Verb Pattern: A Probabilistic Semantic Representation on Verbs

Abstract

Verbs are important in semantic understanding of natural language. Traditional verb representations, such as FrameNet, PropBank, VerbNet, focus on verbs' roles. These roles are too coarse to represent verbs' semantics. In this paper, we introduce verb patterns to represent verbs' semantics, such that each pattern corresponds to a single semantic of the verb. First we analyze the principles for verb patterns: generality and specificity. Then we propose a nonparametric model based on description length. Experimental results prove the high effectiveness of verb patterns. We further apply verb patterns to context-aware conceptualization, to show that verb patterns are helpful in semantic-related tasks.

Introduction

Verb is crucial in sentence understanding BIBREF0 , BIBREF1 . A major issue of verb understanding is polysemy BIBREF2 , which means that a verb has different semantics or senses when collocating with different objects. In this paper, we only focus on verbs that collocate with objects. As illustrated in Example SECREF1 , most verbs are polysemous. Hence, a good semantic representation of verbs should be aware of their polysemy.

Example 1 (Verb Polysemy) eat has the following senses:

Many typical verb representations, including FrameNet BIBREF3 , PropBank BIBREF4 , and VerbNet BIBREF5 , describe verbs' semantic roles (e.g. ingestor and ingestibles for “eat”). However, semantic roles in general are too coarse to differentiate a verb's fine-grained semantics. A verb in different phrases can have different semantics but similar roles. In Example SECREF1 , both “eat”s in “eat breakfast” and “eat apple” have ingestor. But they have different semantics.

The unawareness of verbs' polysemy makes traditional verb representations unable to fully understand the verb in some applications. In sentence I like eating pitaya, people directly know “pitaya” is probably one kind of food since eating a food is the most fundamental semantic of “eat”. This enables context-aware conceptualization of pitaya to food concept. But by only knowing pitaya's role is the “ingestibles”, traditional representations cannot tell if pitaya is a food or a meal.

Verb Patterns We argue that verb patterns (available at http://kw.fudan.edu.cn/verb) can be used to represent more fine-grained semantics of a verb. We design verb patterns based on two word collocations principles proposed in corpus linguistics BIBREF6 : idiom principle and open-choice principle. Following the principles, we designed two types of verb patterns.

According to the above definitions, we use verb patterns to represent the verb's semantics. Phrases assigned to the same pattern have similar semantics, while those assigned to different patterns have different semantics. By verb patterns, we know the “pitaya” in I like eating pitaya is a food by mapping “eat pitaya” to “eat $ INLINEFORM0 food”. On the other hand, idiom patterns specify which phrases should not be conceptualized. We list verb phrases from Example SECREF1 and their verb patterns in Table TABREF7 . And we will show how context-aware conceptualization benefits from our verb patterns in the application section.

Thus, our problem is how to generate conceptualized patterns and idiom patterns for verbs. We use two public data sets for this purpose: Google Syntactic N-Grams (http://commondatastorage.googleapis.com/books/syntactic -ngrams/index.html) and Probase BIBREF7 . Google Syntactic N-grams contains millions of verb phrases, which allows us to mine rich patterns for verbs. Probase contains rich concepts for instances, which enables the conceptualization for objects.

Thus, our problem is given a verb INLINEFORM0 and a set of its phrases, generating a set of patterns (either conceptualized patterns or idiom patterns) for INLINEFORM1 . However, the pattern generation for verbs is non-trivial. In general, the most critical challenge we face is the trade-off between generality and specificity of the generated patterns, as explained below.

Trade-off between Generality and Specificity

We try to answer the question: “what are good verb patterns to summarize a set of verb phrases?” This is hard because in general we have multiple candidate verb patterns. Intuitively, good verb patterns should be aware of the generality and specificity.

Generality In general, we hope to use fewer patterns to represent the verbs' semantics. Otherwise, the extracted patterns will be trivial. Consider one extreme case where all phrases are considered as idiom phrases. Such idiom patterns obviously make no sense since idioms in general are a minority of the verb phrases.

Example 2 In Fig FIGREF9 , (eat $ INLINEFORM0 meal) is obviously better than the three patterns (eat $ INLINEFORM1 breakfast + eat $ INLINEFORM2 lunch+ eat $ INLINEFORM3 dinner). The former case provides a more general representation.

Specificity On the other hand, we expect the generated patterns are specific enough, or the results might be trivial. As shown in Example SECREF11 , we can generate the objects into some high-level concepts such as activity, thing, and item. These conceptualized patterns in general are too vague to characterize a verb's fine-grained semantic.

Example 3 For phrases in Fig FIGREF9 , eat $ INLINEFORM0 activity is more general than eat $

INLINEFORM1 meal. As a result, some wrong verb phrases such as eat shopping or each fishing can be recognized as a valid instance of phrases for eat. Instead, eat $ INLINEFORM2 meal has good specificity. This is because breakfast, lunch, dinner are three typical instances of meal, and meal has few other instances.

Contributions Generality and specificity obviously contradict to each other. How to find a good trade-off between them is the main challenge in this paper. We will use minimum description length (MDL) as the basic framework to reconcile the two objectives. More specifically, our contribution in this paper can be summarized as follows:

We proposed verb patterns, a novel semantic representations of verb. We proposed two types of verb patterns: conceptualized patterns and idiom patterns. The verb pattern is polysemy-aware so that we can use it to distinguish different verb semantics.

We proposed the principles for verb pattern extraction: generality and specificity. We show that the trade-off between them is the main challenge of pattern generation. We further proposed an unsupervised model based on minimum description length to generate verb patterns.

We conducted extensive experiments. The results verify the effectiveness of our model and algorithm. We presented the applications of verb patterns in context-aware conceptualization. The application justifies the effectiveness of verb patterns to represent verb semantics.

Problem Model

In this section, we define the problem of extracting patterns for verb phrases. The goal of pattern extraction is to compute: (1) the pattern for each verb phrase; (2) the pattern distribution for each verb.

Next, we first give some preliminary definitions. Then we formalize our problem based on minimum description length. The patterns of different verbs are independent from each other. Hence, we only need to focus on each individual verb and its phrases. In the following text, we discuss our solution with respect to a given verb.

Preliminary Definitions

First, we formalize the definition of verb phrase, verb pattern, and pattern assignment. A verb phrase INLINEFORM0 is in the form of verb + object (e.g. “eat apple”). We denote the object in INLINEFORM1 as INLINEFORM2 . A verb pattern is either an idiom pattern or a conceptualized pattern. Idiom Pattern is in the form of verb $ INLINEFORM3 object (e.g. eat $ INLINEFORM4 humble pie). Conceptualized Pattern is in the form of verb $ INLINEFORM5 concept (e.g. eat $ INLINEFORM6 meal). We denote the concept in a conceptualized pattern INLINEFORM7 as INLINEFORM8 .

Definition 1 (Pattern Assignment) A pattern assignment is a function INLINEFORM0 that maps an arbitrary phrase INLINEFORM1 to its pattern INLINEFORM2 . INLINEFORM3 means the pattern of INLINEFORM4 is INLINEFORM5 . The assignment has two constraints:

For an idiom pattern verb $ INLINEFORM0 object, only phrase verb object can map to it.

For a conceptualized pattern verb $ INLINEFORM0 concept, a phrase verb object can map to it only if the object belongs to the concept in Probase BIBREF7 .

An example of verb phrases, verb patterns, and a valid pattern assignment is shown in Table TABREF7 .

We assume the phrase distribution is known (in our experiments, such distribution is derived from Google

Syntactic Ngram). So the goal of this paper is to find INLINEFORM0 . With INLINEFORM1 , we can easily compute the pattern distribution INLINEFORM2 by: DISPLAYFORM0

, where INLINEFORM0 is the probability to observe phrase INLINEFORM1 in all phrases of the verb of interest. Note that the second equation holds due to the obvious fact that INLINEFORM2 when INLINEFORM3 . INLINEFORM4 can be directly estimated as the ratio of INLINEFORM5 's frequency as in Eq EQREF45 .

Model

Next, we formalize our model based on minimum description length. We first discuss our intuition to use Minimum Description Length (MDL) BIBREF8 . MDL is based on the idea of data compression. Verb patterns can be regarded as a compressed representation of verb phrases. Intuitively, if the pattern assignment provides a compact description of phrases, it captures the underlying verb semantics well.

Given verb phrases, we seek for the best assignment function INLINEFORM0 that minimizes the code length of phrases. Let INLINEFORM1 be the code length derived by INLINEFORM2 . The problem of verb pattern assignment thus can be formalized as below:

Problem Definition 1 (Pattern Assignment) Given the phrase distribution INLINEFORM0 , find the pattern assignment INLINEFORM1 , such that INLINEFORM2 is minimized: DISPLAYFORM0

We use a two-part encoding schema to encode each phrase. For each phrase INLINEFORM0 , we need to encode its pattern INLINEFORM1 (let the code length be INLINEFORM2) as well as the INLINEFORM3 itself given INLINEFORM4 (let the code length be INLINEFORM5). Thus, we have DISPLAYFORM0

Here INLINEFORM0 is the code length of INLINEFORM1 and consists of INLINEFORM2 and INLINEFORM3 .

INLINEFORM0 : Code Length for Patterns To encode INLINEFORM1 's pattern INLINEFORM2 , we need: DISPLAYFORM0

bits, where INLINEFORM0 is computed by Eq EQREF19 .

INLINEFORM0 : Code Length for Phrase given Pattern After knowing its pattern INLINEFORM1 , we use INLINEFORM2 , the probability of INLINEFORM3 given INLINEFORM4 to encode INLINEFORM5 . INLINEFORM6 is computed from Probase BIBREF7 and is treated as a prior. Thus, we encode INLINEFORM7 with code length INLINEFORM8 . To compute INLINEFORM9 , we consider two cases:

Case 1: INLINEFORM0 is an idiom pattern. Since each idiom pattern has only one phrase, we have INLINEFORM1 .

Case 2: INLINEFORM0 is a conceptualized pattern. In this case, we only need to encode the object INLINEFORM1 given the concept in INLINEFORM2 . We leverage INLINEFORM3 , the probability of object INLINEFORM4 given concept INLINEFORM5 (which is given by the isA taxonomy), to encode the phrase. We will give more details about the probability computation in the experimental settings.

Thus, we have DISPLAYFORM0

Total Length We sum up the code length for all phrases to get the total code length INLINEFORM0 for assignment INLINEFORM1 : DISPLAYFORM0

Note that here we introduce the parameter INLINEFORM0 to control the relative importance of INLINEFORM1 and INLINEFORM2 . Next, we will explain that INLINEFORM3 actually reflects the trade-off between the generality and the specificity of the patterns.

Rationality

Next, we elaborate the rationality of our model by showing how the model reflects principles of verb patterns (i.e. generality and specificity). For simplicity, we define INLINEFORM0 and INLINEFORM1 as below to denote the total code length for patterns and total code length for phrases themselves: DISPLAYFORM0 DISPLAYFORM1

Generality We show that by minimizing INLINEFORM0 , our model can find general patterns. Let INLINEFORM1 be all the patterns that INLINEFORM2 maps to and INLINEFORM3 be the set of each phrase INLINEFORM4 such that INLINEFORM5 . Due to Eq EQREF19 and Eq EQREF30 , we have: DISPLAYFORM0

So INLINEFORM0 is the entropy of the pattern distribution. Minimizing the entropy favors the assignment that maps phrases to fewer patterns. This satisfies the generality principle.

Specificity We show that by minimizing INLINEFORM0 , our model finds specific patterns. The inner part in the last equation of Eq EQREF33 actually is the cross entropy between INLINEFORM1 and INLINEFORM2 . Thus INLINEFORM3 has a small value if INLINEFORM4 and INLINEFORM5 have similar distributions. This reflects the specificity principle. DISPLAYFORM0

Algorithm

In this section, we propose an algorithm based on simulated annealing to solve Problem SECREF21 . We also show how we use external knowledge to optimize the idiom patterns.

We adopted a simulated annealing (SA) algorithm to compute the best pattern assignment INLINEFORM0 . The algorithm proceeds as follows. We first pick a random assignment as the initialization (initial temperature). Then, we generate a new assignment and evaluate it. If it is a better assignment, we replace the previous assignment with it; otherwise we accept it with a certain probability (temperature reduction). The generation and replacement step are repeated until no change occurs in the last INLINEFORM1 iterations (termination condition).

Settings

Verb Phrase Data The pattern assignment uses the phrase distribution INLINEFORM0 . To do this, we use the “English All” dataset in Google Syntactic N-Grams. The dataset contains counted syntactic ngrams extracted from the English portion of the Google Books corpus. It contains 22,230 different verbs (without stemming), and 147,056 verb phrases. For a fixed verb, we compute the probability of phrase INLINEFORM1 by: DISPLAYFORM0

, where INLINEFORM0 is the frequency of INLINEFORM1 in the corpus, and the denominator sums over all phrases of this verb.

IsA Relationship We use Probase to compute the probability of an entity given a concept INLINEFORM0 , as well as the probability of the concept given an entity INLINEFORM1 : DISPLAYFORM0

,where INLINEFORM0 is the frequency that INLINEFORM1 and INLINEFORM2 co-occur in Probase.

Test data We use two data sets to show our solution can achieve consistent effectiveness on both short text and long text. The short text data set contains 1.6 millions of tweets from Twitter BIBREF9 . The long text data set contains 21,578 news articles from Reuters BIBREF10 .

Statistics of Verb Patterns

Now we give an overview of our extracted verb patterns. For all 22,230 verbs, we report the statistics for the top 100 verbs of the highest frequency. After filtering noisy phrases with INLINEFORM0 , each verb has 171 distinct phrases and 97.2 distinct patterns on average. 53% phrases have conceptualized patterns. 47% phrases have idiom patterns. In Table TABREF48 , we list 5 typical verbs and their top patterns. The case study verified that (1) our definition of verb pattern reflects verb's polysemy; (2) most verb patterns we found are meaningful.

Effectiveness

To evaluate the effectiveness of our pattern summarization approach, we report two metrics: (1) (INLINEFORM0) how much of the verb phrases in natural language our solution can find corresponding patterns (2) (INLINEFORM1) how much of the phrases and their corresponding patterns are correctly matched? We compute the two metrics by: DISPLAYFORM0

,where INLINEFORM0 is the number of phrases in the test data for which our solution finds corresponding patterns, INLINEFORM1 is the total number of phrases, INLINEFORM2 is the number of phrases whose corresponding patterns are correct. To evaluate INLINEFORM3 , we randomly selected 100 verb phrases from the test data and ask volunteers to label the correctness of their assigned patterns. We regard a phrase-pattern matching is incorrect if it's either too specific or too general (see examples in Fig FIGREF9). For comparison, we also tested two baselines for pattern summarization:

Idiomatic Baseline (IB) We treat each verb phrase as a idiom.

Conceptualized Baseline (CB) For each phrase, we assign it to a conceptualized pattern. For object INLINEFORM0 , we choose the concept with the highest probability, i.e. INLINEFORM1 , to construct the pattern.

Verb patterns cover 64.3% and 70% verb phrases in Tweets and News, respectively. Considering the spelling errors or parsing errors in Google N-Gram data, the coverage in general is acceptable. We report the precision of the extracted verb patterns (VP) with the comparisons to baselines in Fig FIGREF53 . The results show that our approach (VP) has a significant priority over the baselines in terms of precision. The result suggests that both conceptualized patterns and idiom patterns are necessary for the semantic representation of verbs.

Application: Context-Aware Conceptualization

As suggested in the introduction, we can use verb patterns to improve context-aware conceptualization (i.e. to extract an entity's concept while considering its context). We do this by incorporating the verb patterns into a state-of-the-art entity-based approach BIBREF11 .

Entity-based approach The approach conceptualizes an entity INLINEFORM0 by fully employing the mentioned entities in the context. Let INLINEFORM1 be entities in the context. We denote the probability that INLINEFORM2 is the concept of INLINEFORM3 given the context INLINEFORM4 as INLINEFORM5 . By assuming all these entities are independent for the given concept, we compute INLINEFORM6 by: DISPLAYFORM0

Our approach We add the verb in the context as an additional feature to conceptualize INLINEFORM0 when INLINEFORM1 is an object of the verb. From verb patterns, we can derive INLINEFORM2 , which is the probability to observe the conceptualized pattern with concept INLINEFORM3 in all phrases of verb INLINEFORM4 . Thus, the probability of INLINEFORM5 conditioned on INLINEFORM6 given the context INLINEFORM7 as well as verb INLINEFORM8 is INLINEFORM9 . Similar to Eq EQREF54 , we compute it by: DISPLAYFORM0

Note that if INLINEFORM0 is observed in Google Syntactic N-Grams, which means that we have already learned its pattern, then we can use these verb patterns to do the conceptualization. That is, if INLINEFORM1 is mapped to a conceptualized pattern, we use the pattern's concept as the conceptualization result. If INLINEFORM2 is an idiom pattern, we stop the conceptualization.

Settings and Results For the two datasets used in the experimental section, we use both approaches to conceptualize objects in all verb phrases. Then, we select the concept with the highest probability as the label of the object. We randomly select 100 phrases for which the two approaches generate different labels. For each difference, we manually label if our result is better than, equal to, or worse than the competitor. Results are shown in Fig FIGREF56 . On both datasets, the precisions are significantly improved after adding verb patterns. This verifies that verb patterns are helpful in semantic understanding tasks.

Related Work

Traditional Verb Representations We compare verb patterns with traditional verb representations BIBREF12 . FrameNet BIBREF3 is built upon the idea that the meanings of most words can be best understood by semantic frames BIBREF13 . Semantic frame is a description of a type of event, relation, or entity and the participants in it. And each semantic frame uses frame elements (FEs) to make simple annotations. PropBank BIBREF4 uses manually labeled predicates and arguments of semantic roles, to capture the precise predicate-argument structure. The predicates here are verbs, while arguments are other roles of verb. To make PropBank more formalized, the arguments always consist of agent, patient, instrument, starting point and ending point. VerbNet BIBREF5 classifies verbs according to their syntax patterns based on Levin classes BIBREF14 . All these verb representations focus on different roles of the verb instead of the semantics of verb. While different verb semantics might have similar roles, the existing representations cannot fully characterize the verb's semantics.

Conceptualization One typical application of our work is context-aware conceptualization, which motivates the survey of the conceptualization. Conceptualization determines the most appropriate concept for an entity.Traditional text retrieval based approaches use NER BIBREF15 for conceptualization. But NER usually has only a few predefined coarse concepts. Wu et al. built a knowledge base with large-scale lexical information to provide richer IsA relations BIBREF7 . Using IsA relations, context-aware conceptualization BIBREF16 performs better. Song et al. BIBREF11 proposed a conceptualization mechanism by Naive Bayes. And Wen et al. BIBREF17 proposed a state-of-the-art model by combining co-occurrence network, IsA network and concept clusters.

Semantic Composition We represent verb phrases by verb patterns. while semantic composition works aim to represent the meaning of an arbitrary phrase as a vector or a tree. Vector-space model is widely used to represent the semantic of single word. A straightforward approach to characterize the semantic of a phrase thus is averaging the vectors over all the phrase's words BIBREF18 . But this approach certainly ignores the syntactic relation BIBREF19 between words. Socher et al. BIBREF20 represent the syntactic relation by a binary tree, which is fed into a recursive neural network together with the words' vectors. Recently, word2vec BIBREF21 shows its advantage in single word representation. Mikolov et al. BIBREF22 further revise it to make word2vec capable for phrase vector. In summary, none of these works uses the idiom phrases of verbs and concept of verb's object to represent the semantics of verbs.

Conclusion

Verbs' semantics are important in text understanding. In this paper, we proposed verb patterns, which can distinguish different verb semantics. We built a model based on minimum description length to trade-off between generality and specificity of verb patterns. We also proposed a simulated annealing based algorithm to extract verb patterns. We leverage patterns' typicality to accelerate the convergence by pattern-based candidate generation. Experiments justify the high precision and coverage of our extracted patterns. We also presented a successful application of verb patterns into context-aware conceptualization.