An SVM based learning algorithm for Information Extraction

Yaoyong Li, Kalina Bontcheva, Hamish Cunningham

Department of Computer Science, University of Sheffield

{yaoyong, kalina, hamish}@dcs.shef.ac.uk

STRUCTURE

- Information Extraction and Machine Learning
- An SVM Based Learning Algorithm
- Experimental Results

Machine Learning Workshop, Sheffield, Sept. 2004

Information Extraction (IE)

- IE is about extracting information about pre-specified types of events, entities or relationships from text such as newswire articles or Web pages.
  - Named entity recognition is a basic task for IE.
  - Some IE tasks can be reduced to specify a list of slots in an information template.

- IE can be useful in many applications.
  - Information gathering in a variety of domains
  - Automatic annotations of web pages for semantic web
  - Knowledge management

- Two important events for IE – Message Understanding Conferences (MUCs) and Automatic Content Extraction programme (ACE)
  - MUCs (1991–1998): Developed methods for formal evaluation of IE systems.
  - ACE (1999–): A successor of MUC, has more complex tasks than MUC.

Machine Learning for IE

- Machine learning has been widely used in IE and achieve state of the art results for some tasks.
  - Rule based learning,
    - ∗ Learning rules for extracting information
    - ∗ such as Rapier, BWI, $(\text{LP})^2$
  - Statistical machine learning
    - ∗ Learning classifiers
    - ∗ such as Maximum Entropy, HMM, SVM

- Support Vector Machine (SVM) is a state of the art machine learning algorithm. There are several SVM based systems for named entity recognition.
  - Mayfield etc. (2003) learned an SVM for every plausible transition of named entities, and results on CoNLL-2003 shared task showed it is a competitive systems.
  - Isozaki and Kazawa (2002) learned four SVMs for one entity type. It showed that SVM based system was better than both HMM based and rule based systems.

An SVM Based Learning Algorithm

- Learning two SVM classifiers for every entity type
  - One for start word, another for end word
  - We used less SVM classifiers than both [Isozaki02] and [Mayfield03]

- A variant of the SVM, the SVM with uneven margins, was used

$$
\begin{array}{lll}
\text{minimise}_{\mathbf{w},\ b,\ \xi} & \langle \mathbf{w}, \mathbf{w} \rangle + C \sum_{i=1}^{m} \xi_i & \\
\text{subject to} & \langle \mathbf{w}, \mathbf{x}_i \rangle + \xi_i + b \geq 1 & \text{if} \;\; y_i = +1 \\
 & \langle \mathbf{w}, \mathbf{x}_i \rangle - \xi_i + b \leq -\tau & \text{if} \;\; y_i = -1 \\
 & \xi_i \geq 0 \qquad \text{for} \;\; i = 1, ..., m &
\end{array}
\tag{1}
$$

An SVM Based Learning Algorithm (2)

- The feature vector of a word is based on the word itself as well as its neighbouring words. We used two schemes of weighting context words in feature vector
  - Equally weighting.
  - Reciprocal weight $(1/j)$.

- Three steps of post-processing procedure to identity entities from the SVM results.
  - Remove the spurious start or end tags for one named entity type from a document to keep the start and end tags consistency.
  - Filter out the entity candidate which is too short or too long.
  - Consider all the possible tags for a piece of text and assign it the type of named entity with the highest probability.

The Datasets Used in Experiments

- The corpus of the CoNLL-2003 shared task
  - Four types of named entities: persons, locations, organisation, and names of miscellaneous entities.
  - Conventional named entity recognition task.

- CMU seminars corpus
  - 485 seminar announcements. Four types of information entities about seminar: start time, end time, speaker, and location.
  - Information extraction task: request to fill in the 4 slots of an information template about seminar.

- Computer-related job advertisements corpus
  - 300 job adverts. 17 types of entities such as job's title, salary, computer language(s), and applications, etc.
  - Also request to fill in 17 plots of an information template a bout computer-related job.

Experimental Results

- Compare two weighting schemes for feature vector

Table 1: Two weighting schemes: macro-averaged F_1 on three datasets.

Datasets	Equally weighting	$1/j$ weighting
Seminar	0.860	0.868
Job	0.787	0.778
CoNLL03	0.883	0.890

- Compare three post-processing procedures

Table 2: Three post-processing procedures: macro-averaged F_1 on three datasets.

Dataset	Proc1	Proc2	Proc3
Seminar	0.8676	0.8697	0.8704
Job	0.7875	0.7891	0.7910
CoNLL03	0.8895	0.8912	0.8918

Experimental Results (2)

Compare our algorithm with other systems.

Table 3: Comparisons on CoNLL-2003 dataset.

System	LOC	MISC	ORG	PER	Overall
Ours	0.8925	0.7779	0.8229	0.9092	0.8630
Best one	0.9115	0.8044	0.8467	0.9385	0.8876
Another SVM	0.8877	0.7419	0.7900	0.9067	0.8467

Table 4: Comparisons on CMU seminar corpus:

	Speaker	Location	Stime	Etime	Macro F_1
Ours	0.671	0.792	0.951	0.932	0.836 ± 0.032
$(LP)^2$	0.776	0.750	0.990	0.955	0.868
BWI	0.677	0.767	0.996	0.939	0.845
HMM	0.766	0.786	0.985	0.621	0.790
SRV	0.563	0.723	0.985	0.779	0.763
Rapier	0.530	0.727	0.934	0.962	0.788
Whisk	0.183	0.664	0.926	0.860	0.658

Table 5: Comparisons on the jobs corpus.

Slot	Ours	$(LP)^2$	Rapier
id	0.975	1.000	0.975
title	0.496	0.439	0.405
company	0.782	0.719	0.695
salary	0.790	0.628	0.674
recruiter	0.778	0.806	0.684
state	0.930	0.847	0.902
city	0.951	0.930	0.904
country	0.968	0.810	0.932
language	0.842	0.910	0.806
Platform	0.762	0.805	0.725
Application	0.649	0.784	0.693
Area	0.468	0.669	0.424
Req-years-e	0.816	0.688	0.671
Des-years-e	0.846	0.604	0.875
Req-degree	0.838	0.847	0.815
des-degree	0.509	0.651	0.722
post date	0.994	0.995	0.995
Macro-averaged F_1	0.788± 0.063	0.772	0.758

Adaptive Information Extration

The following table shows the performance of the algorithm when more and more documents were available for learning. It also shows that the SVM with uneven margins is better than the conventional SVM if only a small number of training documents are used.

Table 6: Different numbers of documents for training: macro-averaged F_1 on three datasets. The SVM with uneven margins were compared with the conventional SVM.

Dataset	10	20	30	40
	$\tau = 0.4$:			
Seminar	0.450	0.677	0.704	0.731
Job	0.434	0.489	0.518	0.540
CoNLL03	0.606	0.664	0.704	0.722
	$\tau = 1$:			
Seminar	0.377	0.485	0.572	0.621
Job	0.360	0.416	0.457	0.475
CoNLL03	0.462	0.586	0.652	0.683

Conclusions

- Our SVM based algorithm obtained promising results in three datasets for IE.
- Our algorithm is simpler than other two SVM based algorithms. It still achieved better result than [Mayfield03].
- We proposed and tested two schemes of weighting context words and three post-processing procedures in the algorithm.