Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence

Collective Nominal Semantic Role Labeling for Tweets

Xiaohua Liu [‡ †], Zhongyang Fu [§], Xiangyang Zhou [♯], Furu Wei [†], Ming Zhou [†]
[‡]School of Computer Science and Technology
Harbin Institute of Technology, Harbin, 150001, China
[§]Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, 200240, China
[♯]School of Computer Science and Technology
Shandong University, Jinan, 250100, China
[†]Microsoft Research Asia
Beijing, 100190, China
[†]{*xiaoliu, fuwei, mingzhou*}*@microsoft.com*
[§] *zhongyang.fu@gmail.com* [♯] *v-xzho@microsoft.com*

Abstract

Tweets have become an increasingly popular source of fresh information. We investigate the task of Nominal Semantic Role Labeling (NSRL) for tweets, which aims to identify predicate-argument structures defined by nominals in tweets. Studies of this task can help fine-grained information extraction and retrieval from tweets. There are two main challenges in this task: 1) The lack of information in a single tweet, rooted in the short and noisy nature of tweets; and 2) recovery of implicit arguments. We propose jointly conducting NSRL on multiple similar tweets using a graphical model, leveraging the redundancy in tweets to tackle these challenges. Extensive evaluations on a human annotated data set demonstrate that our method outperforms two baselines with an absolute gain of 2.7% in F1.

Introduction

Tweets are short messages shared through Twitter [1], which now represents an important source of fresh information. And in recent days, we have witnessed increasing research interests in tweets. For example, Kwak et al. (2010) study the problem of detecting influential Twitter users; Liu et al. (2011c) investigate the task of named entity recognition for tweets. Particularly, Semantic Role Labeling (SRL) is revisited in the new context of tweets (Liu et al. 2011b; 2011a).

Current studies of SRL for tweets mainly focus on verbs. Existing approaches to verbal SRL (VSRL) usually consist of two steps. Firstly, a Part-Of-Speech (POS) tagger is used to identify verbs as predicates; and then, a sequential labeling model is adopted to recognize all arguments and classify their semantic roles for each predicate. One main challenge of this task is that a tweet is often short and noisy, and as a result, existing natural language processing tools perform poorly on tweets (Ritter et al. 2011; Liu et al. 2011c). This means conventional features that prove helpful for SRL, such as syntax parsing related features, are now unreliable. To tackle this challenge, Liu et al. (2011a) first cluster similar tweets and then do two-stage labeling for each cluster as follows. In the first stage, a linear Conditional Random Fields (CRF) based labeler is applied to each tweet. In the second stage, another linear CRF based labeler uses statistic information collected from the outputs of the first stage, to refine the preliminary results.

We enlarge the scope of this line of research from verbal predicates to nominal predicates. Two factors motivate our work. On the one hand, we have observed abundant nominal predicates in tweets. As demonstrated by our investigation of 10,000 randomly sampled tweets, 12.3% of tweets contain at least one nominal predicate. On the other hand, we have found that many meaningful events are triggered by nominal predicates. As shown by a manually annotated tweet event corpus, 25.8% events are expressed through nominal predicates and their semantic arguments [2].

There are several remarkable differences between Nominal SRL (NSRL) and VSRL. Above all, NSRL focuses on nominal predicates, whose number is much larger than the number of verbs. Secondly, NSRL requires a predicate recognition module since only a small number of nouns are predicates (Li et al. 2009), in contrast with the fact that nearly all verbs are meaningful predicates. Finally, compared with a verb predicate, a nominal predicate tends to have fewer explicit and more implicit arguments that are not explicitly stated in the current sentence but can be recovered in a larger context (Gerber and Chai 2010). To illustrate, consider the following two sentences [3]: "[arg_0 The two companies] [$pred$ produce] [arg_1 market pulp, container board and white paper]. The goods could be manufactured closer to customers, saving [$pred$ shipping] costs." where:

[1]http://twitter.com/

[2]For instance, ("beginning", revolts, arg_1) in the tweet "this is just the beginning of mass revolts across the us" expresses a DEMONSTRATE event.

[3]This example owes to Gerber and Chai (2010).

"produce" in the first sentence is a verbal predicate with agentive producer arg_0 and produced entity arg_1; "shipping" in the second sentence is a nominal predicate without any associated argument. However, we can infer that "The two companies" refers to the agents (arg_0) of the "shipping" predicate, and "market pulp, container board and white paper" refers to the shipped entities (arg_1) of "shipping".

The challenges of NSRL for tweets lie in two aspects. First, as demonstrated by Liu et al. (2011a; 2010), short and noisy tweets cannot offer the reliable evidence required by SRL using current NLP tools. Second, it is often required to resolve the implicit arguments for a nominal predicate [4]. This in turn requires more contextual information to be provided, which, however, goes beyond the capability of a single tweet.

We propose a factor graph-based method, which collectively conducts NSRL on multiple similar tweets to address these challenges. Our method is based on the following observation [5]: A nominal predicate appearing in multiple similar tweets tends to have similar predicate-argument structures. That means, a predicate-argument segmentation is likely to be shared across multiple similar tweets. We classify two tweets as similar if and only if: 1) Their content is similar, e.g., their cosine similarity is above a threshold (0.4 in our work) and they contain a common hash tag; and 2) they fall into the same time period (half a day in our work). As an illustrated example, consider the following two tweets: "[arg_0 Myanmar]'s [$pred$ release] of some long-time political [arg_1 prisoners] is important step$\cdots$" and "US to send ambassador to [arg_0 Myanmar]!!! upgrading ties after [$pred$ release] of political [arg_1 prisoners]" [6]. It is straightforward to recognize "Myanmar" as the arg_0 of "release" for the first tweet, owing to the short distance between them [7] and the strong signal offered by "'s"; while it is harder for the second, because of the long distance between them and the fact that "Myanmar" is an implicit argument of "release" (since they are separated by "!", and are not in the same sentence.). Intuitively, knowing that ([$pred$ release],[arg_0 Myanmar]) exists in some similar tweet will encourage us to guess the same predicate-argument exists in the second tweet.

Our method consists of two steps. In the first step, it identifies each nominal predicate for each tweet using a Support Vector Machine (SVM) (Cortes and Vapnik 1995) based binary classifier. In the second step, it constructs a factor graph for each nominal predicate on which its semantic arguments are jointly resolved. Let p denote the target nominal predicate, and T_p denote the tweets that contain p [8]. For each tweet in T_p, a random variable is introduced for each word in the tweet, the value (such as arg_0 and arg_1) of which indicates the semantic role played by the word w.r.t. p. Hereafter y_m^i is used to denote the variable for the i^{th} word in the m^{th} tweet in T_p. Each pair of neighboring variables, i.e., y_m^{i-1} and y_m^i, is connected by a factor f_m, forming a set of linear Conditional Random Field (CRF) chains, each representing a tweet. Then a factor is added for every two variables whose corresponding words share the same lemma and come from similar tweets. We use f_{mn}^{ij} to denote the factor connecting y_m^i and y_n^j. Figure 1 illustrates an example of our factor graph.

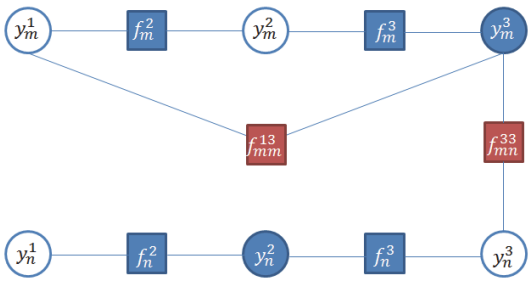

Figure 1: A factor graph for NSRL. Circles represent random variables, each of which denotes the semantic role of the corresponded word played w.r.t. the given predicate; solid circles mean the values of the corresponding variables are visible. Blue and red rectangles represent factors related to a single variable and two variables, respectively.

7,000 tweets are manually annotated as the gold-standard data set. Experimental results on this data set show that our method consistently outperforms the baseline, with an absolute gain of of 2.7% in F1. We also study the contribution of collective inference on multiple tweets.

Our contributions are summarized as follows:

1. We introduce the task of nominal SRL for tweets. Two main challenges are the lack of information in a single tweet and the resolution of implicit arguments.
2. We propose collectively conducting NSRL on multiple similar tweets to tackle these challenges. We evaluate our method on a human-annotated data set and show the advantages of our method.

The rest of our paper is organized as follows. In the next section, we introduce related work. In Section 3 and 4, we define the task and present our method. In Section 5, we evaluate our method. Finally, we conclude in Section 6 with discussions of future work.

Related Work

Related work can be divided into two categories: verbal SRL and nominal SRL.

Verbal SRL

Since Gildea and Jurafsky (2002a) first introduce SRL, verbal predicates have been extensively studied, and various approaches have been practised, among which data driven ap-

[4] In our work, we only consider implicit arguments within one tweet.

[5] Liu et al. (2011a) report similar observations for verbal predicates in tweets.

[6] We only label head words as arguments.

[7] The distance between two words is proportional to the number of words between them.

[8] Before comparing two strings, we transform them into their lowercase lemmas.

proaches dominate, owing to the public availability of annotated corpora such as the PropBank (Kingsbury and Palmer 2003), and the CoNLL shared tasks (Carreras and Màrquez 2005; Surdeanu et al. 2008). The pipelined approach is a standard data driven approach, which divides the task into several successive components such as argument identification, argument classification, global inference, etc., and conquers them individually (Xue 2004; Koomen et al. 2005; Cohn and Blunsom 2005; Punyakanok, Roth, and Yih 2008; Toutanova, Haghighi, and Manning 2005; 2008) .

There is also a large body of other statistic learning based methods. For example, Màrquez et al. (2005) sequentially label the words according to their positions relative to an argument (i.e., inside, outside, or at the beginning); Vickrey and Koller (2008) simplify the input sentence by hand-written and machine learnt rules before SRL; some other approaches resolve all the sub-tasks simultaneously by integrating syntactic parsing and SRL into a single model (Musillo and Merlo 2006; Merlo and Musillo 2008), or by using Markov Logic Networks (MLN) (Richardson and Domingos 2006) as the learning framework (Meza-Ruiz and Riedel 2009).

All of the above studies target formal texts. Recently, a handful of work on tweets has emerged. Liu et al. (2010) first study verbal SRL for tweets. They map predicate-argument structures from news sentences to news tweets (tweets that report news) to obtain training data, based on which a tweet specific system is trained. A linear CRF model is used to integrate conventional features such as lemma and POS. Liu et al. in their recent work (2011a) study the information insufficiency challenge for tweets, and propose clustering similar tweets, and collectively performing SRL on a group of similar tweets. They use two-stage labeling rather than a graphical model, to leverage cross tweet information.

Nominal SRL

Compared with verbal SRL, nominal SRL has been less well studied. Existing methods are mainly based on FrameNet and NomBank. Gildea and Jurafsky (2002b) present an early FrameNet-based SRL system that targets both verbal and nominal predicates. Jiang and Ng (2006) first report an automatic NomBank SRL system, which uses maximum entropy as its classification model. Gerber and Chai (2009) first study the role of implicit argumentation in NSRL, and develop a system that takes implicit argumentation into account, improving overall performance by nearly 5% F1. In their later work (Gerber and Chai 2010), they demonstrate the feasibility of recovering implicit arguments with a supervised classification model. Li et al. (2009) explore Chinese nominal SRL on automatic parse trees with automatic predicate recognition and successfully integrate features derived from Chinese verbal SRL into Chinese nominal SRL with much performance improvement. Srikumar and Roth (2011) present a joint inference model that captures the inter-dependencies between verbal SRL and relations expressed using prepositions, with the goal of extending SRL from verbs and nominals to other kinds of predicates.

Two factors differentiate our method from the above studies. First, we study nominal SRL in the context of tweets, a new genre of texts, which are short and prone to noise. That means conventionally useful features on formal texts are not available on tweets. Second, we adopt a novel graphical model (Koller and Friedman 2009) to jointly resolve the arguments (including implicit arguments), which harvests the redundancy in similar tweets to overcome the limited information in a single tweet.

Task Definition

A tweet is a short text message with no more than 140 characters shared through Twitter, the most popular microblog service, where users use tweets to discuss any topic, report whatever is happening, and communicate with each other. Here is an example of a tweet: "mycraftingworld: #Win Microsoft Office 2010 Home and Student *2Winners* #Contest from @office and @momtobedby8 #Giveaway http://bit.ly/bCsLOr ends 11/14", where "mycraftingworld" is the name of the user who posted this tweet. Words beginning with the "#" character, like ""#Win", "#Contest" and "#Giveaway", are hash tags, usually indicating the topics of the tweet; words starting with "@", like "@office" and "@momtobedby8", represent user names, and "http://bit.ly/bCsLOr" is a shortened link.

Given a set of tweets $T = \{t\}$ as input, our task is to identify every nominal predicate, and for each nominal predicate further identify its semantic arguments. We use the general role schema defined by NomBank, which includes core roles such as arg_0, arg_1 (usually indicating the agent and patient of the predicate, respectively), and auxiliary roles such as arg_{tmp} and arg_{loc} (representing the temporal and location information of the predicate, respectively). Following Màrquez et al. (2005), we label only the head as the the argument. As a pilot study, we limit our scope to English tweets [9].

Here is an example to illustrate our task. For the following input tweet "the death of #stevejobs says something significant about america's decline$\cdots$". The expected output is a set of triples: $\{$ (death,#stevejobs,arg_1),(decline,america,arg_0)$\}$, which says that "#stevejobs" is the patient of "death", and that "america" is the agent of "decline".

Our Method

We first give an overview of our method then discuss its two core components: 1) Nominal predicate identification model; and 2) argument identification and classification model. We concentrate our focus on #2, and discuss its training, inference, and features in detail.

Overview

The task of NSRL is divided into two sub problems: 1) Nominal predicate identification and 2) argument identification and classification. To address #1, we follow Li et

[9] Extending our method to other languages, such as Chinese, requires updating the word breaker and the POS tagger.

al. (2009), and train a binary classifier using the LibSVM toolkit [10] with default settings. As opposed to existing methods, such as Li et al. (2009), that use parsing related features, we adopt shallow features while deciding whether t_m^i, the i^{th} word in tweet t_m, is a nominal predicate, including: 1) POS/lemma of the word before/after t_m^i; 2) neighboring words of t_m^i in a text window of size three; 3) the nearest verb to t_m^i and its position (left or right) and 4) whether t_m^i is in the predefined nominal word list [11]. For every input tweet, we first use a POS tagger [12] to extract all nouns, each of which is then fed into the classifier to check whether it is a nominal predicate or not. In this way, we obtain all nominal predicates, denoted by P.

To address #2, we build a factor graph $\mathcal{G}_p$ for each nominal predicate $p \in P$, using all tweets containing the predicate, denoted by T_p. $\mathcal{G}_p$ is formally defined as $\mathcal{G}_p = (Y_p, F_p, E_p)$, where: $Y_p = \{y_m^i\}_{m,i}$ represents the semantic role variables and y_m^i represents the semantic role [13] of the i^{th} word in the m^{th} tweet in T_p for the predicate p; F_p stands for factor vertices, consisting of $\{f_m^i(y_m^{i-1}, y_m^i)\}$ and $\{f_{mn}^{ij}(y_m^i, y_n^j)\}, \forall t_m^i = t_n^j$, and t_m and t_n are similar [14]; E_p represents edges, consisting of edges between y_m^{i-1} and f_m^i, edges between y_m^i and f_m^i, edges between y_m^i and f_{mn}^{ij}, and edges between y_n^j and f_{mn}^{ij}.

$\mathcal{G}_p = (Y_p, F_p, E_p)$ then defines a probability distribution $P(Y_p|\mathcal{G}_p, T_p)$ according to Formula 1.

$$\begin{aligned}\ln P(Y_p|\mathcal{G}_p, T_p) = &- \ln Z(\mathcal{G}_p, T_p) + \\ &\sum_{m,i} \ln f_m^i(y_m^{i-1}, y_m^i, p) + \\ &\sum_{m,n,i,j} \delta_{mn}^{ij} \cdot \ln f_{mn}^{ij}(y_m^i, y_n^j, p)\end{aligned} \tag{1}$$

where $\delta_{mn}^{ij} = 1$ if and only if: $t_m^i = t_n^j$, and t_m and t_n are similar; otherwise zero; $Z(\mathcal{G}_p, T_p)$ is the partition function as defined in Formula 2.

$$\begin{aligned}Z(\mathcal{G}, T) = &\sum_{Y} \prod_{m,i} f_m^i(y_m^i) \cdot \\ &\prod_{m,n,i,j} f_{mn}^{ij}(y_m^i, y_n^j)^{\delta_{mn}^{ij}}\end{aligned} \tag{2}$$

A factor factorizes according to a set of features, as defined in Formula 3.

$$\begin{aligned}\ln f_m^i(y_m^i) &= \sum_k \lambda_k^{(1)} \phi_k^{(1)}(y_m^{i-1}, y_m^i, p) \\ \ln f_{mn}^{ij}(y_m^i, y_n^j) &= \sum_k \lambda_k^{(2)} \phi_k^{(2)}(y_m^i, y_n^j, p)\end{aligned} \tag{3}$$

$\{\phi_k^{(1)}\}_{k=1}^{K_1}$ and $\{\phi_k^{(2)}\}_{k=1}^{K_2}$ are two feature sets. Each feature has a real value as its weight. $\Theta = \{\lambda_k^{(1)}\}_{k=1}^{K_1} \bigcup \{\lambda_k^{(2)}\}_{k=1}^{K_2}$ denotes the feature weight set, which is also called parameters of $\mathcal{G}_p$. Note that $\forall p \in P$, $\mathcal{G}_p$ shares the same set of model parameters.

Our method jointly decides the values of Y_p. With $\forall y_m^i \in Y_p$ resolved, outputting the predicate-argument structures for predicate p in t_m, denoted by S_m^p, is straightforward:

$$S_m^p = \{(p, t_m^i, y_m^i) | \forall i, y_m^i \neq o\} \tag{4}$$

Training

Given a set of training data T, in which every nominal predicate is annotated, and for each predicate in each tweet, its semantic roles are also annotated, Θ is learnt by maximizing the log data likelihood, as defined in Formula 5.

$$\Theta^* = \arg\max_{\Theta} \sum_p \ln P(Y_p|\Theta, T_p) \tag{5}$$

To solve this optimization problem, we first calculate its gradient:

$$\begin{aligned}\frac{\partial \sum_p \ln P(Y_p|T_p;\Theta)}{\partial \lambda_k^1} = &\sum_p \sum_{m,i} \phi_k^{(1)}(y_m^{i-1}, y_m^i, p) - \\ &\sum_p \sum_{m,i} \sum_{y_m^i} p(y_m^{i-1}, y_m^i|T_p;\Theta) \phi_k^{(1)}(y_m^{i-1}, y_m^i, p)\end{aligned} \tag{6}$$

$$\begin{aligned}\frac{\partial \sum_p \ln P(Y_p|T_p;\Theta)}{\partial \lambda_k^2} = &\sum_p \sum_{m,n,i,j} \delta_{mn}^{ij} \cdot \phi_k^{(2)}(y_m^i, y_n^j, p) \\ -&\sum_p \sum_{m,n,i,j} \delta_{mn}^{ij} \sum_{y_m^i, y_n^j} p(y_m^i, y_n^j|T_p;\Theta) \cdot \phi_k^{(2)}(y_m^i, y_n^j, p)\end{aligned} \tag{7}$$

where, the two marginal probabilities $p(y_m^{i-1}, y_m^i|T_p;\Theta)$ and $p(y_m^i, y_n^j|T_p;\Theta)$ are estimated using the loopy belief propagation algorithm (Murphy, Weiss, and Jordan 1999). Once we have computed the gradient, Θ^* can be figured out by standard techniques such as steepest descent, conjugate gradient, and the limited-memory BFGS algorithm (L-BFGS). L-BFGS is adopted because it is particularly well-suited for optimization problems with a large number of variables, which is exactly the case in our work.

Inference

Given a set of tweets T for testing, we first extract every nominal predicate, and get the predicate set P. For each predicate p we can construct a factor graph $\mathcal{G}_p$. Supposing the model parameters Θ are fixed to Θ^* , the inference problem is to find the most possible assignment of Y_p, i.e.,

$$Y_p^* = \arg\max_{Y_p} \ln P(Y_p|\Theta^*, T_p) \tag{8}$$

We adopt the max-product algorithm to solve this inference problem. The max-product algorithm is nearly identical to the loopy belief propagation algorithm, except that the sums are replaced by maxima in the definitions. Iterating all p in P, we can output predicate-argument structures for all the testing tweets.

[10] http://www.csie.ntu.edu.tw/ cjlin/libsvm/

[11] We compile a list of nominal predicates from NomBank.

[12] We use the POS tagger fine tuned for tweets (Ritter et al. 2011)

[13] We introduce a special role type "*o*", meaning the word does not play any role for the predicate.

[14] We use case insensitive comparison. For example, "release" and "Release" are regarded as equal.

Features

Features $\{\phi_k^{(1)}(y_m^{i-1}, y_m^i, p)\}_{k=1}^{K_1}$ consist of local features and global features. Local features are related to tweet t_m, including: 1) Lemma/POS of t_m^i/t_m^{i-1}; 2) # of words between p and t_m^i; 3) whether t_m^i is on the left or right of p; and 4) y_m^{i-1} and y_m^i. Global features are statistics collected from the whole corpus (training and testing data set), including: 1) Co-occurrence times of t_m^i and p in the same tweet/sentence; and 2) co-occurrence times of t_m^i and p in the same text window of size three.

Features $\{\phi_k^{(2)}(y_m^i, y_n^j, p)\}_{k=1}^{K_2}$ include: 1) The similarity between t_m and t_n, as defined in Formula 9, where $\vec{t}$ is the bag-of-words vector representation of t with stop words removed. The stop words are mainly from http://www.textfixer.com/resources/common-english-words.txt; 2) whether t_m and t_n fall into the same time period e.g., half a day; 3) whether t_m and t_n have one common hash tag/verb; 4) whether a re-tweet/reply relationship exists between t_m and t_n; 5) whether t_m and t_n contain the same link; and 6) whether t_m^i and t_n^j have some common neighboring word in a size three text window.

$$sim(t_m, t_n) = \frac{\vec{t_m} \cdot \vec{t_n}}{|\vec{t_m}||\vec{t_n}|} \tag{9}$$

Note that a feature with a real value r is mapped to a binary feature with value 1 if and only if $P_{\text{norm}}(x > r) \leq 0.2$. Here we assume a normal distribution of $P_{\text{norm}}(\cdot|\hat{\mu}, \hat{\sigma}^2)$ for any real value feature, as defined in Formula 10.

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} r_i, \hat{\sigma}^2 = \frac{1}{n-1}\sum_{i=1}^{n}(r_i - \hat{\mu})^2 \tag{10}$$

Experiments

In this section, we evaluate our method on a manually annotated data set, and show that our system outperforms the baselines.

Data Preparation

We use the Twitter API to crawl tweets from October 5^{th}, 2011 to October 12^{th}, 2011. After dropping non-English tweets, we get 17,421,446 tweets, from which 7,000 tweets are randomly sampled. The selected tweets are then labeled by two annotators following the annotation guidelines for NomBank, with one exception: For phrasal arguments, to be consistent with the word level labeling system, only the head word is labeled as the argument. The inter-rater agreement measured by kappa is 0.67. Each inconsistently annotated case is discussed by the two annotators to reach a consensus. In total, we get 5,867 tweets containing at least one nominal predicate, and 1,133 without any nominal predicate. On average, a tweet has 1.05 nominal predicates, and a nominal predicate has 1.3 arguments. arg_0 and arg_1 account for 47.3% and 41.7% of all arguments, respectively; arg_2, arg_{tmp}, and arg_{loc} combined represent 9.5%; and others constitute the remaining. 15.8% arguments are implicit, among which 49.5% and 45.8% are of type arg_0 and arg_1, respectively. 800 tweets are randomly chosen for development and the remainder are used for 10-fold cross-validation.

Evaluation Metrics

We adopt the widely used Precision, Recall and F1 as the evaluation metrics. Precision tells us what percentage of the predicted labels are correct, and recall is a measure of what percentage of the labels in the gold-standard data set are correctly identified, while F1 is the harmonic mean of precision and recall.

Baseline

Two baseline systems are developed. One is the SVM based system (Gerber, Chai, and Bart 2011) (SRL_S), the state-of-the-art system trained on the Nombank; the other is the verbal SRL system tuned for tweets (Liu et al. 2011a), denoted by SRL_T. Following Liu et al. (2011a), we use OpenNLP and the Stanford parser to extract conventional features for the baselines. Note that all systems are re-trained on our annotated data set, and run 10-fold cross-validation. To concentrate our focus on argument identification and classification, the nominal predicate identification module we developed is used for all systems [15].

Result

Table 1 shows the Precision, Recall and F1 of the baselines and ours (SRL_G). It can be seen that our system performs better than SRL_T, with an absolute F1 gain of 2.7% ($p < 0.04$). This suggests that the graphical model based joint inference is more efficient than the clustering-based two-stage labeling. It can also be seen that out system significantly outperforms SRL_S ($p < 0.005$). Be reminded that SRL_S works on a single tweet and adopts linguistically motivated features, so this reaffirms the challenge of NSRL for tweets, and indicates the need to collectively consider multiple tweets.

As a case study, consider the following two tweets: 1) "[arg_0 #Tigers]! Thanks sportscenter for letting us know how lebron is feeling about the [arg_1 yankees] [$pred$ elimination]." ; and 2) "[arg_0 #Tigers] deserve #WorldSeries title now for [$pred$ elimination] the[arg_1 Yankees] from the playoffs!". The baselines incorrectly recognize "lebron" as the arg_0 of "elimination" for the first tweet, though it correctly identify "Tigers" as the "arg_0" of "elimination" for the second tweet. In contrast, our system successfully outputs ([$pred$ elimination],[arg_0 #Tigers]) for the first tweet, since it leverages the cross tweet evidence that ([$pred$ elimination],[arg_0 #Tigers]) occurs in a recent and similar tweet.

Table 2 shows the Precision, Recall and F1 of our system and the baselines for implicit arguments. As can be seen, our system yields considerably better F1 than the baselines ($p < 0.02$). This suggests the effectiveness of our method in resolving implicit arguments. We also see that the overall F1 for implicit arguments is lower than that for explicit

[15] Our nominal predicate model achieves an F1 of 71.5%, as compared to 66.0% of Gerber and Chai's system (2011).

System	Pre.	Rec.	F1
SRL_G	61.0	60.0	60.5
SRL_T	60.9	54.7	57.8
SRL_S	48.7	49.5	49.1

Table 1: Overall 10-fold cross-validation results for all arguments (%).

System	Pre.	Rec.	F1
SRL_G	56.5	45.8	50.6
SRL_T	31.5	39.6	35.1
SRL_S	26.2	38.1	31.0

Table 2: Overall 10-fold cross-validation results for implicit arguments(%).

arguments for all systems. This indicates the difficulty of implicit argument resolution.

Table 3 reports the Precision, Recall and F1 of our system for different types of arguments. As can be seen, our system performs best for arg_0, followed by arg_1 and others, which is consistent with their distributions in the gold-standard data set.

Finally, to study the contributions of jointly learning on multiple similar tweets, we modify our method to remove $\{\phi_k^{(2)}\}$ features. The modified system is similar to SRL_S except that it uses sequential labeling rather than classification, and that it adopts shallow features plus global features instead of advanced linguistic features. It achieves a Precision, Recall and F1 of 56.2%, 52.4% and 54.2%, respectively, outperforming the SRL_S while remarkably lagging behind our system. This confirms the positive influence of collective inference on multiple tweets.

Discussion

A great portion of the errors made by our system are caused by the false positive and false negative errors [16] propagated from the nominal predicate identification model, accounting for 31.5% and 25.6% of all errors, respectively. As an illustrative example, consider the tweet: "$\cdots$rest in peace steve jobs$\cdots$". "rest" is incorrectly identified as a nominal predicate (false positive); and "peace" and "steve jobs" are recognized as its arg_{loc} and arg_0, respectively. In other cases, nominal predicates are not recognized (false negative). For example, for this tweet, "160 photos of #occupywallstreet today in nyc - glorious summer day", "photos" is not recognized as a nominal predicate, and as a result it is impossible for our system to identify "of" and "today" as the arg_1 and arg_{tmp} of "photos", respectively. In the future, we plan to improve the nominal predicate classifier using a joint inference model similar to the model we have developed for argument identification and classification.

Furthermore, a great number of errors, about 35.5%, are related to the fact that SRL is fundamentally a task that requires some understanding of the meaning of the tweets, which cannot be captured by shallow features. Take the following tweet for example: "$\cdots$gas prices up! but #gadhafi's death could bring decline$\cdots$", for which, "death" is incorrectly labeled as the arg_1 (instead of arg_0) of "decline". This is a reasonable mistake, considering that the semantic dependency between "death", "bring" and "decline" is ignored. A promising approach to fixing these errors is to jointly resolve verbal SRL and nominal SRL, allowing them to interact with each other.

The remaining errors mainly consist of implicit argument resolution errors, which are largely rooted in one limitation of our system, i.e., considering only the arguments of the same nominal predicate across similar tweets for resolving implicit arguments. For instance, for the tweet "$\cdots$stand with #occupywallst and demand real democracy! send a message of support to #ows here$\cdots$", our method fails to recognize "#occupywallst" as the arg_1 of "support". However, if "#occupywallst" could be identified as the arg_1 of the verbal predicate "stand with", which is relatively easy, and the connection between the two predicates "stand with" and "support" could be established, it should be more likely for our system to guess "#occupywallst" as the arg_1 of "support".

[16] A false positive error means a false nominal predicate is outputted, while a false negative error means a true nominal predicate is not identified.

Argument	Pre.	Rec.	F1
arg_0	62.8	64.5	63.6
arg_1	60.2	59.6	59.9
$others$	45.2	54.5	49.4

Table 3: 10-fold cross-validation results of our system on arg_0, arg_1 and other argument types (%).

Conclusions and Future work

We study the task of nominal SRL for tweets. There are two main challenges for this task: Limited information in a single tweet and inference of implicit arguments. We propose jointly conducting NSRL for a given nominal predicate on similar tweets, leveraging redundancy in tweets to address these challenges. We evaluate our method on a manually annotated data set, and show that it outperforms two baselines, demonstrating the effectiveness of joint learning.

As for future work, we plan to apply the same idea to nominal predicate identification: i.e., deciding whether a noun is a predicate on multiple similar tweets simultaneously. We are also interested in extending our framework to jointly perform verbal and nominal SRL for tweets .

References

Carreras, X., and Màrquez, L. 2005. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, 152–164. Ann Arbor, Michigan: Association for Computational Linguistics.

Cohn, T., and Blunsom, P. 2005. Semantic role labelling with tree conditional random fields. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, CONLL '05,

169–172. Morristown, NJ, USA: Association for Computational Linguistics.

Cortes, C., and Vapnik, V. 1995. Support-vector networks. 20(3):273–297.

Gerber, M., and Chai, J. 2010. Beyond nombank: A study of implicit arguments for nominal predicates. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 1583–1592. Uppsala, Sweden: Association for Computational Linguistics.

Gerber, M.; Chai, J. Y.; and Bart, R. 2011. A joint model of implicit arguments for nominal predicates. In *Proceeding RELMS '11 Proceedings of the ACL 2011 Workshop on Relational Models of Semantics*. Association for Computational Linguistics.

Gerber, M.; Chai, J.; and Meyers, A. 2009. The role of implicit argumentation in nominal srl. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 146–154. Boulder, Colorado: Association for Computational Linguistics.

Gildea, D., and Jurafsky, D. 2002a. Automatic labeling of semantic roles. *Comput. Linguist.* 28:245–288.

Gildea, D., and Jurafsky, D. 2002b. Automatic labeling of semantic roles. *Comput. Linguist.* 28(3):245–288.

Jiang, Z. P., and Ng, H. T. 2006. Semantic role labeling of nombank: a maximum entropy approach. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, 138–145. Stroudsburg, PA, USA: Association for Computational Linguistics.

Kingsbury, P., and Palmer, M. 2003. Propbank: The next level of treebank. In *Proceedings of Treebanks and Lexical Theories*.

Koller, D., and Friedman, N. 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Koomen, P.; Punyakanok, V.; Roth, D.; and Yih, W.-t. 2005. Generalized inference with multiple semantic role labeling systems. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, CONLL '05, 181–184. Morristown, NJ, USA: Association for Computational Linguistics.

Kwak, H.; Lee, C.; Park, H.; and Moon, S. 2010. What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, WWW '10, 591–600. New York, NY, USA: ACM.

Li, J.; Zhou, G.; Zhao, H.; Zhu, Q.; and Qian, P. 2009. Improving nominal srl in chinese language with verbal srl information and automatic predicate recognition. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3*, EMNLP '09, 1280–1288. Stroudsburg, PA, USA: Association for Computational Linguistics.

Liu, X.; Li, K.; Han, B.; Zhou, M.; Jiang, L.; Xiong, Z.; and Huang, C. 2010. Semantic role labeling for news tweets. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, 698–706. Beijing, China: Coling 2010 Organizing Committee.

Liu, X.; Li, K.; Zhou, M.; and Xiong, Z. 2011a. Collective semantic role labeling for tweets with clustering. In *IJCAI*, 1832–1837.

Liu, X.; Li, K.; Zhou, M.; and Xiong, Z. 2011b. Enhancing semantic role labeling for tweets using self-training. In *AAAI*.

Liu, X.; Zhang, S.; Wei, F.; and Zhou, M. 2011c. Recognizing named entities in tweets. In *ACL*, 359–367.

Màrquez, L.; Comas, P.; Giménez, J.; and Català, N. 2005. Semantic role labeling as sequential tagging. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, CONLL '05, 193–196. Morristown, NJ, USA: Association for Computational Linguistics.

Merlo, P., and Musillo, G. 2008. Semantic parsing for high-precision semantic role labelling. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning*, CoNLL '08, 1–8. Morristown, NJ, USA: Association for Computational Linguistics.

Meza-Ruiz, I., and Riedel, S. 2009. Jointly identifying predicates, arguments and senses using markov logic. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, 155–163. Morristown, NJ, USA: Association for Computational Linguistics.

Murphy, K. P.; Weiss, Y.; and Jordan, M. I. 1999. Loopy belief propagation for approximate inference: An empirical study. In *In Proceedings of Uncertainty in AI*, 467–475.

Musillo, G., and Merlo, P. 2006. Accurate parsing of the proposition bank. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers on XX*, NAACL '06, 101–104. Morristown, NJ, USA: Association for Computational Linguistics.

Punyakanok, V.; Roth, D.; and Yih, W.-t. 2008. The importance of syntactic parsing and inference in semantic role labeling. *Comput. Linguist.* 34:257–287.

Richardson, M., and Domingos, P. 2006. Markov logic networks. *Mach. Learn.* 62:107–136.

Ritter, A.; Clark, S.; Mausam; and Etzioni, O. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 1524–1534. Edinburgh, Scotland, UK.: Association for Computational Linguistics.

Srikumar, V., and Roth, D. 2011. A joint model for extended semantic role labeling. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 129–139. Edinburgh, Scotland, UK.: Association for Computational Linguistics.

Surdeanu, M.; Johansson, R.; Meyers, A.; Màrquez, L.; and Nivre, J. 2008. The conll-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning*, CoNLL '08, 159–177. Morristown, NJ, USA: Association for Computational Linguistics.

Toutanova, K.; Haghighi, A.; and Manning, C. D. 2005. Joint learning improves semantic role labeling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, 589–596. Morristown, NJ, USA: Association for Computational Linguistics.

Toutanova, K.; Haghighi, A.; and Manning, C. D. 2008. A global joint model for semantic role labeling. *Comput. Linguist.* 34:161–191.

Vickrey, D., and Koller, D. 2008. Applying sentence simplification to the conll-2008 shared task. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning*, CoNLL '08, 268–272. Morristown, NJ, USA: Association for Computational Linguistics.

Xue, N. 2004. Calibrating features for semantic role labeling. In *In Proceedings of EMNLP 2004*, 88–94.