Topic-Aware Sentiment Prediction for Chinese ConceptNet

Po-Hao Chou*, Chi-Chia Huang†, Richard Tzong-Han Tsai‡ and Jane Yung-jen Hsu§
*†§*Computer Science and Information Engineering, National Taiwan University*
‡*Computer Science and Information Engineering, National Central University*
*Email: { *r02922017, †d01944003, §yjhsu}@csie.ntu.edu.tw ‡thtsai@csie.ncu.edu.tw*

***Abstract*—Sentiment analysis aims to identify the attitudes or emotions behind texts. For many sentiment analysis approaches, sentiment information of terms or phrases plays an important role. However, in Chinese sentiment analysis, the coverage of such information is still limited. To increase the coverage, some methods has been developed to predict sentiments for nodes in Chinese ConceptNet due to its large size and high semantic level nodes. In ConceptNet, the notion of a node is extended from purely lexical terms to include higher-order compound concepts, e.g., 'eat lunch', 'satisfy hunger', so these nodes are called concepts.**

Current approaches aim to assign one sentiment to each concept, but in fact a concept may have different sentiments on different contexts, such as 'scream' and 'sudden' in Chinese. In this paper, our first goal is to extract the hidden contextual information in Chinese ConceptNet and use it to estimate sentiments in different situations for each concept. To achieve this goal, we propose a topic-aware sentiment propagation approach. We apply Latent Dirichlet Allocation to divide Chinese ConceptNet into different topic layers and use sentiment propagation on each topic layer to predict topic-aware sentiments for Chinese concepts. Our another goal is to use the generated topic-aware sentiments of concepts to improve the polarity classification for texts. We combine other co-occurring concepts to identify topics and select sentiments for concepts in texts. Then, experiments conducted on dialogue dataset and microblog posts show the improvement of topic-aware prediction for concepts and texts.

1. Introduction

With the rise of social media services, more and more people share their opinions, feelings, experiences on the web. The roles of users shift from information consumers to information producers. To understand the attitudes and emotions behind these texts, sentiment analysis has become a popular research topic in recent years.

Sentiment dictionaries tell machine how a writer or speaker may feel when using some term or phrase. They are important elements for several sentiment analysis approaches [**?**], [**?**], [**?**], [**?**], but in Chinese, they are relatively scarce or non-public. As the result, it is helpful to collect the sentiment information in Chinese. However, there are two problems in collecting sentiment information. First, its coverage and quality highly affect the performance of sentiment analysis for texts, but collecting the information with high coverage, quality but low cost is challenging. The second problem is that the sentiments of several terms and phrases are context-dependent. How to define context, collect the sentiments on different contexts efficiently and decide which context and sentiment should be used to predict sentiments of texts are challenges.

For the first problem, several approaches [**?**], [**?**], [**?**], [**?**], [**?**], [**?**], [**?**] have used the external knowledge such as WordNet [**?**] and ConceptNet [**?**], [**?**] to build sentiment dictionaries automatically. The relationship information in WordNet and ConceptNet is used to propagate sentiments from some seeds, which had been compiled manually.

ConceptNet is a semantic network which represents knowledge into more computable representations. The nodes in ConceptNet are called *concepts* because the coverage of nodes contains not only lexical terms but also higher-order compound concepts, e.g., 'accomplish goal', 'leave behind'. A directed edge connecting two nodes is called *relation*, and is associated with one of the predefined types of labels to represent the semantic relationships between two *concepts* in real world, e.g., 'CapableOf', 'Causes'. The former *concept*, latter *concept* and their *relation* form an *assertion*, such as "oven UsedFor cook", "eat HasSubevent swallow". Because ConceptNet has large amount of *concepts* (In Chinese part [**?**], [**?**], there are at least 220000 *concepts*, still growing...), and its *concepts* have higher semantic meaning than traditional lexical terms, it is a good foundation to build a larger sentiment dictionary. In this paper, we collect sentiments based on *concepts* and the structure in Chinese ConceptNet.

However, previous propagation approaches in ConceptNet didn't deal with the issue that a *concept*'s neighbors may come from different scenarios. Take Figure 1 for example, previous approaches aggregate the sentiments from neighbors disregarding their different scenarios, and assign the aggregated sentiment to 'do not have to work'. 'do dot have to work' will propagate this sentiment to all its neighbors in the next iteration, which makes sentiments be propagated between different scenarios.

The second problem is that each *concept* should be assigned different sentiments in different scenarios. For example, 'scream' is more possible to be negative when 'pervert' or 'cockroach' appears, but is more likely to be

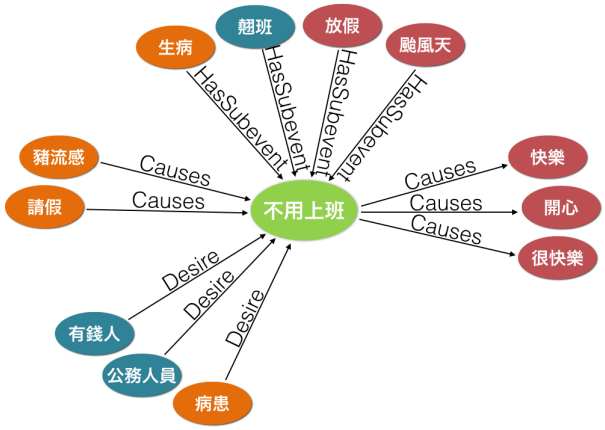


Figure 1. Neighbors of 'do not have to work' come from different scenarios.

positive when 'idol' or 'win' appears. Previous approaches [?], [?], [?] use domain-specific corpus to modify sentiment value according to the corpus. However, hidden contextual information in Chinese ConceptNet is also abundant, like Figure 1. If we could know which scenario an assertion belongs to, a *concept*'s sentiment in a scenario can be determined from its assertions which belongs to the scenario.

To deal with the two problems, this paper develops a topic-aware propagation method for Chinese ConceptNet. We extract hidden contextual information by applying Latent Dirichlet Allocation [?] to Chinese ConceptNet. The information divides Chinese ConceptNet into different topic layers where each topic is a distribution over concepts. Then, sentiment propagation can be performed on each topic layer. This not only avoids propagating sentiments between different scenarios but also generating sentiment on each topic for each concept. Then we present how to apply these topic-aware sentiment values of *concepts* to polarity classification for texts. Finally, two experiments are conducted. The first experiment use part of concepts to test the results of topic-insensitive and topic-aware propagation. It shows the effect of introducing topics to avoid propagation between different scenarios. The second experiment is conducting on the microblog dataset provided by the Chinese Microblog Sentiment Analysis Evaluation (CMSAE) task in the conference on Natural Language Processing & Chinese Computing (NLP & CC) 2013 [1]. The result show that compared with assigning a concept a fixed sentiment value, identifying its topic and select a proper sentiment value performs better.

2. Related Work

2.1. Sentiment Prediction for ConceptNet

Random walk is commonly used to spread values on a graph [?], [?], [?], [?]. It spreads values through synonym and antonym edges. The equation of random walk with restart is as follows:

$$\boldsymbol{s}_{t+1} = (1-\alpha)\boldsymbol{W}\boldsymbol{s}_t + \alpha\boldsymbol{s}_0 \quad (1)$$

where $\boldsymbol{s}_t$ is the values of each node when the t-th iteration. $\boldsymbol{W}$ is a similarity matrix, which can be acquired from sources like Ontology, ConceptNet, corpus, etc. Similarity matrix of a standard random walk is an out-link normalized matrix. α is the restarting weight. Random walk is an iterative process, and after n iteration, each node spreads its value to the neighbors that are n links distant from it.

In previous research [?], [?], random walk is applied to propagate values on ConceptNet. They found that in ConceptNet, performing in-link normalized on similarity matrix is better than performing out-link normalized because out-link normalization will underestimate the influence of concepts with more neighbors. In in-link normalization, each concept's new sentiment value in the $(t+1)$-th iteration is the average of all its neighbors in the t-th iteration.

2.2. Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) [?] is a generative probabilistic model of a corpus, in which documents are represented as random mixtures over latent topics, and each topic is characterized by a distribution over words. The intuition behind LDA is that each document exhibits multiple topics. It exploits word co-occurrence information to capture latent topics in the corpus.

In LDA, each document d is represented by bag of words (so the order of words in a document is not considered), and each document is assumed to be generated by the following (given parameters of Dirichlet distribution $\boldsymbol{\alpha}$ and per-topic word distribution $\boldsymbol{\beta}$):

1) Choose probability coefficients over topics $\boldsymbol{\theta}_d \sim Dir(\boldsymbol{\alpha})$
2) For each of the N word positions $x_{d,n}$:
   a) Choose a topic assignment $z_{d,n} \sim Multinomial(\boldsymbol{\theta})$
   b) Choose a word $w_{d,n}$ conditioned on the chosen topic $z_{d,n}$ and the per-topic word distribution $\boldsymbol{\beta}_{z_{d,n}}$

From the generative process, the graphical model representation of basic LDA is shown in Figure 2. Observing word co-occurrence in each document, LDA infers topic probability coefficients $\boldsymbol{\theta}_d$ for each document and topic assignment $z_{d,n}$ for each word of each document to maximize the likelihood of this corpus. However, the posterior distribution is intractable for exact inference of each $\boldsymbol{theta}_d$ and $z_{d,n}$. Blei *et al.* use variational EM to infer these latent variables and find parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to maximize the log likelihood of given corpus.

Because the words coverage for each document is sparse, they smooth multinomial parameters $\boldsymbol{\beta}$ by generating $\boldsymbol{\beta}_k \sim Dir(\boldsymbol{\eta})$ for each topic, shown in Figure 3. They infer $\boldsymbol{\beta}_k$ by modify the variational model, like inferring $\boldsymbol{\theta}_d$.

1. http://tcci.ccf.org.cn/conference/2013/pages/page04_dg.html

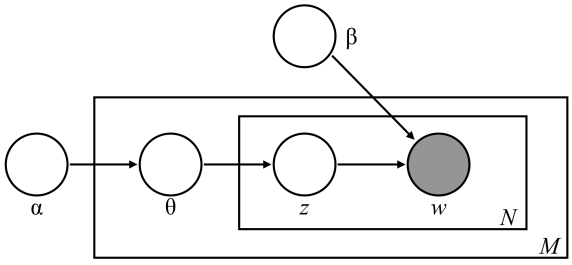


Figure 2. Graphical model representation of LDA.

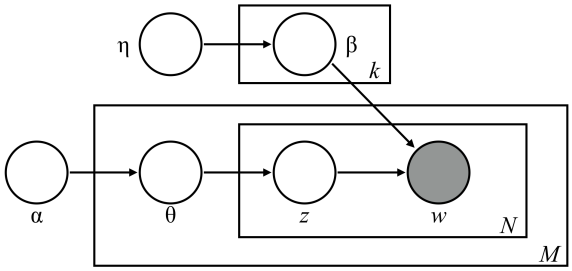


Figure 3. Graphical model representation of smoothed LDA.

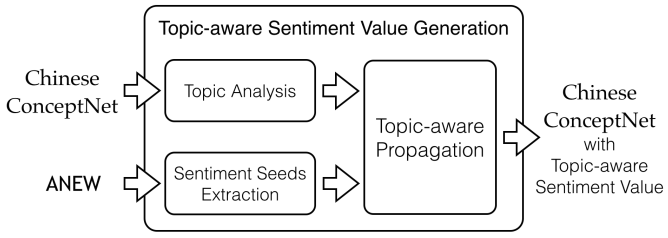


Figure 4. System Architecture.

As the result, LDA infers $\boldsymbol{\theta}_d$ for each document, $z_{d,n}$ for each word of document and $\boldsymbol{\beta}_k$ for each topic. Besides, it estimates $\boldsymbol{\alpha}$ and $\boldsymbol{\eta}$ as the model of this corpus.

3. Topic-Aware Sentiment Value Prediction for Chinese ConceptNet

In topic models such as LDA, each abstract "topic" is characterized by a distribution over words. Such topics are one kind of representation of contexts or scenarios. In this section, we aim to define the topics in Chinese ConceptNet and predict a sentiment value $\in [-1, 1]$ on each topic for each concept. Figure 4 shows the architecture of our system.

3.1. Topic Analysis on ConceptNet

We choose LDA to estimate topics for two reasons: The first one is that LDA allows each document to exhibit multiple topics to different degrees. The second one is that there are corpus parameters after LDA estimation, and we can use these corpus parameters to infer a new document by running the E-step in LDA.

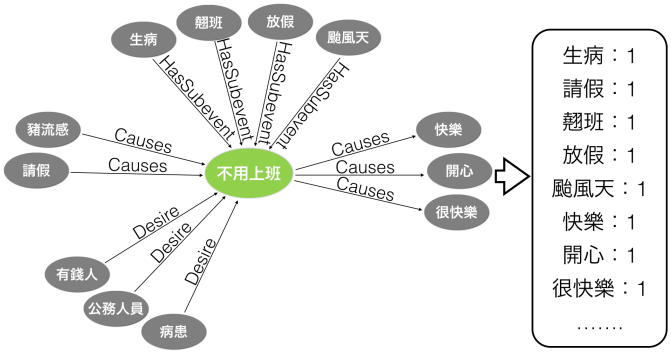


Figure 5. The document generated by 'do not have to work'.

Similar to the assumption of generating a corpus in LDA, we assume that there is a topic distribution for each concept and its assertions are sampled from this distribution. We aim to find which topic each assertion comes from for each concept. Therefore, we try to apply LDA to find such latent information.

First, each concept forms a document using its neighbors such that these neighbors have one co-occurrence observation when LDA estimates. See Figure 5 for illustration, the concept 'do not have to work' generates a document using neighbors as words, with a value indicating how many times the neighbor occurs in assertions.

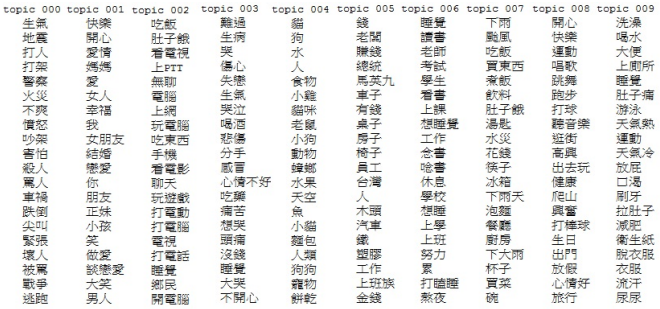
Then we apply LDA to find $\boldsymbol{\theta}_m$ for each document d_m and $z_{m,n}$ for each word $w_{m,n}$ in d_m based on this collection of M documents. Words in each document stand for neighbors of concept, so each resulted $z_{m,n}$ indicates which topicID the neighbor $w_{m,n}$ comes from for concept c_m.

As the result, we can design a topic assignment matrix, $\boldsymbol{T}$ where each entry $t_{i,j}$ is $z_{i,j}$ if j is one of i's neighbors, otherwise $unknown$. That is, we can check $t_{i,j}$ for which topicID i's neighbor j belongs to.

Here we present the result of applying LDA with 10 topics to Chinese ConceptNet. Figure 6 shows the per-topic word distribution, $\boldsymbol{\beta}_k$. Each column shows the top 20 words of a topic.

3.2. Sentiment Seeds Extraction

Sentiment seeds here are concepts whose sentiments are known. Common ways to collect sentiment seeds are compiling manually or using other existing sentiment dictionaries. Although concepts we consider here are in Chinese, we use words in Affective Norms for English Words (ANEW) [**?**] as our seeds. One reason is its good quality: it was compiled manually by a group of Introductory Psychology class students and provides ratings for 1034 English



topic 000	topic 001	topic 002	topic 003	topic 004	topic 005	topic 006	topic 007	topic 008	topic 009
生氣	快樂	吃飯	難過	貓	錢	睡覺	下雨	開心	洗澡
地震	開心	肚子餓	生病	狗	老闆	讀書	颱風	快樂	喝水
打人	愛情	看電視	哭	水	賺錢	老師	吃飯	運動	大便
打架	媽媽	上PTT	傷心	人	總統	考試	買東西	唱歌	上廁所
警察	愛	無聊	失戀	食物	馬英九	學生	煮飯	跳舞	睡覺
火災	女人	電腦	生氣	小雞	車子	看書	飲料	跑步	肚子痛
不爽	幸福	上網	哭泣	貓咪	有錢	上課	肚子餓	打球	游泳
憤怒	我	玩電腦	喝酒	老鼠	桌子	想睡覺	湯匙	聽音樂	天氣熱
吵架	女朋友	吃東西	悲傷	小狗	房子	工作	水災	逛街	運動
害怕	結婚	手機	分手	動物	椅子	念書	花錢	高興	天氣冷
殺人	戀愛	看電影	感冒	蟑螂	員工	唸書	筷子	出去玩	放屁
罵人	你	聊天	心情不好	水果	台灣	休息	冰箱	健康	口渴
車禍	朋友	玩遊戲	吃藥	天空	人	學校	下雨天	爬山	刷牙
跌倒	正妹	打電動	痛苦	魚	木頭	想睡	泡麵	興奮	拉肚子
尖叫	小孩	打電腦	想哭	小貓	汽車	上學	餐廳	打棒球	減肥
緊張	笑	電視	頭痛	麵包	鐵	上班	廚房	生日	衛生紙
壞人	做愛	打電話	沒錢	人類	塑膠	努力	下大雨	出門	脫衣服
被罵	談戀愛	睡覺	睡覺	狗狗	工作	累	杯子	放假	衣服
戰爭	大笑	鄉民	大哭	寵物	上班族	打瞌睡	買菜	心情好	流汗
逃跑	男人	開電腦	不開心	餅乾	金錢	熬夜	碗	旅行	尿尿

Figure 6. The top 20 words of each topic.

words through a normative rating procedure. Another reason is that sentiments in ANEW is value-level. Value-level sentiments provide additional intensity information.

We want to assign each concept a sentiment value $\in [-1, 1]$ measuring how pleasant or unpleasant people feel about it, so we use the value in "Pleasure" dimension. Values in ANEW are $\in [1, 9]$, so we perform linear normalization on them to value $\in [-1, 1]$.

We apply Google Translate to translate all 1034 words in ANEW into Chinese. There may be multiple translations of a English word, we take all of them into account and verify manually. We verify whether the polarities of a English word and its translations are similar. To match more concepts in Chinese ConceptNet, we expand these translations by the approach in our previous research [**?**]. After expansion, we have 27842 Chinese phrases with sentiment values. In our experiment, there are totally 3047 of them match concepts in Chinese ConceptNet, and then these concepts are used as sentiment seeds.

As the result, given totally M concepts in ConceptNet, we generate a $M \times 1$ vector $\boldsymbol{s}_0$ to denote sentiment values of seed concepts. For each phrase in our expanded translations, if it appears in ConceptNet with concept ID c, the (c,1) entry of $\boldsymbol{s}_0$ is its linear normalized sentiment value in ANEW. Other entries are $unknown$.

3.3. Topic-Aware Sentiment Propagation

After topic analysis, we know which topic each neighbor of each concept comes from by topic assignment matrix $\boldsymbol{T}$. When we predict c_m's sentiment value on the latent topic z, we consider neighbors which come from topic z and use their sentiment values on topic z.

However, not all assertions are good for propagation. Next, We select sentiment related relation types and their directions by validation. More specifically, 10% of positive/neutral/negative sentiment seeds are sampled as validation, and use the remaining 90% to propagate. The result is shown in Table 1.

Starting from seed concepts, we can use equation 1 to iteratively propagate sentiment values on different topics. We design $\boldsymbol{W}$, the $M \times M$ propagation matrix by Algorithm 1. Each element $w_{i,j}$ denotes the weight of sentiment value propagation from concept j to concept i.

TABLE 1. PROPAGATION RULES ON 13 RELATION TYPES

Relation Type	Propagation rule
HasFirstSubevent	Not used
MadeOf	Not used
IsA	Not used
AtLocation	Not used
UsedFor	Not used
CapableOf	Not used
MotivatedByGoal	Latter to Former
Desires	Not used
SymbolOf	Former to Latter
CausesDesire	Not used
Causes	Both Direction
HasSubevent	Former to Latter
PartOf	Not used

Algorithm 1 determine $\boldsymbol{W}$ for a given topic z

Require: z: current topicID; $\boldsymbol{T}$: topic assignment matrix; A $= \{(c1, c2, rel)\}$: all ConceptNet assertions;
Ensure: $M \times M$ propagation matrix $\boldsymbol{W}$;
initialize $M \times M$ matrix $\boldsymbol{W}$ with each entry $w_{i,j} = 0$
for each assertion (c_i, c_j, rel) in A **do**
  rule = searchPropagationRule((c_i, c_j, rel));
  if rule = "Former to Latter" **then**
    $w_{j,i}$ += 1;
  else if rule = "Latter to Former" **then**
    $w_{i,j}$ += 1;
  else if rule = "Both Direction" **then**
    $w_{j,i}$ += 1, $w_{i,j}$ += 1;
  end if
end for
for each $t_{i,j}$ in $\boldsymbol{T}$ **do**
  if $t_{i,j} \neq z$ **then** $w_{i,j} = 0$; ▷ For concept i, consider only neighbors in topic z
  end if
end for
return $\boldsymbol{W}$;

We can design the propagation matrix for each topic and use it to propagate sentiment values separately. No matter which topic, propagation starts from $\boldsymbol{s}_0$ because the seeds are less ambiguous on different topics. In each iteration i, we perform in-link normalization to the propagation matrix. After propagating $iteration$ times, we have the final $\boldsymbol{s}_{iteration}$, which stands for the final sentiment values on topic z for all concepts in ConceptNet. We set sentiment value to 0 for concepts which are still labelled $unknown$ in $\boldsymbol{s}_{iteration}$. Then, this process repeats for other topics. Finally, each concept has a sentiment value $\in [-1, 1]$ on each topic.

4. Topic Inference and Sentiment Value Selection

When predicting polarities for texts, concepts in the texts provide important information. For example, if 'F score', 'sad' and 'die' co-occur in a text, the text is more likely to be negative. We claim that when we want to know

a concept's sentiment value, it's better to return a value based on the current topic, rather than a fixed value. For example, in the sentence 'The case is too sudden', 'sudden' reflects sentiment, but it is both possible to be positive and negative depending on the description about 'the case'. The description may consist of concepts such as 'surprise', 'nervous' and 'help'. Hence, we use co-occurring concepts to help us identify topics and sentiment values for a concept in texts.

That is, given a query concepts c_q and a set of co-occurring concepts COO $= \{(c_i, count_i) \mid c_i$ occurs $count_i$ times$\}$, the output is $(topic, val)$ where $topic$ is the topicID c_q most likely belong to, and val is the sentiment value $\in [-1, 1]$ of c_q when $topic$.

First, we combine c_q and COO to form a document d_q, which is a set of tuples $(c_i, count_i)$ indicating c_i occurs $count_i$ times in c_q and COO. Next, we use the corpus LDA parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}_k$ trained using ConceptNet to infer d_q. By running the E-step, LDA estimates topic distribution of d_q and the topic assignments for concepts in d_q. As the result, we know which topics c_q and COO come from, and use the topic of c_q to access the topic-aware sentiment results in section 3. Finally, the most possible topic and sentiment value of c_q is returned.

5. Experiments

5.1. Experimental Setup

In our experiment, we use the Chinese part of ConceptNet which we collect by ChickenPTT [**?**] until May, 19, 2015. There are totally 713139 assertions and 223871 concepts.

ANEW is used as our seeds. There are 1034 English words in ANEW. After our translation and expansion, these English words generate 27842 Chinese phrases. There are 3047 of them existing in ConceptNet. Among these 3047 concepts, there are 1654 positive, 1042 negative, 351 neutral according to sentiment values in ANEW. (After linear normalization, we define neutral using ± 0.125 as threshold.)

5.2. Experiment 1: Propagation on the Same Topic

This experiment aims to evaluate whether propagating on each topic separately is better than the original method which decides sentiment values from all neighbors together disregarding topics.

5.2.1. Test Data. From 3047 seed concepts, we sample 10% of them to evaluate our propagation result and use the remaining 90% as propagation seeds. To make the distributions of 10% test data and the 90% training data similar, we sample test data according to the proportions of positive, neutral and negative concepts. Therefore, in the test set there are totally 304 concepts where 165 of them are positive, 104 of them are negative and 35 of them are neutral.

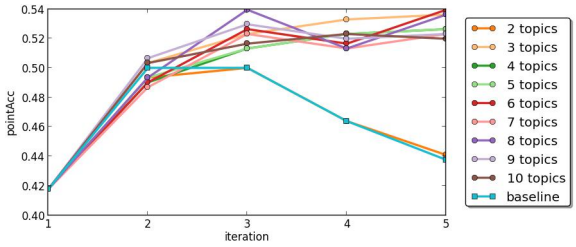


Figure 7. Point-wise accuracy of all test concepts for the first 5 iterations.

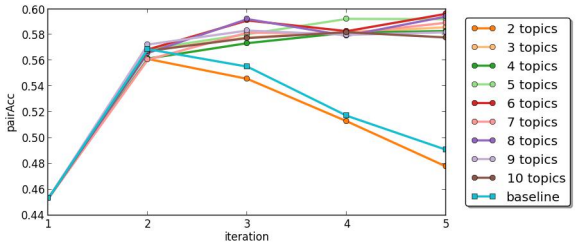


Figure 8. Pairwise accuracy of all test concepts for the first 5 iterations.

5.2.2. Evaluation Metrics. We use point-wise accuracy and pair-wise accuracy to evaluate. Point-wise accuracy measures the proportion of concepts whose predicted polarities are same as their ground-truth to all test concepts. Pair-wise accuracy measures the proportion of pairs of concepts which have the same relative relation as ground-truth to all pairs.

5.2.3. Post-Processing. In our topic-aware results, each concept has different sentiment values on different topics. That is, each concept is associated with multiple sentiment values. To compare with original propagation, we aggregate values on different topics into one value. We compute a weighted arithmetic mean over sentiments on different topics. The weight of sentiment on topic z is the proportion of valid neighbors belonging to z among all valid neighbors.

5.2.4. Results and Discussion. The results of topic-insensitive and topic-aware propagation are discussed here. The former is the baseline, and we compare it with our topic-aware propagations with different topic numbers.

The results of point-wise and pair-wise accuracy are shown in Fig 7 and Fig 8 respectively. We can see that the performance of baseline become worse when propagating sentiments to more concepts. In contrast, topic-aware propagations with topic number > 2 have higher and more stable point-wise accuracy and pairwise accuracy as iteration goes.

Here we investigate topic number $<= 10$. Larger topic number causes topic layers of Chinese ConceptNet sparse and makes sentiment propagation fail. We found generalize contextual information into $<= 10$ topics performs better.

Some of test concepts didn't get sentiment values when propagation and are considered as neutral in our system. We exclude these concepts and investigate the performance of the remaining concepts for each iteration in Fig 9 and Fig 10. The point-wise accuracy and pairwise accuracy reach 0.7.

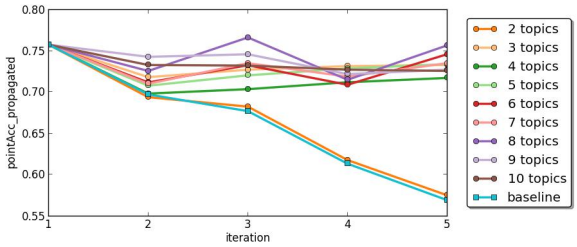


Figure 9. Point-wise accuracy of test concepts propagated for the first 5 iterations.

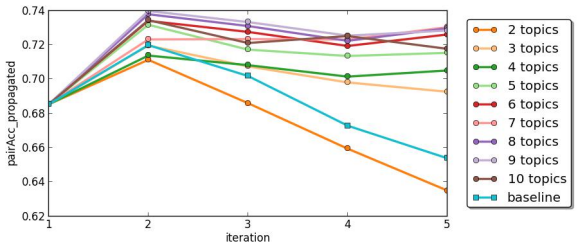


Figure 10. Pairwise accuracy of all test concepts propagated for the first 5 iterations.

Except the topic number $= 2$ one, topic-aware ones have higher accuracies.

5.3. Experiment 2: Polarity Classification for Posts of Microblog

5.3.1. Experimental Settings.

6. Conclusion