI
L1	20.11±1.13	0.2043±0.0004	50.65±2.26	0.5191±0.0024
L1+L2	40.74±0.65	0.4295±0.0002	72.14±1.61	0.7835±0.0018
L1+L3	32.65±1.04	0.3470±0.0005	66.46±0.98	0.6991±0.0016
L1+L4	28.09±0.25	0.3086±0.0000	70.45±1.82	0.7671±0.0047
L1+L5	34.14±0.89	0.3592±0.0004	68.07±1.52	0.7359±0.0017
L1+L2+L3	38.23±0.92	0.3885±0.0012	67.17±1.08	0.7132±0.0017
L1+L2+L4	32.65±0.79	0.3372±0.0009	64.01±1.52	0.6927±0.0013
L1+L2+L5	41.45±1.06	0.4383±0.0013	67.54±1.92	0.7112±0.0030
L1+L3+L4	36.72±0.47	0.3726±0.0007	73.67±1.84	0.7900±0.0020
L1+L3+L5	38.34±0.83	0.3902±0.0010	71.53±1.71	0.7642±0.0018
L1+L4+L5	35.19±0.50	0.3599±0.0006	70.98±1.63	0.7517±0.0024
L1+L2+L3+L4	39.98±0.87	0.4063±0.0011	64.53±0.72	0.6692±0.0018
L1+L2+L3+L5	43.34±1.21	0.4501±0.0020	70.17±1.81	0.7146±0.0030
L1+L2+L4+L5	33.06±0.32	0.3599±0.0003	69.67±1.48	0.7001±0.0028
L1+L3+L4+L5	35.83±0.87	0.3700±0.0010	72.92±1.42	0.7444±0.0017
L1+L2+L3+L4+L5	34.57±0.85	0.3698±0.0011	71.18±0.88	0.7297±0.0017

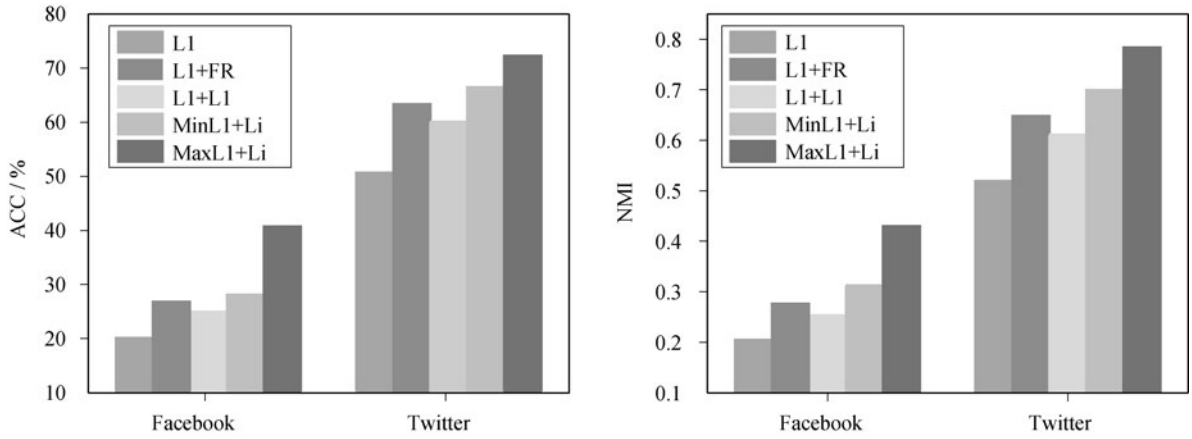


Fig. 5 Impact of machine translator

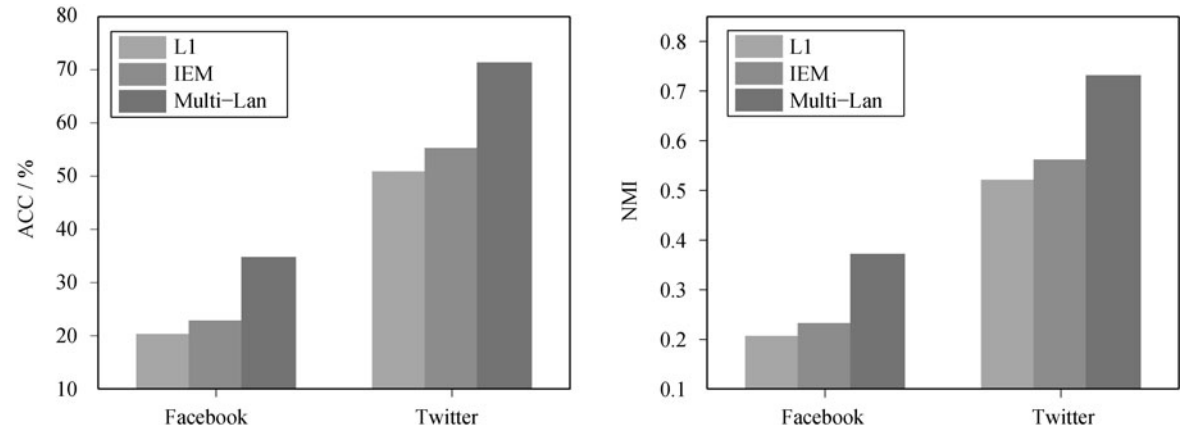


Fig. 6 Impact of integration framework

effectiveness of our integration framework, which is verified in the next subsection.

• Integration framework

Our integration framework can perform multi-language integration and feature reduction simultaneously. As mentioned above, one intuitive way to enrich the text representation is to expand the original dictionary from other languages. We compare our multi-language integration framework (Multi-Lan) with this intuitive enriching method (IEM) and the results are shown in Fig. 6. Even for the intuitive way, the performance is also improved. This part of improvement is purely from multi-language integration. The four problems mentioned in Section 1 are permeating in short texts in social media. We believe that these problems could be partially addressed by Google Translate. Compared to IEM, the performance of Multi-Lan is significantly improved, which supports the notion that our proposed integration framework can address the issues of curse of dimensionality and noise introduced by translator.

• Removing contextual information

A machine translator does not translate a sentence word by word. In other words, it takes into account contextual information. If we translate a short text word by word, it discards the contextual information. We would expect that such a translation would not be able to capture accurate term meanings. We use Google Translate to translate word by word from English to other four languages and the results are shown in Fig. 7, where "TwitterW" means the dataset translated from Twitter word by word and "FacebookW" indicates the dataset translated from Facebook word by word. The performance is degraded in all cases when the contextual information is not considered for the translation of short texts.

• Discussion

Comparing with "L1", "L1+L1" gains 35% and 20% relative improvement in Facebook and Twitter datasets, respectively. Since the impact of translation is fixed, this part of improvement is entirely due to integration framework. We use "MaxL1Li+wo" to represent the best performance when two languages knowledge are integrated in the intuitive way, removing the impact from our framework. "MaxL1Li+wo"

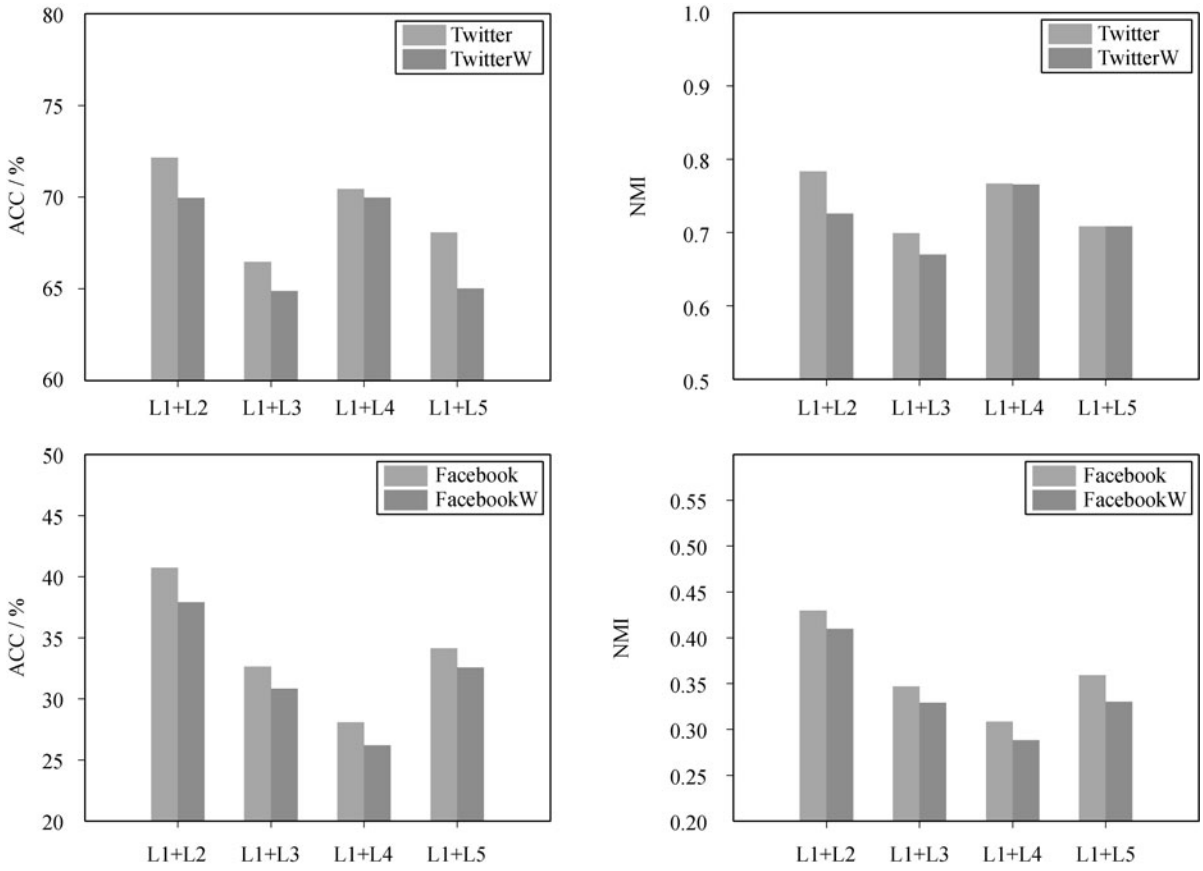


Fig. 7 Impact of contextual information

gains 17% and 13% relative improvement respectively. Since the impact of our proposed integration framework is fixed, this part of improvement is totally from translation. Thus both impacts contribute to the improvement. However, when we consider both impacts, "MaxL1Li" gains 50% and 40% relative improvement respectively, which is much better than the improvement of each individual impact. This supports the effectiveness of our framework, which performs language knowledge integration and feature reduction simultaneously.

4.7 Scalability

Theoretical analysis shows that the time complexity of the proposed method is linear with respect to the number of short texts and the number of languages integrated. We verify it empirically.

First, we fix the number of languages to two to study the relationship between time complexity and the number of short texts. Given a specific ratio (e.g., 50%), we select texts randomly from the whole dataset, and record the time for our method. The process is repeated ten times and the average elapsed time is reported. As shown in Fig. 8(a), the time spent increases linearly when more texts are added. Also the slopes of both lines are very shallow, which means the time complexity increases slowly with respect to the number of texts. The time spent on Facebook is longer because there are more texts in this data set and the average length of texts in this dataset is longer.

We fix the number of texts in both datasets and vary the number of languages to be integrated. The total time for both datasets are presented in Fig. 8(b). Clearly, the total time taken is linear with respect to the number of languages.

5 Related work

Integrating external knowledge to text mining has recently attracted more attention. Based on the types of external resources being used, prior work belongs to one of the three categories: thesaurus, web knowledge, and a combination of both.

Thesaurus or dictionary groups words according to similarity of meaning. WordNet and MeSH are the two dictionaries that have been widely used in text mining. Hotho et al. [2] propose to incorporate synonyms from WordNet into text representation; they show that the extra features can improve text

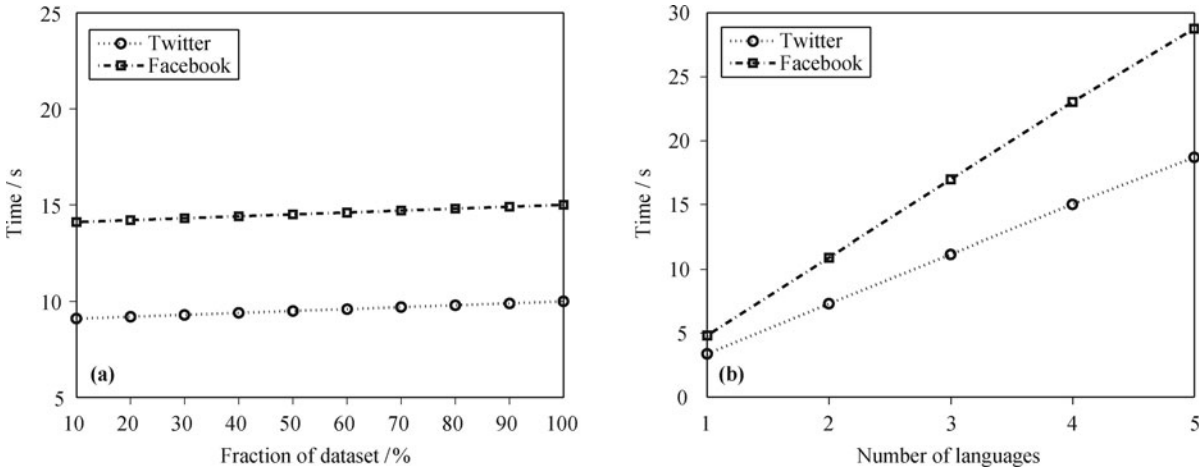


Fig. 8 Scalability w.r.t. the number of text texts and integrated languages

clustering quality. Halkidi, et al. [16] propose to leverage link structure and word similarity based on WordNet for effective web document characterization. Yoo et al. [17] map terms in a document into MeSH concepts through the MeSH thesaurus and find that this strategy can improve the performance of text clustering. However, using a thesaurus should be done with caution. For example, by utilizing WordNet synsets, Dave et al. [11] find that without performing WSD, the performance of clustering decreases.

Web knowledge is deemed a collective wisdom. ODP and Wikipedia are well recognized. Text categorization performance is improved by augmenting the bag-of-words representation with new features from ODP and Wikipedia [18,19]. In another work, Banerjee et al. [13] propose to cluster Google news by incorporating the titles of the top-relevant Wikipedia articles as extra features. On the other hand, adding new features (concepts, titles) could increase the dimension significantly [20].

Combining thesaurus and web knowledge could provide further improvement in text mining tasks in practice. Hu et al. [15] cluster short texts (i.e., Google snippets) by first extracting the important phrases and expanding the feature space by adding semantically close terms or phrases from WordNet and Wikipedia. Kasneci et al. [21] build (and maintain) a taxonomy of entities by taking advantage of the knowledge of WordNet and Wikipedia. Search engines such as TopX employ WordNet and Wikipedia to expand queries, measure similarity, relate concepts, etc. [22].

6 Conclusions

In this work, we propose a novel integration framework which can perform language knowledge integration and feature reduction simultaneously through matrix factorization techniques. Experimental results show promising findings: 1) the proposed approach significantly improves the short texts clustering performance; 2) different languages contribute unevenly to text clustering; 3) having more languages does not necessarily result in better performance; and 4) the proposed method scales linearly with the number of short texts and the number of integrated languages.

This study also suggests some interesting problems for further exploration. As mentioned above, aside from the four main problems, there are many other problems such as misspelling and weird grammar for short texts from social media. Solving these issues before translation might further improve the performance. Since our method is independent of specific tasks, more potential applications such as classification might be employed in the future work.

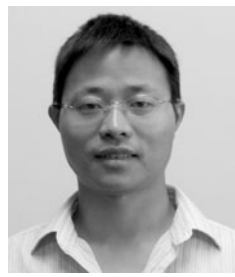
Acknowledgements The work was supported by ONR (N000141010091) and NSF (0812551).

References

1. Adamic L A, Zhang J, Bakshy E, Ackerman M S. Knowledge sharing and yahoo answers: everyone knows something. In: Proceedings of 17th International Conference on World Wide Web. 2008, 665–674
2. Hotho A, Staab S, Stumme G. Wordnet improves text document clustering. In: Proceedings of 2003 SIGIR Semantic Web Workshop. 2003, 541–544
3. Reforgiato Recupero D. A new unsupervised method for document clustering by using WordNet lexical and conceptual relations. Information Retrieval, 2007, 10(6): 563–579
4. Hu J, Fang L, Cao Y, Zeng H J, Li H, Yang Q, Chen Z. Enhancing text clustering by leveraging Wikipedia semantics. In: Proceedings of 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 2008, 179–186

5. Hu X, Zhang X, Lu C, Park E K, Zhou X. Exploiting Wikipedia as external knowledge for document clustering. In: Proceedings of 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2009, 389–396
6. Blei D M, Ng A Y, Jordan M I. Latent Dirichlet allocation. Journal of Machine Learning Research, 2003, 3: 993–1022
7. Hofmann T. Probabilistic latent semantic indexing. In: Proceedings of 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 1999, 50–57
8. Xu W, Liu X, Gong Y. Document clustering based on non-negative matrix factorization. In: Proceedings of 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 2003, 267–273
9. Lin C J. Projected gradient methods for non-negative matrix factorization. Neural Computation, 2007, 19(10): 2756–2779
10. Cutting D R, Pedersen J O, Karger D R, Tukey J W. Scatter/gather: a cluster-based approach to browsing large document collections. In: Proceedings of 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 1992, 318–329
11. Dave K, Lawrence S, Pennock D M. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: Proceedings of 12th International Conference on World Wide Web. 2003, 519–528
12. Steinbach M, Karypis G, Kumar V. A comparison of document clustering techniques. In: Proceedings of 2000 KDD Workshop on Text Mining. 2000, 525–526
13. Banerjee S, Ramanathan K, Gupta A. Clustering short texts using Wikipedia. In: Proceedings of 30th Annual International ACM SIGIR Conference on Research and Development. 2007, 787–788
14. Lee D D, Seung H S. Algorithms for non-negative matrix factorization. In: Proceedings of 2000 Neural Information Processing Systems. 2000, 556–562
15. Hu X, Sun N, Zhang C, Chua T S. Exploiting internal and external semantics for the clustering of short texts using world knowledge. In: Proceedings of 18th ACM Conference on Information and Knowledge Management. 2009, 919–928
16. Halkdi M, Nguyen B, Varlamis I, Vazirgiannis M. THESUS: organizing Web document collections based on link sematics. The VLDB Journal, 2003, 12(4): 320–332
17. Yoo I, Hu X, Song I Y. Integration of semantic-based bipartite graph representation and mutual refinement strategy for biomedical literature clustering. In: Proceedings of 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2006, 791–796
18. Gabrilovich E, Markovitch S. Feature generation for text categorization using world knowledge. In: Proceedings of 19th International Joint Conference on Artificial Intelligence. 2005, 1048–1053
19. Gabrilovich E, Markovitch S. Overcoming the brittleness bottleneck using Wikipedia: enhancing text categorization with encyclopedic knowledge. In: Proceedings of 21st National Conference on Artificial Intelligence, Vol 2. 2006, 1301–1306
20. Fodeh S, Punch B, Tan P N. On ontology-driven document clustering using core semantic features. Knowledge and Information Systems, 2011, 28(2): 395–421
21. Kasneci G, Ramanath M, Suchanek F, Weikum G. The YAGO-NAGA approach to knowledge discovery. ACM SIGMOD Record, 2008, 37(4): 41–47
22. Theobald M, Bast H, Majumdar D, Schenkel R, Weikum G. TopX: efficient and versatile top-*k* query processing for semistructured data. The VLDB Journal, 2008, 17(1): 81–115

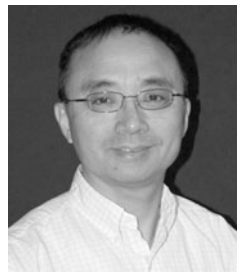
Jiliang Tang is a PhD student in computer science and engineering at Arizona State University. He received his BSc and MSc degrees from Beijing Institute of Technology in 2008 and 2010. His research interests include data mining and machine learning. Specifically, he is interested in social computing and feature selection in social media.



Xufei Wang is a PhD student in computer science and engineering at Arizona State University. He received his Masters degree from Tsinghua University, and Bachelor degree of Science from Zhejiang University, China. His research interests are in social computing and data mining. Specifically, he is interested in mining social media data, social network analysis, mining ego-centric friend structure, tag network, crowdsourcing, etc. He is an IEEE student member.

Huiji Gao is a PhD student in Data Mining and Machine Learning (DMML) Lab at Arizona State University (ASU). He received his BSc and MSc degrees from Beijing University of Posts and Telecommunications, China in 2007 and 2010. His research interests include social computing, data mining, and social media mining, in particular, crowdsourcing and spatial-temporal mining. Contact him at huiji.gao@asu.edu.

Xia Hu is a PhD student of Computer Science and Engineering at Arizona State University. He received his Master and Bachelor degrees from the School of Computer Science and Engineering of Beihang University. His research interests are in text analytics in social media, social network analysis, machine learning, text representation, sentiment analysis, etc. He was awarded an ASU GPSA Travel Grant, Machine Learning Summer School at Purdue Fellowship, SDM Doctoral Student Forum Fellowship, and various Student Travel Awards and Scholarships from ASU, NUS, and BUAA.



Dr. Huan Liu is a professor of Computer Science and Engineering at Arizona State University. He obtained his PhD degree in Computer Science at the University of Southern California and BEng degree in Computer Science and Electrical Engineering at Shanghai Jiao Tong University. His research focus is centered on investigating problems that arise in many real-world applications with high-dimensional data of disparate forms such as analyzing social media, group interaction and modeling, feature selection, and text/web mining. His well-cited publications include books, book chapters, encyclopedia entries as well as conference and journal papers.