# A distributed semantic model based method for instance disambiguation in user-generated short texts

Jiaqi Yang
School of Computer
Northwestern Polytechnical University
Xi'an, Shaanxi 710072, China
1468608569@qq.com

Yongjun Li*
School of Computer
Northwestern Polytechnical University
Xi'an, Shaanxi 710072, China
lyj@nwpu.edu.cn

Congjie Gao
School of Computer
Northwestern Polytechnical University
Xi'an, Shaanxi 710072, China
2451408761@qq.com

## ABSTRACT

Instance disambiguation is to obtain the concept of the target instance in context, which has been attracting much attention from academia. Existing methods are highly dependent on similar or related instances in context. However, the number of instances that can be extracted from a user-generated short text is limited. To tackle this problem, we propose a distributed semantic model (DSM) based method, which consists of three parts. 1) Measuring the correlation between contextual terms and each concept of the ambiguous instance based on DSMs; 2) Filtering out uninformative terms based on the correlations distribution over the concepts, which reduces noise interference; 3) Prioritizing the informative terms to highlight their discriminating capabilities. The concept with the maximum correlation score is considered as the meaning of the target instance. Experiment results demonstrate that the proposed method outperforms baseline methods.

## KEYWORDS

instance disambiguation; distributed semantic model; user-generated short text

**ACM Reference Format:**
Jiaqi Yang, Yongjun Li, and Congjie Gao. 2020. A distributed semantic model based method for instance disambiguation in user-generated short texts. In *Proceedings of KaRS 2019 Second Workshop on Knowledge-Aware and Conversational Recommender Systems (KaRS 2109).* ACM, New York, NY, USA, 4 pages.

## 1 INTRODUCTION

In recent years, user-generated short texts (UGSTs) swept the world at an alarming rate. The study of these data could bring tremendous value for business organizations. To fully exploit these data, we need to understand them better. However, there are some ambiguous instances in UGSTs, which has a great impact on understanding. Therefore, instance disambiguation has been attracting much attention from academia.

Many scholars attempt to eliminate ambiguity based on instances [6] in context. However, an inevitable challenge is the number of instances contained in a UGST is limited. Recently, some efforts have been made to learn knowledge from the context of target instance to improve the performance of disambiguation [1–3]. Generally, there are two strategies. The first one is to use statistical models to obtain the topic of the UGST, and then determine the meaning of the ambiguous instance based on the topic [3]. Due to the sparsity of textual content, building an effective statistical model may not be easy. The second strategy is to use other types of terms for help. Wen et al. [1] found that verbs and adjectives are also helpful for disambiguation. Thus, they constructed a co-occurrence network for typed terms, and then chose the most related contextual term for disambiguation. However, the co-occurrence networks are word-based, which cannot apply to multi-word expressions (MWEs).

In this paper, we propose an Instance Disambiguation method with Context Awareness (IDwCA), which focuses on utilizing various types of contextual terms for disambiguation. Generally, some contextual terms cannot provide us with useful disambiguation information. For convenience, we call them uninformative terms. Otherwise, they are informative terms. To avoid noise interference, we calculate the correlation between contextual terms and each concept of the target instance to filter out uninformative terms. An important basis is the measurement of correlation. The DSMs and Probase are used in the measurement of correlation, which is effective and lightweight. Further, for the remaining contextual terms (informative terms), we prioritize each term to highlight their discrimination. Finally, we recalculated the correlation between informative terms and each concept of the target instance. The concept with the maximum score is considered as the meaning of the target instance. Experiments on ground-truth datasets illustrate the superiority of IDwCA over the-state-of-art methods.

## 2 INSTANCE DISAMBIGUATION

### 2.1 Problem definition

A term $t$ is a word or a MWE. In this paper, we only consider noun terms, verb ($v$) terms and adjective ($adj$) terms, which are very helpful for disambiguation. In addition, for noun terms, we refine them into instances and concepts. While an instance $e$ is a concrete object and a concept $c$ is a general and abstract description of a set of instances. For example, "banana" and "grape" are instances, and they can be explained by the concept "fruit".

Problem Formulation 1. ***Instance disambiguation.*** *Given a UGST $T = \{t_1, t_2, ..., t_m\}$, wherein $t_i$ denotes a term. Assume term $t_k$ is an ambiguous term, and its candidate concept set is denoted by $C = \{c_j | j = 1, 2, ..., l\}$. We define $t_k$ as the target instance and other*

[*] Yongjun Li is the corresponding author.

*KaRS 2109, November 3rd-7th, 2019, Beijing, China.*

*terms in $T$ as contextual terms for $t_k$. The task of IDwCA is to identify the most approximate concept of $t_k$ from $C$.*

The key issue of Problem 1 is to select related terms that have high discriminating capabilities for disambiguation. The main difference from existing work is that we use the corpus and knowledge information together to measure the semantic correlation of terms and then choose more types of contextual terms for disambiguation rather than solely relying on instances.

## 2.2 Proposed approach

In IDwCA, first, DSMs and Probase are used to measure the correlation between all contextual terms and each concept of the target instance. Second, the Kullback Leiber (KL) divergence is employed to filter out uninformative terms. Then for the remaining informative terms, we prioritize them to highlight their discrimination. Finally, based on these informative terms, we obtain the concept of the target instance.

*2.2.1 Correlation calculation between terms and concepts.* We could easily determine the most appropriate concept of the target instance, if we have the knowledge about the semantic correlation between contextual terms and concepts. We use DSMs for help, which focuses on surrounding context of a word and is ideal for calculating correlation. However, they cannot deal with MWEs. We use semantic composition to solve this problem. Given a MWE, denotes as $p$. Assume there are $N$ words in $p$. Given the semantic vector of each word, the vector of $p$ can be calculated by Eq.(1).

$$v(p) = \sum_{c=1}^{N} v(w_c) \tag{1}$$

That is, the vector of $p$ is the sum of the vectors of all the words in it. However, it ignores the syntactic relation between words and may introduce too much noise. To solve this problem, we assign weights to words based on their part-of-speech in $p$, where the weights of nouns, verbs and adjectives are set to 1, and the rest is set to 0. Then, the Eq.(1) can be further expressed as Eq.(2).

$$v(p) = \sum_{c=1}^{N} a_c * v(w_c) \tag{2}$$

where $a_c$ denotes the weight of $w_c$, $a_c \in \{0, 1\}$. Finally, the cosine metric is used to calculate the correlation, as shown in Eq.(3).

$$R_D(t, c) = \cos(v(t), v(c)) \tag{3}$$

Preliminary evaluation shows that the DSM-based method works reasonably well for many pairs of terms, but for some noun terms, the results are less satisfactory. We use Probase to fill this gap, which provides isA knowledge for concepts and instances, and two typicality scores for a concept/instance pair <c,e>: $P(e|c) = n(c, e)/n(c)$ and $P(c|e) = n(c, e)/n(e)$, where $n(\bullet)$ refers to the number of occurrences of a given term or a pair of terms in Probase. Following [5], we use the corresponding context of terms to calculate correlation.

Given a term $t$, we first extract its context $S_t$ from Probase according to its type. The context of term $t$ is detailed as follows.

- If $t$ is a concept, its context is all the instances that can be explained by it.
- If $t$ is an instance, the context is all the concepts it belongs to.
- If $t$ is a verb, or an adjective, because it has no hypernyms [7] in Probase, thus its context is empty.

After then, we transfer the context $S_t$ into a vector $I_t$ as shown in Eq.(4), where each element is the typicality score between $t$ and the term in its context.

$$I_t = \begin{cases} \{P(c_{i1}|t)|i1 = 1, ..., m1\}, & t.type = e \\ \{P(e_{i2}|t)|i2 = 1, ..., m2\}, & t.type = c \end{cases} \tag{4}$$

Then, the measurement of correlation based on Probase can be expressed as Eq.(5)

$$R_P(t, c) = \begin{cases} \frac{\sum_{e_{i2} \in S_t \cap S_c} P(e_{i2}|c) * P(e_{i2}|t)}{||I_t|| * ||I_c||}, & t.type = c \\ \sum_{c_{i1} \in S_t} P(c_{i1}|t) * R_P(c_{i1}, c), & t.type = e \end{cases} \tag{5}$$

where $|| \bullet ||$ denotes the norm of a vector.

Finally, we use a strategy to integrate two parts linearly. In summary, the semantic correlation between terms and concepts can be calculated by Eq.(6).

$$R(t, c) = \begin{cases} R_D(t, c), & t.type \in \{v, adj\} \\ \theta * R_D(t, c) + (1 - \theta) * R_P(t, c), & t.type \in \{e, c\} \end{cases} \tag{6}$$

where $\theta$ is a tuning parameter.

*2.2.2 Contextual term filtering.* Normally, some contextual terms do not contains useful disambiguation information, so we filter them out to avoid noise interference. For clarity, we take "the apple is really delicious" as an example. Based on "delicious", we know "apple" is "a kind of fruit". This is because "delicious" is more related to "fruit" than to "company". However, if we filter out the uninformative terms directly according to the correlation scores, we need to set a threshold dynamically, which poses a big challenge. Following [1], we employ the KL divergence. First, we assume that the probabilities of concepts of the target instance are the same. That is, it fits a uniform distribution. Second we calculate the correlation between contextual terms and each concept, and normalize the scores to get a new distribution. Then, the KL divergence is used to measure the divergence between two distributions. The greater the divergence is, the more important the role of the term is. Finally, based on KL divergence, we set a threshold to filter out uninformative terms and obtain a new set of informative terms, denotes as $ICT$.

*2.2.3 Weights of informative terms.* Generally, the concept of the target instance depends heavily on the choice of contextual terms. Take "the engineer is eating the apple" as an example, the $ICT$ is {"engineer","eating"}, the concept of "apple" is "company" according to "engineer", while its concept is "fruit" if based on "eating". However, an ambiguous instance cannot has different concepts simultaneously. To solve this problem, we prioritize each informative term to highlight their contributions. Intuition is that the closer the informative is to the target instance, the greater its contribution. We propose a weighting function based on sigmoid, which is described in Eq.(7).

$$weight(t_i) = 1.5 - \frac{1}{1 + e^{-x}} \tag{7}$$

where $x$ represents the context distance, and the context distance refers to the number of terms between $t_i$ and the target instance.

Based on Eq.(6) and Eq.(7), we define the semantic correlation between all informative terms and a concept of the target instance, $R(ICT, c)$, as described in Eq.(8).

$$R(ICT, c) = \sum_{t_p \in ICT} weight(t_p) * R(t_p, c) \tag{8}$$

The concept with the maximum score is the result of IDwCA.

# 3 EXPERMIMENTS

## 3.1 Datasets and baseline algorithms

As we know, there is no gold standard metric for evaluating instance disambiguation methods. Therefore, we evaluate our method in terms of classification. To verify the validity and generality of the method, we chose Foursquare, Twitter and Facebook as data sources. These social networking sites are popular sites and provide us with open data acquisition APIs. Then, we randomly selected UGSTs from the acquired data contained ambiguous instance "apple", "Harry Potter" and "python". We classified the data manually. For convenience, three datasets are abbreviated as FS, FB and TW, respectively. Table 1 shows the statistics of the ambiguous instance "apple" on three datasets.And the continuous Bag-of-Words model is used in our experiments to obtain the semantic vector of words, which is the one of the most commonly used DMSs. The wiki[1] dataset is used for training the model. We compare our approach with the following representative methods: STC-NB [6] and TD [4].

**Table 1: Details of FS, FB and TW**

| Datasets<br>category | FS | FB | TW |
|---|---|---|---|
| fruit | 134 | 10 | 131 |
| company | 42 | 674 | 19 |

## 3.2 Performance comparison between IDwCA and existing work

We illustrate the results on three datasets in Figure 1. From the results, we reach the following conclusions. IDwCA outperforms all baselines, which validates its effectiveness. It is reasonable since IDwCA 1) utilizes information from DSMs and Probase to measure the semantic correlation, and then chooses various types of contextual terms for disambiguation, not just relying on instances; 2) assigns weights to informative terms based on their context distances, which reduces noise interference.

The STC-NB performs worse than other methods, because it only considers similar instances, and the correlation between terms are calculated by their co-occurrence times in Probase. Compared with IDwCA, TD achieves worse performances. This is because it divides terms into two types: instances and concepts, which may lead to wrong judgements. And its correlation calculation method does not work well in oral expressions.

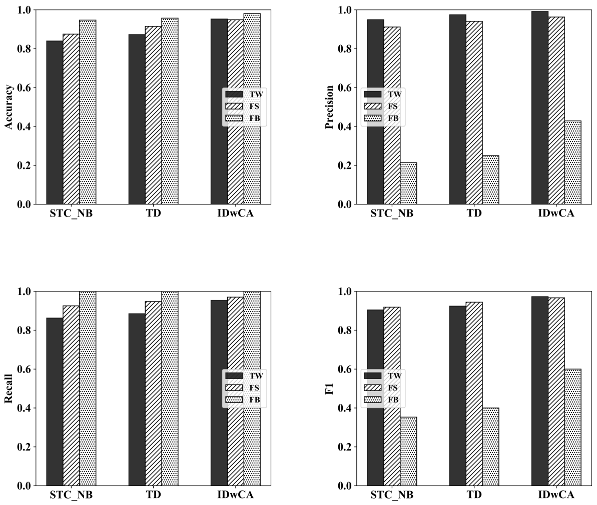

**Figure 1: Results on TW, FS and FB**

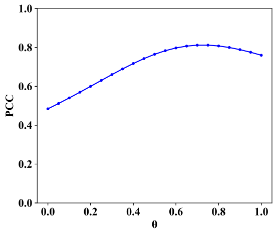

**Figure 2: Results w.r.t. $\theta$**

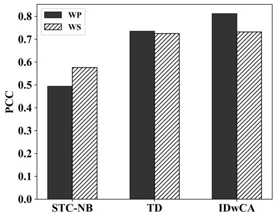

**Figure 3: Results on WP, WS**

## 3.3 Performance of correlation calculation method

Further, we explore the performance our correlation calculation method. We utilize two datasets in the following experiments: one well-known dataset WordSim353 [2] (WS) for words and one labeled data WP for MWEs created by [5]. We compare our method with the baseline algorithms. To evaluate the experiment, we computed the Pearson Correlation Coefficient (PCC) to measure the machine ratings and the human ratings over the two datasets. From the results shown in Figure 3, we observe that IDwCA performs the best on all datasets. This is because knowledge bases are more suitable for noun-based terms than for other types of terms, and IDwCA uses a combination of DSMs to solve ts problem. Meanwhile, as shown in Eq.(6), the threshold $\theta$ is used to tune the importance of each part. To study the effect of $\theta$, we conduct experiment based on different values of $\theta$. The WP dataset is used in the experiment. As shown in Figure 2, we can see DSMs contribute more to the correlation. This is mainly due to the fact that DSMs are more suitable for oral expressions. In our experiments, we select the value of $\theta = 0.75$ as an optimal value.

# 4 CONCLUSION

In this paper, we use DSMs and Probase to measure the correlation of terms and then choose various types of contextual terms for disambiguation. Experiments on ground-truth datasets validate the effectiveness of the proposed method.

[1] https://dumps.wikimedia.org/enwiki/latest/

[2] http://alfonseca.org/eng/research/wordsim353.html

## REFERENCES

[1] Wen Hua, Zhongyuan Wang, Haixun Wang, Kai Zheng, and Xiaofang Zhou. 2017. Understand Short Texts by Harvesting and Analyzing Semantic Knowledge. *IEEE Trans. Knowl. Data Eng.* 29, 3 (2017), 499–512.

[2] Heyan Huang, Yashen Wang, Chong Feng, Zhirun Liu, and Qiang Zhou. 2018. Leveraging Conceptualization for Short-Text Embedding. *IEEE Trans. Knowl. Data Eng.* 30, 7 (2018), 1282–1295.

[3] Dongwoo Kim, Haixun Wang, and Alice H. Oh. 2013. Context-Dependent Conceptualization. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, Francesca Rossi (Ed.). IJCAI/AAAI, Palo Alto, CA, USA, 2654–2661.

[4] Pei-Pei Li, Lu He, Haiyan Wang, Xuegang Hu, Yuhong Zhang, Lei Li, and Xindong Wu. 2018. Learning From Short Text Streams With Topic Drifts. *IEEE Trans. Cybernetics* 48, 9 (2018), 2697–2711.

[5] Pei-Pei Li, Haixun Wang, Kenny Q. Zhu, Zhongyuan Wang, Xuegang Hu, and Xindong Wu. 2015. A Large Probabilistic Semantic Network Based Approach to Compute Term Similarity. *IEEE Trans. Knowl. Data Eng.* 27, 10 (2015), 2604–2617.

[6] Yangqiu Song, Haixun Wang, Zhongyuan Wang, Hongsong Li, and Weizhu Chen. 2011. Short Text Conceptualization Using a Probabilistic Knowledgebase. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, Toby Walsh (Ed.). IJCAI/AAAI, Palo Alto, CA, USA, 2330–2336.

[7] Wentao Wu, Hongsong Li, Haixun Wang, and Kenny Qili Zhu. 2012. Probase: a probabilistic taxonomy for text understanding. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2012, Scottsdale, AZ, USA, May 20-24, 2012.* ACM, New York, NY, USA, 481–492.