Volume 43, issue 4, July 2007

ISSN 0306-4573

INFORMATION PROCESSING & MANAGEMENT

An International Journal

Email: ipm@scils.rutgers.edu
Web site: http://www.elsevier.com/locate/infoproman

0306-4573(200707)43:4;1-U

Available online at www.sciencedirect.com

ScienceDirect

Information Processing and Management 43 (2007) 969–982

INFORMATION PROCESSING & MANAGEMENT

www.elsevier.com/locate/infoproman

Extracting relation information from text documents by exploring various types of knowledge ☆

Zhou GuoDong [a,b,*], Zhang Min [b]

[a] School of Computer Science and Technology, Suzhou Univ., 1 ShiZi Street, SuZhou, JiangSu 215006, China
[b] Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613, Singapore

Received 6 June 2006; received in revised form 14 September 2006; accepted 20 September 2006
Available online 2 November 2006

Abstract

Extracting semantic relationships between entities from text documents is challenging in information extraction and important for deep information processing and management. This paper investigates the incorporation of diverse lexical, syntactic and semantic knowledge in feature-based relation extraction using support vector machines. Our study illustrates that the base phrase chunking information is very effective for relation extraction and contributes to most of the performance improvement from syntactic aspect while current commonly used features from full parsing give limited further enhancement. This suggests that most of useful information in full parse trees for relation extraction is shallow and can be captured by chunking. This indicates that a cheap and robust solution in relation extraction can be achieved without decreasing too much in performance. We also demonstrate how semantic information such as WordNet, can be used in feature-based relation extraction to further improve the performance. Evaluation on the ACE benchmark corpora shows that effective incorporation of diverse features enables our system outperform previously best-reported systems. It also shows that our feature-based system significantly outperforms tree kernel-based systems. This suggests that current tree kernels fail to effectively explore structured syntactic information in relation extraction.

Keywords: Information extraction; Feature-based relation extraction; Knowledge exploration; Support vector machines

1. Introduction

With the dramatic increase in the amount of textual information available in digital archives and the WWW, there has been growing interest in techniques for automatically extracting information from text documents. Information extraction (IE) is such a technology that IE systems are expected to identify relevant

☆ A previous version of this paper has been published in ACL'2005 (Zhou, Su, Zhang, & Zhang, 2005).

[*] Corresponding author. Address: School of Computer Science and Technology, Suzhou Univ., 1 ShiZi Street, SuZhou, JiangSu 215006, China. Tel./fax: +86 512 65113205.

E-mail addresses: gdzhou@suda.edu.cn, zhougd@i2r.a-star.edu.sg (G. Zhou), mzhang@i2r.a-star.edu.sg (M. Zhang).

0306-4573/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2006.09.012

information (usually of pre-defined types) from text documents in a certain domain and put them in a structured format.

According to the scope of the NIST Automatic Content Extraction (ACE) program (ACE, 2000–2005), current research in IE has three main objectives: Entity Detection and Tracking (EDT), Relation Detection and Characterization (RDC), and Event Detection and Characterization (EDC). The EDT task entails the detection of entity mentions and chaining them together by identifying their coreference relationships. In ACE vocabulary, entities are objects, mentions are references to them, and relations are semantic relationships between entities. For example, the ACE EDT 2003 task de-fries five entity types, i.e. persons, organizations, locations, facilities and geo-political entities (GPE: geographically defined regions that indicate a political boundary, e.g. countries, states, cities, etc.) and have three levels of entity mentions, i.e. names, nomial expressions and pronouns. The RDC task detects and classifies implicit and explicit relations[1] between entities identified by the EDT task. For example, we want to determine whether a person is at a location, based on the evidence in the context-Extraction of semantic relationships between entities can be very useful for applications such as question answering, e.g. to answer the query "Who is the president of the United States?", and information retrieval, e.g. to expand the query "George W. Bush" with "the president of the United States" via his relationships with the country "the United States".

This paper focuses on the ACE RDC task and employs diverse lexical, syntactic and semantic knowledge in feature-based relation extraction using support vector machines (SVM). Our study illustrates that the base phrase chunking information contributes to most of the performance improvement from syntactic aspect while current commonly used features from full parsing do not contribute much, largely due to the fact that most of relations defined in the ACE corpora are within a very short distance. We also demonstrate how semantic information such as WordNet (Miller, 1990) can be used in the feature-based framework. Evaluation on the ACE benchmark corpora shows that effective incorporation of diverse features enables our system outperform previously best-reported systems. It also shows that our feature-based approach significantly outperforms tree kernel-based approaches.

The rest of this paper is organized as follows. First, Section 2 presents related work. Then, a brief introduction about feature-based relation extraction is given in Section 3 while various features are described in Section 4. Finally, we present experimental setting and results in Section 5 and conclude with some general observations and remarks in relation extraction in Section 6.

2. Related work

The relation extraction task was formulated as a critical part of information extraction at the 7th Message Understanding Conference (MUC-7, 1998) and is starting to be addressed more and more within the natural language processing and machine learning communities. Representative related works can be classified into three categories according to different approaches they used: generative models (Miller, Fox, Ramshaw, & Weischedel, 2000), tree kernel-based approaches (Bunescu & Mooney, 2005; Culotta & Sorensen, 2004; Zelenko, Aone, & Rochardella, 2003; Zhang, Su, Wang, Zhou, & Tan, 2005), and feature-based approaches (Roth & Yih, 2002; Kambhatla, 2004 & Zhao & Grisman, 2005).[2]

Miller et al. (2000) augmented syntactic full parse trees with semantic information corresponding to entities and relations, and built generative models to integrate various tasks such as POS tagging, named entity recognition, template element extraction and relation extraction. The problem is that such integration may impose big challenges such as the need of a large scale annotated corpus. Generative models typically apply some smoothing techniques to integrate different scales of contexts in parameter estimation, e.g. the back-off approach in Miller et al. (2000).

Zelenko et al. (2003) proposed extracting relations by computing kernel functions between parse trees. Culotta and Sorensen (2004) extended this work to estimate kernel functions between augmented dependency

[1] In ACE (http://www.ldc.upeJni.edu/Projects/ACE). explicit relations occur in text with explicit evidence suggesting the relationships. Implicit relations need not have explicit supporting evidence in text, though they should be evident from a reading of the document.

[2] Here, we classify this paper into feature-based approaches since the feature space in the kernels of Zhao and Grisman (2005) can be easily represented by an explicit feature vector.

trees and achieved the F-measure of 45.8 on the 5 relation types in the ACE RDC 2003 corpus.[3] Bunescu and Mooney (2005) proposed a shortest path dependency kernel. They argued that the information to model a relationship between two entities can be typically captured by the shortest path between them in the dependency graph. It achieved the F-measure of 52.5 on the 5 relation types in the ACE RDC 2003 corpus. Zhang et al. (2005) adopted clustering algorithms in unsupervised relation extraction using tree kernels. Various scales of sub-trees are normally applied in the tree kernel computation.

Comparably, feature-based approaches achieved much success recently. Roth and Yih (2002) used the SNoW classifier to incorporate various features such as word, part-of-speech and semantic information from WordNet, and proposed a probabilistic reasoning approach to integrate named entity recognition and relation extraction. Kambhatla (2004) employed maximum entropy models with features derived from word, entity type, mention level, overlap, dependency tree, parse tree, and achieved the F-measure of 52.8 on the 24 relation subtypes in the ACE RDC 2003 corpus. Zhao and Grisman (2005) combined various kinds of knowledge from tokenization, sentence parsing and deep dependency analysis through support vector machines and achieved the F-measure of 70.1 on the 7 relation types of the ACE RDC 2004 corpus.[4] Feature-based approaches normally incorporate various scales of contexts into the feature vector extensively. These approaches then depend on adopted learning algorithms to weight and combine each feature effectively. For example, an exponential model and a linear model are applied in the maximum entropy models and support vector machines (in the feature-based framework) respectively to combine each feature via the learned weight vector.

Tree kernel-based approaches, such as the ones proposed by Zelenko et al. (2003), Culotta and Sorensen (2004) and Bunescu and Mooney (2005), are able to explore the implicit feature space without much feature engineering. Yet further research work is still expected to make it effective with complicated relation extraction tasks such as the one defined in ACE. Complicated relation extraction tasks may also impose a big challenge to the generative modeling approaches, such as the one used by Miller et al. (2000) which integrates various tasks (including part-of-speech tagging, named entity recognition, template element extraction and relation extraction) in a single model.

This paper will further explore the feature-based approach with a systematic study on the extensive incorporation of diverse lexical, syntactic and semantic information. Compared with others, we separately incorporate the base phrase chunking information, which contributes to most of the performance improvement from syntactic aspect. We also show how semantic information like WordNet can be equipped to further improve the performance. Evaluation on the ACE corpora shows that our system outperforms other feature-based systems. It also shows that our system significantly outperforms tree kernel-based systems.

3. Feature-based relation extraction

In this paper, relation extraction is recast as a classification problem using a machine learning approach. Just like most supervised machine learning approaches, our feature-based relation extraction relies on feature-based representation of annotated relation instances. That is, an annotated relation instance is transformed into a collection of features $f_1, f_2, \ldots, f_N$, thereby producing an N-dimension feature vector. In training, a classifier learning algorithm uses the annotated relation instances to learn a classifier while, in testing, the learned classifier is applied to input instances to determine their relation classes and thus extract possible relations.

In this paper, we select support vector machines (SVM) as the classifier since SVM represent the state-of-the-art in the machine learning research community, and there are good implementations of the algorithm available, In our implementation, we use the binary-class SVM Light[5] developed by Joachims (1998). SVM is a supervised machine learning technique motivated by the statistical learning theory (Vapnik, 1998). Based on the structural risk minimization of the statistical learning theory, SVM seeks an optimal separating

[3] The ACE RDC 2003 corpus defines 5/24 relation types/subtypes between 4 entity types.

[4] The ACE RDC 2004 corpus defines 7/27 relation types/subtypes between 7 entity types.

[5] Joachims has just released a new version of SVM Light for multi-class classification. However, this paper only uses the binary-class version. For details about SVM Light, please see http://svmlight.joachims.org/.

hyper-plane to divide the training examples into two classes and make decisions based on support vectors which are selected as the only effective instances in the training set.

Basically, SVM is a binary classifier. Therefore, we must extend SVM to multi-class (e.g. K) classification such as the ACE RDC task. For efficiency, we apply the *one vs. others* strategy, which builds K classifiers so as to separate one class from all others, instead of the *pairwise* strategy, which builds $K * (K - 1)/2$ classifiers considering all pairs of classes. The final decision of an instance in the multiple binary classification is determined by the class which has the maximal SVM output. By default, we will only apply the simple linear kernel unless otherwise specified. To evaluate the effect of different kernels in relation extraction, we will also report our performance using the polynomial kernel.

4. Features

Semantic relationship between two entities is determined from the context of their entity mentions. In addition, we distinguish the argument order of the two mentions (M1 for the first mention and M2 for the second mention), e.g. M1-Parent-Of-M2 vs. M2-Parent-Of-M1. For each pair of mentions,[6] we compute various lexical, syntactic and semantic features. Since this paper focuses on the ACE RDC task, we will explain relevant features in the context of the ACE RDC 2003 and 2004 corpora.

Fig. 1 gives an illustration of all the various types of features extracted from a relation instance of type "SOCIAL.OTHER-PERSONAL" between two entity mentions "200 domestic partners" and "their own workers" in the sentence "to provide benefits to 200 domestic partners of their own workers in New York". In this paper, eight types of features are employed. Next, we will describe each of them one by one.

4.1. Words

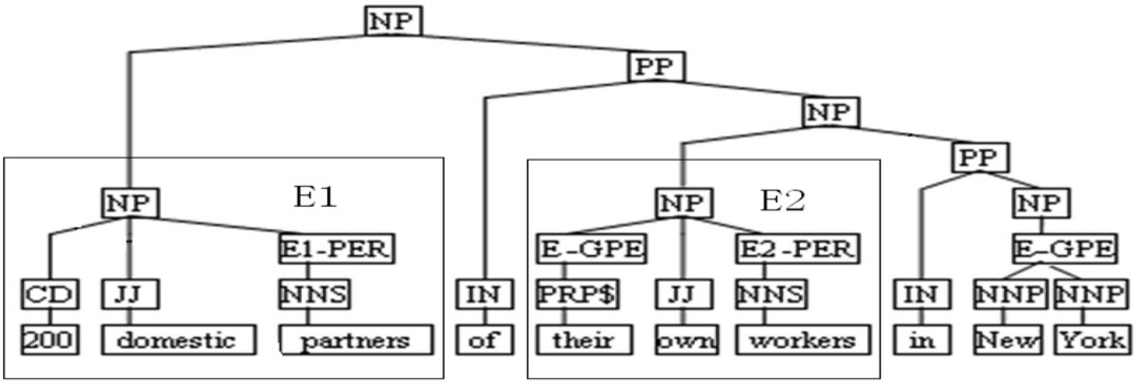
According to their positions, four categories of words are considered: (1) the words of both the mentions; (2) the words between the two mentions; (3) the words before M1; and (4) the words after M2. For the words of both the mentions, we also differentiate the head word[7] of a mention from other words since the head word is generally much more important and informative. The words between the two mentions are classified into three bins: the first word in between, the last word in between and other words in between. Both the words before M1 and after M2 are classified into two bins: the first word next to the mention and the second word next to the mention. This means that we only consider the two words before M1 and after M2. Since a pronominal mention (especially neutral pronoun such as 'it' and 'its') contains little information about the sense of the mention, the coreference chain is used to decide its sense. This is done by replacing the pronominal mention with the most recent non-pronominal antecedent when determining the word features.

In details, this category of features includes:

- WM1: bag-of-words in M1.
- HM1: head word of M1.
- WM2: bag-of-words in M2.
- HM2: head word of M2.
- HM12: combination of HM1 and HM2.
- WBNULL: when no word in between.
- WBFL: the only word in between when only one word in between.

[6] In ACE, each mention has a head annotation and an extent annotation. In all our experimentation, we only consider the word string between the beginning point of the extent annotation and the end point of the head annotation. This has a effect of choosing the base phrase contained in the extent annotation. In addition, this also can reduce noises without losing much of information in the mention. For example, in the case where the noun phrase "the former CEO of McDonald" has the head annotation of "CEO" and the extent annotation of "the former CEO of McDonald", we only consider "the former CEO" in this paper.

[7] In this paper, the head word of a mention is normally set as the last word of the mention. However, when a preposition exists in the mention, its head word is set as the last word before the preposition. For example, the head word of the name mention "University of Michigan" is "University".



(VP provide/VB (NP benefits/NNS) (PP to/IN (NP (NP 200/CD domestic/JJ partners/NNS) (PP of/IN (NP (NP their/PRP$ own/JJ workers/NNS) (PP in/IN (NP New/NNP York/NNP)))))))

(1) Full parsing

(VP provide/VB) (NP benefits/NNS) (PP to/IN) (NP 200/CD domestic/JJ partners/NNS) (PP of/IN) (NP their/PRP$ own/JJ workers/NNS) (PP in/IN) (NP New/NNP York/NNP)

(2) Base phrase chunking

1. Words: WM1_200, WM1_domestic, WM1_partners; HM1_partners; WM2_their, WM1_own, WM1_workers; HM2_workers; HM12_partners+workers; WBFL_of; BM1#1_to; BM1#2_benefits; AM2#1_in; AM2#2_New;
2. Entity Type: ET12_PER+PER;
3. Mention Level: ML12_NOMINAL+NOMINAL;
4. Overlap: #MB_0; #WB_1; M1>M2_NO; M1<M2_NO; ET12_PER+PER·M1>M2_NO; ET12_PER+PER·M1<M2_NO; HM12_partners+workers·M1>M2_NO; HM12_partners+workers ·M1<M2_NO;
5. Base Phrase Chunking: CPHBFL_of; CPHBM1#1_to; CPHBM1#2_benefits; CPHAM2#1_in; CPHAM2#2_New+York; CPP_NP+PP+NP; CPPH_ NP+PP(of)+NP;
6. Dependency Tree: ET1DW1_PER+200; ET1DW1_PER+domestic; H1DW1_partners+200; H1DW1_partners+domestic; ET2DW2_PER+their; ET2DW2_PER+own; H2DW2_workers+their; H2DW2_workers+own; ET12_PER+PER·SameNP_YES;
7. Parse Tree: PTP_NP+PP+NP; PTPH_NP(partners)+PP+NP;
8. Semantic Resources: ET1SC2_PER+NONRelative; SC1ET2_ NONRelative +PER;

(3) Various features extracted from this instance

Fig. 1. A relation instance in the sentence "…provide benefits to *200 domestic partners of their own workers* in New York", where E1 denotes that the current sub-tree is the 1st entity and "El-PER" denotes the head of the 1st entity with type "PERSON", and likewise for the others. The relation instance is excerpted from the ACE 2003 corpus, where a relation "SOCIAL.Other-Personal" exists between entities "200 domestic partners" (PER) and "their own workers" (PER).

- WBF: first word in between when at least two words in between.
- WBL: last word in between when at least two words in between.
- WBO: other words in between except first and last words when at least three words in between.
- BM1#1: first word before M1.

- BM1#2: second word before M1.
- AM2#1: first word after M2.
- AM2#2: second word after M2.

4.2. Entity type

This category of features concerns about the entity types of both the mentions, e.g. which can be PERSON, ORGANIZATION, FACILITY, LOCATION and GPE in the ACE RDC 2003 task, their entity subtypes and entity classes (defining the kind of reference an entity makes to something in the world) if available:

- ET12: combination of mention entity types.
- EST12: combination of mention entity subtypes.
- EC12: combination of mention entity classes.

4.3. Mention level

This category of features considers the entity level of both the mentions, e.g. which can be NAME, NOMIAL and PRONOUN in the ACE RDC 2003 task, and the more detailed LDC mention types if available:

- ML12: combination of mention levels.
- MT12: combination of LDC mention types.

4.4. Overlap

This category of features includes:

- #MB: number of other mentions in between.
- #WB: number of words in between.
- M1 > M2 or M1 < M2: flag indicating whether M2/M1 is included in M1/M2.

Normally, the above overlap features are too general to be effective alone. Therefore, they are also combined with other features: (1) ET12(or EST12) + M1 > M2; (2) ET12(or EST12) + M1 < M2; (3) HM12 + M1 > M2; (4) HM12 + M1 < M2.

4.5. Base phrase chunking

It is well known that chunking plays a critical role in the Template Relation task of the 7th Message Understanding Conference (MUC-7, 1998). However, the related work mentioned in Section 2 only explored the information embedded in the full parse trees. In this paper, we separate the features of base phrase chunking from those of full parsing. In this way, we can separately evaluate the contributions of base phrase chunking and full parsing. Here, the base phrase chunks are derived from full parse trees using the Perl script [8] written by Sabine Buchholz from Tilburg University and the Collins' parser (Collins, 1999) is employed for full parsing. Most of the chunking features concern about the head words of the phrases between the two mentions. Similar to word features, three categories of phrase heads are considered: (1) the phrase heads in between, which are classified into three bins: the first phrase head in between, the last phrase head in between and other phrase heads in between; (2) the phrase heads before M1, which are classified into two bins: the first phrase head before and the second phrase head before; (3) the phrase heads after M2, which are classified into two bins: the first phrase head after and the second phrase bead after. This means that we only consider the two phrases before M1 and after M2. Moreover, we also consider the phrase path in between:

[8] httpy://illk.kub.nl/~ sabine/chunklink/.

- CPHBNULL when no phrase in between.
- CPHBFL: the only phrase head when only one phrase in between.
- CPHBF: first phrase head in between when at least two phrases in between.
- CPHBL: last phrase head in between when at least two phrase heads in between.
- CPHBO: other phrase heads in between except first and last phrase heads when at least three phrases in between.
- CPHBM1#1: first phrase head before M1.
- CPHBM1#2: second phrase head before M1.
- CPHAM2#1: first phrase head after M2.
- CPHAM2#2: second phrase head after M2.
- CPP: path of phrase labels connecting the two mentions in the chunking.
- CPPH: path of phrase labels connecting the two mentions in the chunking augmented with head words of the phrases in between, if at most two phrases in between.

4.6. Dependency tree

This category of features includes information about the words, part-of-speeches and phrase labels of the words on which the mentions are dependent in the dependency tree derived from the syntactic full parse tree. The dependency tree is built by using the phrase head information returned by the Collins' parser and linking all the other fragments in a phrase to its head. It also includes flags indicating whether the two mentions are in the same NP/PP/VP.

- ET1DW1: combination of the entity type and the dependent word for M1.
- H1DW1: combination of the head word and the dependent word for M1.
- ET2DW2: combination of the entity type and the dependent word for M2.
- H2DW2: combination of the head word and the dependent word for M2.
- ET12SameNP: combination of ET12 and whether M1 and M2 included in the same NP.
- ET12SamePP: combination of ET12 and whether M1 and M2 exist in the same PP.
- ET12SameVP: combination of ET12 and whether M1 and M2 included in the same VP.

4.7. Parse tree

This category of features concerns about the information inherent only in the full parse tree:

- PTP: path of phrase labels (removing duplicates) connecting M1 and M2 in the parse tree.
- PTPH: path of phrase labels (removing duplicates) connecting M1 and M2 in the parse tree augmented with the head word of the top phrase in the path.

4.8. Semantic resources

Semantic information from various resources, such as WordNet, is used to classify important words into different semantic lists according to their indicating relationships. On the one hand, such information can be used to differentiate various relations. On the other hand, it can help resolve the data sparseness problem in relation extraction.

4.8.1. Country name list

This is mainly used to differentiate the relation subtype "ROLE.Citizen-Of' in the ACE RDC 2003 task, which defines the relationship between a person and the country of the person's citizenship, from other role relation subtypes, especially "ROLE.Residence" in the ACE RDC 2003 task, which defines the relationship between a person and the location in which the person lives. For the ACE RDC 2004 task, this country name list is not directly useful to differentiate various relations since no relation is closely related with the country

GPE (Geo-Political Entities, such as country names and provincial names). Its impact is indirect and mainly due to its effect in resolving the data sparseness problem. Two features are defined to include this information:

- ET1 Country: the entity type of M1 when M2 is a country name.
- CountryET2: the entity type of M2 when M1 is a country name.

4.8.2. Personal relative trigger word list

This is mainly used to differentiate the six personal social relation subtypes in the ACE RDC 2003 task: Parent, Grandparent, Spouse, Sibling, Other-Relative and Other-Personal. This trigger word list is first gathered from WordNet by checking whether a word has the semantic class "per-son|…| relative". Then, all the trigger words are semi-automatically[9] classified into different categories according to their related personal social relation subtypes. We also extend the list by collecting the trigger words from the head words of the mentions in the training data according to their indicating relationships. For the ACE RDC 2004 task, this list can be used to differentiate "PER-SOC.Family", which defines any familial relationship between two entities, from other personal/social relation subtypes, especially "PER-SOC.Business", which defines any professional relationship between two entities. Two features are defined to include this information:

- ET1SC2: combination of the entity type of M1 and the semantic class of M2 when M2 triggers a personal social subtype.
- SC1ET2: combination of the entity type of M2 and the semantic class of M1 when the first mention triggers a personal social subtype.

5. Experimentation

This paper mainly uses the ACE RDC 2003 corpus provided by LDC to train and evaluate our feature-based relation extraction system for both final and detailed performance. This ACE corpus is gathered from various newspapers, newswire and broadcasts. In this paper, we only model explicit relations because of poor inter-annotator agreement in the annotation of implicit relations and their limited number. Detailed evaluation has been also done on the ACE RDC 2004 corpus and shows similar tendency with the ACE RDC 2003 corpus. To avoid redundancy, we will report only the final performance on the ACE RDC 2004 corpus.

5.1. Experimental setting

We mainly use the official ACE RDC 2003 corpus from LDC, which is divided into a training set and a testing set, for both detailed and final evaluation. The training set consists of 674 annotated text documents (~300*k* words) and 9683 instances of relations. During development, 519 documents in the training set are used for training while the remaining 155 (674 – 519) documents are set aside for fine-tuning the system. The testing set is held out only for final evaluation. It consists of 97 documents (~50*k* words) and 1386 instances of relations. The ACE RDC 2003 task defines 5 relation types and 24 subtypes between 5 entity types, i.e. person, organization, location, facility and GPE. Table 1 lists the types and subtypes of relations for the ACE RDC 2003 task, along with their frequencies of occurrence in the training set. It shows that this ACE RDC 2003 corpus suffers from a small amount of annotated data for a few subtypes such as the subtype "Founder" under the type "ROLE". It also shows that the ACE RDC 2003 task defines some difficult subtypes such as the subtypes "Based-In", "Located" and "Residence" under the type "AT", which are difficult even for human experts to differentiate.

Moreover, we also report our final performance on the ACE RDC 2004 corpus. Compared with the ACE RDC 2003 task, the ACE RDC 2004 task defines two more entity types, i.e. weapon and vehicle, much more entity subtypes, and different 7 relation types and 23 subtypes between 7 entity types. The ACE RDC 2004

[9] Those words that have the semantic classes "Parent", "GrandParent", "Spouse" and "Sibling" are automatically set with the same classes without change. However, The remaining words that do not have above four classes are manually classified.

Table 1
Relation types and subtypes in the ACE RDC 2003 training data

Type	Subtype	Occurrence frequency
AT(2781)	Based-In	347
	Located	2126
	Residence	308
NEAR(201)	Relative-Location*	201
PART(1298)	Part-Of	947
	Subsidiary	355
	Other	6
ROLE(4756)	Affiliate-Partner	204
	Citizen-Of	328
	Client	144
	Founder	26
	General-Staff	1331
	Management	1242
	Member	1091
	Owner	232
	Other	158
SOCIAL(827)	Associate*	91
	Grandparent	12
	Other-Personal	85
	Other-Professional*	339
	Other-Relative*	78
	Parent	127
	Sibling*	18
	Spouse*	77

Relations marked with an * are symmetric relations.

corpus from LDC contains 451 documents and 5702 relation instances. For comparison with Zhao and Grisman (2005), we only use the same set of 348 documents in the corpus, which contain 125*k* words and 4400 relation instances. Evaluation was done using 5-fold cross-validation.

In this paper, we iterate over all pairs of entity mentions occurring in the same sentence to generate potential relation instances. We also explicitly model the argument order of the two mentions involved. For example, when comparing mentions m1 and m2, we distinguish between m1-ROLE.Citizen-Of-m2 and m2-ROLE.Citizen-Of-m1. Note that, in the ACE RDC 2003 task, 6 of these 24 relation subtypes are symmetric: "NEAR.Relative-Location", "SOCIAL.Associate", "SOCIAL. Other-Relative", "SOCIAL.Other-Professional", "SOCIAL.Sibling", and "SOCIAL.Spouse". In this way, we model relation extraction as a multi-class classification task with 43 ($24 \times 2 - 6 + 1$) classes, two for each relation subtype (except the above 6 symmetric subtypes) and a "NONE" class for the case where the two mentions are not related. For the ACE RDC 2004 task, 6 of these 23 relation subtypes are symmetric: "PHYS.Near", "PER-SOC.Business", "PER-SOC.Family", "PER-SOC.Other", "EMP-ORG.Partner", and "EMP-ORG.Other". In this way, we model relation extraction as a multi-class classification task with 41 ($23 \times 2 - 6 + 1$) classes, two for each relation subtype (except the above 6 symmetric subtypes) and a "NONE" class for the case where the two mentions are not related.

5.2. Experimental results

In this paper, we only measure the performance of relation extraction on "true" mentions with "true" chaining of coreference (i.e. as annotated by the corpus annotators) in the ACE corpora. Table 2 measures the performance of our relation extraction system over the 24 relation subtypes on the testing set of the ACE RDC 2003 corpus. It shows that our system achieves best performance of 63.1%/49.5%/55.5 in precision/recall/*F*-measure when combining all the diverse lexical, syntactic and semantic features. Therefore, all

Table 2
Contribution of different features over 43 relation subtypes in the ACE RDC 2003 test data

Features	Precision	Recall	*F*-measure
Words	69.2	23.7	35.3
+Entity type	67.1	32.1	43.4
+Mention level	67.1	33.0	44.2
+Overlap	57.4	40.9	47.8
+Chunking	61.5	46.5	53.0
+Dependency tree	62.1	47.2	53.6
+Parse tree	62.3	47.6	54.0
+Semantic resources	63.1	49.5	55.5

the features described in Table 2 are employed thereafter in our system since such combination shows best performance. Table 2 also measures the contributions of different features by gradually increasing the feature set in the increasing order of feature complexity. It shows that:

- Using word features only achieves the performance of 69.2%/23.7%/ 35.3 in precision/recall/*F*-measure.
- Entity type features are very useful and improve the *F*-measure by 8.1 units largely due to the recall increase.
- The usefulness of mention level features is quite limited. It only improves the *F*-measure by 0.8 units due to the recall increase.
- Incorporating the overlap features gives some balance between precision and recall. It increases the *F*-measure by 3.6 units with a big precision decrease and a big recall increase.
- Chunking features are very useful. It increases the precision/recall/*F*-measure by 4.1%/5.6%/5.2 units, respectively.
- To our surprise, incorporating the dependency tree and parse tree features only improve the *F*-measure by 0.6 and 0.4 units, respectively. This may be due to the fact that most of relations in the ACE RDC 2003 corpus are quite local. Table 3 shows that about 70% of relations exist where two mentions are embedded in each other or separated by at most one word, although the context information in the left and right of the two mentions is also useful in relation extraction. While short-distance relations dominate and can be resolved by above simple features, the dependency tree and parse tree features can only take effect in the remaining much less long-distance relations. However, full parsing is always prone to long-distance errors although the Collins' parser used in our system represents the state-of-the-art in full parsing. Another reason may be that current dependency tree and parse tree features extracted in this paper is not effective to reflect the syntactic structure in relation extraction.
- Incorporating semantic resources such as the country name list and the personal relative trigger word list further increases the *F*-measure by 1.5 units largely due to the differentiation of the relation subtype

Table 3
Distribution of relations over # words and # other mentions in between in the ACE RDC 2003 training data

# of relations		# of other mentions in between					
		0	1	2	3	⩾4	Overall
# of the words in between	0	3991	161	11	0	0	4163
	1	2350	315	26	2	0	2693
	2	465	95	7	2	0	569
	3	311	234	14	0	0	559
	4	204	225	29	2	3	463
	5	111	113	38	2	1	265
	⩾6	262	297	277	148	134	1118
	Overall	7694	1440	402	156	138	**9830**

"ROLE.Citizen-Of" from "ROLE. Residence" by distinguishing country GPEs from other GPEs. The effect of personal relative trigger words is very limited due to the limited number of testing instances over personal social relation subtypes.

Table 4 separately measures the performance of different relation types and major subtypes in the ACE RDC 2003 corpus. It also indicates the number of testing instances, the number of correctly classified instances and the number of wrongly classified instances for each type or subtype. It is not surprising that the performance on the relation type "NEAR" is low because it occurs rarely in both the training and testing data. Others like "PART.Subsidary" and "SOCIAL.Other-Professional" also suffer from their low frequencies of occurrence. It also shows that our system performs best on the subtype "SOCIAL.Parent" and "ROLE.Citizen-Of". This is largely due to incorporation of two semantic resources, i.e. the country name list and the personal relative trigger word list. Table 4 also indicates the low performance on the relation type "AT" although it frequently occurs in both the training and testing data. This suggests the difficulty of detecting and classifying the relation type "AT" and its subtypes.

Table 5 separates the performance of relation detection from overall performance on the testing set of the ACE RDC 2003 corpus. It shows that our system achieves the performance of 84.8%/66.7%/74.7 in precision/recall/*F*-measure on relation detection. It also shows that our system achieves overall performance of 77.2%/60.7%/68.0 and 63.1%/49.5%/55.5 in precision/recall/*F*-measure on the 5 relation types and 24 relation subtypes, respectively. Compared with the best-reported system in Kambhatla (2004), our system achieves better

Table 4
Performance of different relation types and major subtypes in the ACE RDC 2003 test data

Type	Subtype	# Testing instances	# Correct	# Error	Precision	Recall	*F*-measure
AT		**392**	**224**	**105**	**68.1**	**57.1**	**62.1**
	Based-In	85	39	10	79.6	45.9	58.2
	Located	241	132	120	52.4	54.8	53.5
	Residence	66	19	9	67.9	28.8	40.4
NEAR		**35**	**8**	**1**	**88.9**	**22.9**	**36.4**
	Relative- Location	35	8	1	88.9	22.9	36.4
PART		**164**	**106**	**39**	**73.1**	**64.6**	**68.6**
	Part-Of	136	76	32	70.4	55.9	62.3
	Subsidiary	27	14	23	37.8	51.9	43.8
ROLE		**699**	**443**	**82**	**84.4**	**63.4**	**72.4**
	Citizen-Of	36	25	8	75.8	69.4	72.6
	General-Staff	201	108	46	71.1	53.7	62.3
	Management	165	106	72	59.6	64.2	61.8
	Member	224	104	36	74.3	46.4	57.1
SOCIAL		**95**	**60**	**21**	**74.1**	**63.2**	**68.5**
	Other-Professional	29	16	32	33.3	55.2	41.6
	Parent	25	17	0	100	68.0	81.0

Table 5
Comparison of our system with other best-reported systems in the ACE RDC 2003 test corpus

System	Relation detection			RDC on types			RDC on subtypes		
	P	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
Ours: SVM (linear kernel)[a]	84.8	66.7	74.7	77.2	60.7	68.0	63.1	49.5	55.5
Ours: SVM (polynomial[a] kernel)	89.3	63.9	74.5	81.9	58.3	68.1	69.8	49.7	58.1
Kambhatla (2004): MaxEnt[a]	–	–	–	–	–	–	63.5	45.2	52.8
Culotta and Sorensen (2004): tree kernel	81.2	51.8	63.2	67.1	35.0	45.8	–	–	–
Bunescu and Mooney (2005): tree kernel	–	–	–	65.5	43.8	52.5	–	–	–

[a] For feature-based.

performance by ~3 units in *F*-measure largely due to its gain in recall. Moreover, we also evaluate our system using the polynomial kernel with degree $d = 2$. It shows that this can further improve the *F*-measure by 2.6 units in the extraction of 24 relation subtypes, largely due to its gain in precision, although it makes little difference in the extraction of 5 relation types. Finally, it also shows that feature-based methods dramatically outperform kernel methods. This suggests that feature-based methods can effectively combine different features from a variety of sources (e.g. WordNet) that can be brought to bear on relation extraction while current tree kernels developed in Culotta and Sorensen (2004) and Bunescu and Mooney (2005) mainly capture the structured syntactic information and are yet to be effective on the ACE RDC 2003 task.

We also evaluate our final performance in the ACE RDC 2004 corpus using 5-fold cross-validation. Table 6 shows that our system achieves the performance of 87.6%/64.0%/74.0 in precision/recall/*F*-measure on relation detection. It also shows that our system achieves overall performance of 81.4%/59.5%/68.7 and 73.8%/54.1%/62.4 in precision/recall/*F*-measure on the 7 relation types and 23 relation subtypes, respectively. This indicates that the ACE RDC 2004 task is much easier than the ACE RDC 2003 task, especially in the extraction of relation subtypes, largely due to the fine definition of entity subtypes in the ACE RDC 2004 task. Moreover, we also evaluate our system using the polynomial kernel with degree $d = 2$. It shows that this can further improve the *F*-measure by 1.3 units in the extraction of 23 relation subtypes. Compared with the best-reported system in Zhao and Grisman (2005) which applies a complicated composite polynomial kernel, our system achieves better performance by 0.7 units in *F*-measure in the extraction of 7 relation types. It is interesting to note that our system and Zhao and Grisman (2005) are quite complementary in precision and recall: our system achieves much higher precision and lower recall while Zhao and Grisman (2005) has lower precision and higher precision. This indicates possible performance improvement by integrating them.

Due to the difficulty in building a large annotated corpus, another interesting question is about the adaptability of our system. Fig. 2 shows the effect of different training data sizes for some major relation subtypes in the ACE RDC 2003 corpus while keeping all the training examples of remaining relation subtypes. It shows that the three major relation subtypes can achieve a bit stable performance on different training data sizes ranging from 700 to 1100 training examples. It also shows that further steady improvement can be achieved for the three major relation subtypes after the turning points. This indicates that, as the current largest anno-

Table 6
Comparison of our system with other best-reported systems in the ACE RDC 2004 corpus using 5-fold cross-validation

System	Relation detection			RDC on types			RDC on subtypes		
	P	*R*	*F*	*P*	*R*	*F*	*P*	*R*	*F*
Ours: SVM (linear kernel)[a]	87.6	64.0	74.0	81.4	59.5	68.7	73.8	54.1	62.4
Ours: SVM (polynomial kernel)[a]	89.0	66.6	76.2	82.8	62.1	71.0	74.3	55.9	63.7
Zhao and Grisman (2005): SVM (composite polynomial)[a]	–	–	–	69.2	70.5	70.3	–	–	–

[a] For feature-based.

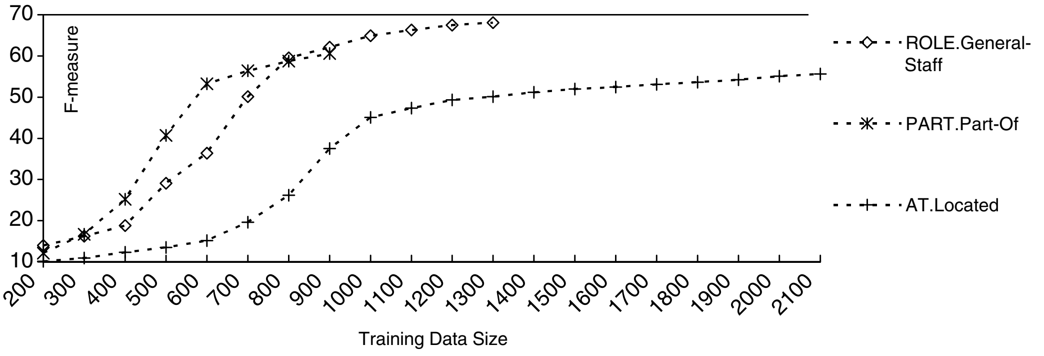


Fig. 2. The impact of different training data sizes on the three major relation subtypes in the ACE RDC 2003 corpus.

Table 7
Distribution of errors in the ACE RDC 2003 corpus

Error type	#Errors
Detection error	
False negative	462
False Positive	165
Characterization error	
Cross-type error	154
Inside type error	83

tated corpus in relation extraction, the ACE RDC 2003 corpus still suffers from the lack of training data, even for major relation subtypes.

Finally, Table 7 shows the distributions of errors in the ACE RDC 2003 corpus. It shows that 73% (627/864) of errors results from relation detection and 27% (237/864) of errors results from relation characterization, among which 17.8% (154/864) of errors are from misclassification across relation types and 9.6% (83/364) of errors are from misclassification of relation subtypes inside the same relation types. This suggests that performance improvement on relation detection is critical for the success of relation extraction.

6. Discussion and conclusion

In this paper, we have presented a feature-based approach for relation extraction where diverse lexical, syntactic and semantic knowledge are employed. Instead of exploring the full parse tree information directly as previous related work, we incorporate the base phrase chunking information first Evaluation on the ACE RDC corpora shows that base phrase chunking contributes to most of the performance improvement from syntactic aspect while further incorporation of the parse tree and dependence tree information only slightly improves the performance. This may be due to three reasons: First, most of relations defined in the ACE RDC task have two mentions being close to each other While short-distance relations dominate and can be resolved by simple features such as word and chunking features, the further dependency tree and parse tree features can only take effect in the remaining much less and more difficult long-distance relations. Second, full parsing is always prone to long-distance parsing errors although the Collins' parser used in our system achieves the state-of-the-art performance. Therefore, the state-of-art full parsing still needs to be further enhanced to provide accurate enough information, especially PP (Preposition Phrase) attachment.[10] Last, effective way need to be explored to incorporate information embedded in the full parse trees. This means that cup rent fixed dependency tree and parse tree features used in this paper are too simple to effectively reflect the syntactic structure in relation extraction. However, this also suggests that a cheap and robust solution can be achieved in relation extraction with the near state-of-the-art performance. Besides, we also demonstrate how semantic information such as WordNet, can be used in feature-based relation extraction to further improve the performance.

The effective incorporation of diverse features enables our system outperform previously best-reported systems on the ACE RDC corpora. Although tree kernel-based approaches facilitate the exploration of the implicit feature space with the parse tree structure, yet the current technologies are expected to be further advanced to effectively explore the structured syntactic information. Evaluation on the ACE RDC corpora shows that our approach of combining various kinds of evidence can scale better to problems, where we have a lot of relations with a relatively small amount of annotated data. The experiment result also shows that our feature-based approach significantly outperforms the tree kernel-based approaches. However, we find it difficult to further improve the performance in feature-based relation extraction by incorporating more features. Moreover, our error analysis shows that relation detection is critical in relation extraction. We think that the structured syntactic information may play a critical role in detecting a relation. This indicates that future

[10] NP/PP/VP all play an important role in semantic relation extraction. However, compared to NP/VP recognition, PP attachment is a much harder problem in full parsing, which is yet to be resolved successfully.

success in relation extraction largely depends on effectively exploring structured syntactic information. This suggests the urgency in exploring more effective tree kernels in relation extraction in the future.

In the future work, we will focus on research and development in tree kernels. A straightforward way is to explore existing kernels, e.g. the convolution kernel (Haussler, 1999) as used in full parsing (Collins & Duffy, 2001) and semantic role labeling (Moschitti, 2004), in relation extraction. In the meanwhile, we will explore more semantic knowledge in relation extraction, which has not been covered deeply by current research. Finally, our current work is done when the entity detection and tracking (EDT) task has been perfectly done. Therefore, it would be interesting to see how imperfect EDT affects the performance in relation extraction.

References

ACE, (2000–2005). Automatic content extraction. Available from http://www.ldc.enn.edu/Projects/ACE.

Bunescu, R. & Mooney, R. J. (2005). A shortest path dependency kernel for relation extraction. *HLT/EMNL.P'2005* (pp. 724–731), 6–8 October 2005, Vancover, BC.

Collins, M. (1999). Head-driven statistical models for natural language parsing. Ph.D. Dissertation, University of Pennsylvania.

Collins, M., & Duffy, N. (2001). Covolution kernels for natural language. *NIPS-14* (pp. 625–632) Cambridge, MA.

Culotta, A., Sorensen, J. (2004). Dependency tree kernels for relation extraction. *ACL'2004* (pp. 423–429) 21–26 July 2004, Barcelona, Spain.

Haussler, D. (1999). Covention kernels on discrete structures. Technical Report UCS-CRL-99-10, University of California, Santa Cruz.

Joachimi, T. (1998). Text categorization with support vector machines: Learning with many relevant features. *ECML'1998*:1370142, 21–23 April 1998, Chemnitz, Germany.

Kambhatla, N. (2004). Combining lexical, syntactic and semantic features with maximum entropy models for extracting relations. *ACL'2004(poster)* (pp. 178–181), 21–26 July 2004, Barcelona, Spain.

Miller, G. A. (1990). WordNet: An online lexical database. *International Journal of Lexicography, 3*(4), 235–312.

Miller, S., Fox, H., Ramshaw, L., & Weischedel, R. (2000). A novel use of statistical parsing to extract information from text. *ANLP'2000* (pp. 226–233) 29 April–4 May 2000, Seattle, USA.

Moschitti, A. (2004). A study on Convolution kernels for semantic role labeling. *ACL'2004* (pp. 335–342), 21–26 July 2004, Barcelona, Spain.

MUC-7 (1998). *Proceedings of the 7th Message Understanding Conference (MUC-7)*. San Mateo, CA: Morgan Kaufmann.

Roth, D., & Yih, W. T. (2002). Probabilistic reasoning for entities and relation recognition. *CoL-ING'2'2002* (pp. 835–841), 24 August–1 September 2002, Taiwan.

Vapnik, V. (1998). *Statistical learning theory*. Chichester, GB: Whiley.

Zelenko, D., Aone, C., & Rochardella, A. (2003). Kernel methods for relation extraction. *Journal of Machine Learning Research, 3*, 1083–1106.

Zhang, M., Su, J., Wang, D. M., Zhou, G. D., & Tan, C. L. (2005). Discovering relations from a large raw corpus using tree similarity-based clustering. *IJCNLP'2005, Lecture Notes in Computer Science (LNCS 3651)* (pp. 378–389), 11–16 October 2005, Jeju Island, South Korea.

Zhao, S. B., & Grisman, R. (2005). Extracting relations with integrated information using kernel methods. *ACL'2005* (pp. 419–426), 25–30 June 2005, Ann Arbor, USA.

Zhou, G. D., Su, J., Zhang, J., & Zhang, M. (2005). Exploring various knowledge in relation extraction. *ACL'2005* (pp. 427–434), 25–30 June 2005, Ann Arbor, USA.

Dr. Zhou GuoDong received his B.Sc, M.Sc. and Ph.D. from XI'AN Jiaotong Univ. in 1989, Shanghai Jiaotong Univ. in 1992 and National Univ. of Singapore in 1999, respectively. He joined the Institute for Infocomm Research, Singapore in 1999 and had been an associate lead scientist at the institute until August 2006. Currently, he is a professor at the School of Computer Science and Technology, Suzhou Univ., China.

Dr. Zhang Min received his B.Sc, M.Sc. and Ph.D. from Harbin Institute of Technology in 1991, 1994 and 1997, respectively. He joined the Institute for Infocomm Research in 2003 and currently is a scientist at the institute.