Entity Relation Detection with Factorial Hidden Markov Models and Maximum Entropy Discriminant Latent Dirichlet Allocations

A DISSERTATION

SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL

OF THE UNIVERSITY OF MINNESOTA

BY

Dingcheng Li

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

Doctor of Philosophy

Jeanette Gundel, William Schuler

January, 2012

Acknowledgements

There are many people that have earned my gratitude for their contribution to my time in graduate school. I would like to express my sincere thanks and gratitude to my thesis advisor Professor Jeanette Gundel and Professor William Schuler for their constant encouragement, infinite patience and valuable guidance through these years of my graduate study. I would like to thank my NLP labmates Tim Miller, Stephen Wu and Luan Nguyen for the discussion on the development of FHMM based coreference resolution models. I would like to thank my Siemens intern adviser Swapna Sundaran for her guidance on the part about relation detection. I also like to thank Hooi Ling Soh for her guidance on my first two years' linguistics study. Thanks also go to Dr. Guergana Savova who was my adviser in my internships at Mayo Clinic and one of my advisers during my 2-year BICB traineeship and Professor Arindam Banerjee who has given much help in improving my machine learning during my thesis development and during the weekly machine learning seminar in his group.

I would like to thank all other labmates from NLP lab of the computer science department and my colleagues and professors from linguistics department. I also would like to thank many friends from related tracks and departments, like Jiaping Zheng and Bridget Thomson McInnes who are from NLP as well, Qiang Fu, Hanhuai Shan and Huahua Wang who are from machine learning, Yu Jin and Yu Gu, from network, Guquan Huang from robotics, Likun Zheng, from mathematics and Ming Huang from scientific computation and so on for their valuable help on problems I met during my research.

I would like to thank my grandparents and my father for their unconditional trust and love. I want to thank my younger sister Dingxue Li and her husband Andy Wu for their long-term support and understanding during the years of my zig-zag path to my PhD Study.

Finally, my heartfelt gratitude goes to my wife Qiongying Xiu for her faithful love and encouragement. Without her endless support in the past few years, it is impossible for me to

finish all the models designs and coding and painful debugging day and night.

Dedication

Dedicated to my grandparents, my wife and my sister. And also to NLP, the field I really enjoy. My study experience confirm that God help those who help themselves.

Abstract

Coreference resolution (CR) and entity relation detection (ERD) aim at finding predefined relations between pairs of entities in text. CR focuses on resolving identity relations while ERD focuses on detecting non-identity relations. Both CR and ERD are important as they can potentially improve other natural language processing (NLP) related tasks such information retrieval and extraction, web-searching, and question answering and also enhance non-NLP tasks such as computer vision, database constructions or ontologies.

In this thesis, I propose models to handle both coreference resolution (CR) and entity relation detection (ERD). Both systems are built on machine learning models. The CR system is based on Factorial Hidden Markov Models (FHMMs). The ERD is based on Maximum Entropy Discriminant Latent Dirichlet Allocation (MEDLDA). The work on CR only resolves pronouns. It is a supervised system trained on annotated corpus. The basic idea is that the hidden states of FHMMs are an explicit short-term memory with an antecedent buffer containing recently described referents. Thus an observed pronoun can find its antecedent from the hidden buffer, or in terms of a generative model, the entries in the hidden buffer generate the corresponding pronouns. In the hidden buffer, all references are expressed as diverse features. In this work, besides the common gender, number, person and animacy, I converted Givenness Hierarchy and Centering Theories to probabilistic features, thus greatly improving the accuracy. A system implementing this model is evaluated on the ACE corpus and I2B2 medical corpus with promising performance.

For ERD, a novel application of topic models is proposed to do this task. In order to make use of the latent semantics of text, the task of relation detection is reformulated as a topic modeling problem. The motivation is to find underlying topics which are indicative of relations between named entities. The approach considers pairs of named entities and features associated with them as mini documents. The system, called ERD-MEDLDA, adapts Maximum Entropy Discriminant Latent Dirichlet Allocation (MedLDA) with mixed membership for relation detection. By using supervision, ERD-MedLDA is able to learn topic distributions indicative of relation types. Further, ERD- MEDLDA is a topic model that combines the benefits of both Maximum Likelihood Estimation (MLE) and Maximum Margin

Estimation (MME), and the mixed membership formulation enables the system to incorporate heterogeneous features. We incorporate diverse features into the system and perform experiments on the ACE 2005 corpus. Our approach achieves better overall performance for precision, recall and Fmeasure metrics as compared to SVM-based and LDA-based models. ERD-MedLDA also shows better overall performance than state-of-the-art kernels used previously for relation detection.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In recent years, with the explosion of the electronically available information, it is hard for human beings to handle such a large amount of information without the help of external tools. In natural language processing (NLP), information retrieval (IR), database management, computer vision and other fields, the same entities in the real world may involve different names, descriptions and perspectives. It is necessary to correlate these names, descriptions and perspectives to the real entities for a deeper data understanding. In computer vision, we must be able to make consistent entity judgment and linkage before we begin to manipulate them. In database management, creating a data set with clear reference via record linkage and reduplication may make data mining more accurate. Likewise, in NLP, if we can resolve expressions that point to the same entities or if we can construct relations between entities, we can perform information extraction (IE), named entity recognition (NER), parsing or other NLP tasks more accurately. The process of identifying the predefined relations between pairs of entities in text is called entity relation detection (ERD) and the process of identifying linguistic expressions or other data points, e.g. images, that refer to the same entity is called coreference resolution (CR).

ERD and CR are closely related. Both of them study the relations between named entities. In fact, CR can be regarded to one of the subsets of ERD or a special ERD since CR cares about identity relationship between entities while ERD is a broader task. Nonetheless, in convention, ERD, the term is usually used to detect relations other than identity ones. In addition, systems for ERD and those for CR are usually constructed with different models

because both tasks differ much in specific methodologies. Therefore, in this thesis, following conventions, we use ERD refer to non-identity relation detection and use CR to refer to identity relation detection.

In this thesis, two systems are proposed with one of them handling coreference resolution and the other handling relation detection. Both models are statistical models. The coreference resolution system is based on Factorial Hidden Markov models (FHMM) and the relation detection system is based on Maximum Entropy Discriminant Latent Dirichlet Allocation.

Though the two systems aim at solving all kind of relations between entities, we know that this is a too ambitious goal and thus have to narrow our scope in the research process. The coreference resolution system only focuses on coreference between pronouns and their antecedents. The coreference relations include both intra-sentential and inter-sentential. Obviously, pronoun coreference discussed in this thesis excludes cross-documents. It is meaningless to talk about antecedents across documents

though the literature in natural language processing often talks about cross-docoment coreference. The reason that we only focus on pronouns is that pronoun coreference resolution has quite different features from resolution of nominal coreference. Pronoun coreference resolution depends more on gender, number, animacy and person features, while nominal coreference resolution requires more consideration of semantic features relating to denotations of words.For example, in medical fields, each medical term has a few semantic categories such as Concept Unique Identifier (CUI) defined in the Unified Medical Language System (UMLS). Detection of coreference between medical terms can use CUI or other semantic categories as features. With such observations, we decide to divide and conquer. In this thesis, we handle pronoun coreference resolution though it should not be a difficult mission to expand the present system to include nominal coreference resolution.

For ERD, things are more complex since besides identity relations, there are numerous other relations between entities, which can be intra-sentential and inter-sentential, as well as across documents. Therefore, we have to deÞne a range for an ERD system.

With this in mind and also with the available ACE corpus version 2005, the ERD system is defined as an intra-sentential relation detection system and further, the relations to detect are those defined in ACE 2005.

Intra-sentential relation detection here involves two meanings. Firstly, such a system

aims at detecting relations within one sentence. Secondly, the relation detected excludes coreference as the latter is covered under the CR system. The intra-sentential relations refer to relations such as part-whole, person-social, physical or organization-affiliate and so on. I restrict the broader relation detection research to intra-sentential due to two reasons. Firstly, most of the existing research focuses on intra-sentential relation detection at present. I plan to develop a supervised system based on ACE corpus. ACE corpus in fact only has intra-sentential relation annotations. Hence, this restriction is a better strategy for fair evaluations and comparisons with previous studies. Secondly, the coreference resolution system resolves both inter-sentential coreference and intra-sentential coreference.

The resolution of coreference can complement the broader relation detection system. Namely, once a coreference chain is set up and intra-sentential relations are detectedl, most non-coreferring inter-sentential relations can be resolved as well. This is also one of the reasons why this thesis sometimes uses ERD to refers to both tasks. Without doubt, the remaining unresolved inter-sentential relations need more exploration. While this is beyond the range of this thesis. It will be addressed in future work.

- Chapter 2 briefly presents related work for coreference resolution, including different approaches to coreference resolution.
- Chapter 3 describe in detail about how to construct Factorial Hidden Markov Models (FHMM) to do coreference resolution. This chapter includes results of research conducted when I worked at NLP lab at UMN and the related paper was published in ACL 2011.
- Chapter 4 illustrate what extensions have been made to improve the FHMM-based coreference resolution models.
- Chapter 5 a brief literature review on relation detection.
- Chapter 6 introduce initial experiments on relation detection with LDA models.
- Chapter 7 describe in detail about how to construct Maximum Entropy Discriminative Latent Dirichlet Allocation to do relation detection. This chapter includes results of research conducted while I was working as a summer intern at Siemens supervised by

Dr. Swapna Somasundaran. The results have been published in the workshop Graph-based Methods for Natural Language Processing under ACL. The longer version of that paper has been accepted for publication in Journal of Natural Language Engineering.

- Chapter 8 summarizes what I have done for conference resolution and relation detection, their interactions between each other and future work.

Chapter 2

Literature Review on Coreference Resolution

Many approaches have been adopted to tackle the problem of coreference resolution. In essence, we can classify these approaches into two main categories: deterministic (rule-based methods which rely heavily on linguistic and domain knowledge), and non-deterministic (statistical/probabilistic methods which treat the corpus as data composed of random variables). Deterministic methods try to find rules and apply them to unseen data. The rules are often based on linguistic theories. Non-deterministic methods assume that the corpora display certain statistical distributions which can be modeled numerically to create predictors. Both approaches have their strengths and weaknesses. In this section, I will give a general overview of the major efforts for coreference resolution.

2.1 Linguistic Approaches

2.1.1 Hobbs' Algorithm

Hobbs' algorithm (Hobbs 1986) is based on searching or traversal of the syntactic parse tree of the sentences. It makes use of syntactic and semantic constraints when resolving pronouns.

The searching algorithm goes as follows (Hobbs 1986):

1. Hobbs Searching Algorithm

step 1: begin at NP node immediately dominating the pronoun

step 2: go up the tree to the first NP or S node encountered call this node X and call the path to reach X "p"

step 3: traverse all branches below node X to the left of path p, in left-to-right, breadth-first manner propose as the antecedent any NP node that is encountered that has an NP or an S node between it and X

step 4: if node X is the highest S node in the sentence traverse the parse trees of previous sentences in order of recency (the most recent first), from left-to-right, breadth-first and propose as antecedent the first NP encountered else go to step (5)

step 5: from node X go up the tree to the first NP or S node encountered call this new node 'X', and call the path traversed to reach it from the original X 'p'

step 6: if X is an NP node AND if the path p to X did not pass through the N-bar node that X immediately dominates, propose X as the antecedent

step 7: go to step 4

This algorithm traverses the surface parse tree, searching for a noun phrase without violating selection constraints.

Syntactic constraints are one of the main tenets. They mainly involve binding principles proposed by Chomsky (Chomsky 1981). Binding principles are composed of three main parts:

2. Binding principle

   (a) Principle A states that anaphors (reflexives and reciprocals, such as 'each other') must always be bound in their domains;

   (b) Principle B states that a pronoun must never be bound within its domain;

   (c) Principle C states that R-expressions must never be bound;

The importance of binding is shown by the following facts:

3. Sample One

   (a) $John_i$ saw him_i (ungrammatical with co-reference)

(b) John saw himself. (unambiguously co-referent)

(c) *Himself saw John. (ungrammatical)

(d) $John_i$ saw $John_i$. (ungrammatical, unless it refers to two distinct Johns)

Since there is nothing to bind the anaphor $himself$ in Sample One (c), principle A is violated, and the sentence is ungrammatical. If, in (a), $John$ and him are co-referring, then this coreference violates Principle B, resulting in ungrammaticality since the pronoun him would be bound by $John$.

R-expressions are referential expressions: non-pronoun, uniquely identifiable entities, such as $thedog$, or proper names such as $John$. In (d), the first instance of $John$ binds the second, resulting in the ungrammaticality due to Principle C, which states that R-expressions must never be bound.

Besides binding constraints, other constraints such as gender, number or person features of words are employed. The following figure displays how Hobbs algorithm works.

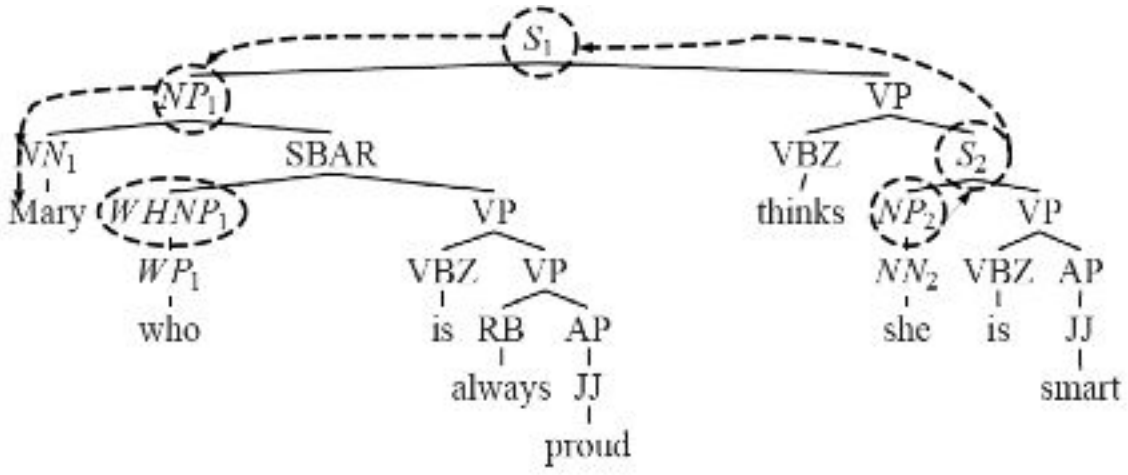

Figure 2.1: Hobbs Algorithm on a CNF Sample Tree

Let us suppose that $Mary$ and who have been found coreferring. The traversal starts from node NP_2 and then NP_2 rises to node S_2 with Step 2. Step 3 will be skipped since there is no NP or S node between S_2 and NP_2. Step 4 doesn't apply either since S_2 is not the highest node. Step 5 would work and thus go up to S_1. Step 6 and step 7 don't

apply since S_1 is not an NP node. Then step 8 pushes the search down to NP_1 and even down to NN_1 and find possible antecedent $Mary$.

The position of a pronoun in the sentence restricts the possible antecedents within that sentence. When we look for antecedents in previous sentences, the referents of antecedents that occur in the subject position are more salient since a breadth-first left-to-right search is performed starting at the root S node of the sentence. Depth of a node in the syntactic tree is thus a very important factor to determine discourse prominence.

Hobbs' algorithm mainly is used for intra-sentential coreference resolution. It is hard for it to expand to inter-sentential coreference resolution (though possible). The reason is that it will be hard to search cross-trees and the search space will be huge.

2.2 Centering Theory

Centering theory (B.J.Grosz et al., 1995) was proposed in order to model the relationships among focus of attention, choice of referring expression and perceived coherence of utterances within a discourse segment". The model draws inspiration from previous discourse processing papers, and its principles have been widely used directly or indirectly in much later work.

One of the main goals of Centering is to track the entities in focus in a given sentence. It has a few assumptions: a discourse segment consists of a sequence of utterances U_1 , ... , U_i, ... ,U_m. Each utterance is a sentence, which is associated with a list of $forward-lookingcenters$, $C_f(U_i)$. Ranking of an entity on this list corresponds roughly to the likelihood that it will be the primary focus of subsequent discourse; the first entity on this list is the preferred center, $C_p(U_i)$. Each utterance, U_i, is assigned only one real center to one entity at a time, which is called $backward-lookingcenter$, $C_b(U_i)$. It is a confirmation of an entity that has already been introduced into the discourse. That is, it must be realized in the immediately preceding utterance, U_{i-1}. Related to these concepts are transitions types which are based on two factors: whether or not the center of attention, C_b, is the same from U_{i-1} to U_i , whether or not this entity coincides with the preferred center of U_i. Definitions of these transitions types are in

$Table 2-1$.

Table 2.1: Transition Types of Centering

	$C_b(U_{i+1})$=$C_b(U_i)$	$C_b(U_{i+1})$!= $C_b(U_i)$
$C_b(U_{i+1})$=$C_p(U_i)$	(1) Continue	(3) Smooth-shift
$C_b(U_{i+1})$!= $C_p(U_i)$	(2)Retain	(4) Rough-shift

Based on the above definition of centers and transition types, we can predict the antecedent for an anaphor with proposed constraints and rules (Brennan, Friedman et al. 1987).

4. Constraints

   (a) there is precisely one C_b

   (b) Every element of $C_f(U_i)$ must be realized in U_i

   (c) $C_b(U_i)$ is the highest-ranked element of $C_f(U_{i-1})$ that is realized in U_i

5. Rules

   (a) If some element of $C_f(U_{i-1})$ is realized as a pronoun in U_i, then so is $C_b(U_i)$.

   (b) Continuing $\succ$ retaining $\succ$ smooth-shifting $\succ$ rough-shifting

Ranking of the items on the forward center list, C_f is crucial. BFP rank them by obliqueness of grammatical relation of the subcategorized functions of the main verb: subject $\succ$ indirect object $\succ$ other $\succ$ adjunct.

In practice, the algorithm using centering theory can be specified as four steps: $Construct$, $Filter$, $Classify$ and $Select$. $Construct$ proposes the possible C_f, C_p and C_b; $Filter$ uses rules and constraints to rule out wrong assumptions based on rules and constraints which include syntactic constraints; $Classify$ lists possible transition types of U_i; finally $Select$ determines what transition type is more likely and then resolves coreference.

Now, let us use the following example (Elango 2006) to illustrate how centering theory is used in coreference resolution.

6. Sample Two

   (a) Terry really goofs sometimes.

   (b) Yesterday was a beautiful day and he was excited about trying out his new sailboat.

   (c) He wanted Tony to join him on a sailing expedition.

   (d) He called him at 6 AM.

   (e) He was sick and furious at being woken up so early.

   In the above example, $Terry$ is the subject from (a) to (d) and $Tony$ is the object in (d) and and the subject in (e). The subject changes from $Terry$ to $Tony$ though the pronoun used is the same in (e). We can say the focus is continued in the first 4 utterances and the focus shifts in (e). Formerly, it goes like the following:

7. The reasoning of U_1

$$\begin{aligned} U_1: &\quad Terry\ really\ goofs\ sometimes. \\ Construct: &\quad C_f :< Terry = TERRY > \\ &\quad C_b :< NIL > \\ &\quad C_p :< Terry = TERRY > \\ Filter: &\quad X \\ Classify &\quad - \\ Select: &\quad - \end{aligned}$$

   U_1 is the first sentence, so C_b is null. There is only one entity (Terry) in U_i, so Terry is the only element in C_f and it is C_p as well. Then, filter, classify and select do not do anything.

8. The reasoning of U_2

U_2: $Yesterday\ was\ a\ beautiful\ day\ and\ he\ was\ excited\ about\ trying\ out\ his\ new\ sailboat.$

$Construct$: $C_f :< he = TERRY, his = TERRY >$

$C_b :< he = TERRY > or < his = TERRY >$

$C_p :< he = TERRY >$

$Filter$: $by,$**he or his refers to C_b**

$Classify$

$1) C_b :< he = TERRY >$

$so, C_b(U_2) = C_b(U_1), C_b(U_2) = C_p(U_2)$

$i.e.$**Continuing**

$2) C_b :< his = TERRY >$

$so, C_b(U_2) = C_b(U_1), C_b(U_2) \neq C_p(U_2)$

$i.e.$**Retaining**

$Select$: **Continuing by 5, which says that Continuing > Retaining**

9. The reasoning of U_3

U_3: $He\ wanted\ Tony\ to\ join\ him\ on\ a\ sailing\ expedition$

$Construct$: $a.C_f :< he = TERRY, Tony = TONY, him = TERRY >$

$or b. < he = TONY, Tony = TONY, him = TERRY >$

$or. < he = TERRY, Tony = TONY, him = TONY >$

$ord. < he = TONY, Tony = TONY, him = TONY >$

$C_b :< he = TERRY > or < him = TERRY >$

$C_p :< he = TERRY > or < he = TONY >$

$Filter$: $by,$ **he or him refers to C_b**

$by\ syntactic\ constraints,\ b,\ c\ and\ d\ in\ Cf\ are\ ruled\ out$
$since\ all\ of\ them\ violate\ the\ principle\ that\ pronouns\ can$
$never\ be\ bound\ within\ their\ domains.$
$Thus,\ Cp\ cannot\ be < he = TONY >\ then.$

$Classify$

1) $C_b : < he = TERRY >$

$so,\ C_b(U_3) = C_b(U_2), C_b(U_3) = C_p(U_3)$

$i.e.$ **Continuing**

2) $C_b :< his = TERRY >$

$so,\ C_b(U_3) = C_b(U_2), C_b(U_2) \neq C_p(U_2)$

$i.e.$ **Retaining**

$Select$: **Continuing by 5, which says that Continuing > Retaining**

10. The reasoning of U_4

U_4: $He called him at 6 AM.$

$Construct: Cf$: $a. < he = TERRY, him = TERRY >$
$or\ b. < he = TONY, him = TONY >$
$or\ c. < he = TERRY, him = TONY >$
$or\ d. < he = TONY, him = TERRY >$

Cb: $< he = TERRY > or < him = TERRY >$
$or < he = TONY > or < him = TONY >$

Cp: $< he = TERRY > or < he = TONY >$

$Filter$: $by, he or him refers to Cb;$
$by\ syntactic\ constraints, a\ and\ b\ are\ ruled\ out$
$since\ all\ of\ them\ violate\ the\ principle\ that\ pronouns$
$can\ never\ be\ bound\ within\ their\ domains.$

$Classify$:

1) $Cb: he = TERRY$
$so,\ Cb(U4) = Cb(U3), Cb(U4) = Cp(U4)$
$i.e. Continuing$

2) $Cb: him = TERRY$
$so,\ Cb(U4) = Cb(U3), Cb(U4) \eta Cp(U4)$
$i.e. Retaining$

3) $Cb: he = TONY$
$so,\ Cb(U4) \eta Cb(U3), Cb(U4) = Cp(U4)$
$i.e. Smooth Shifting$

4) $Cb: him = TONY$
$so,\ Cb(U4) \eta Cb(U3), Cb(U4) \eta Cp(U4)$
$i.e. Rough Shifting$

$Select$: $Continuing\ by\ 5.\ which\ says$
$that\ Continuing > Retaining$
$> Smooth\ Shifting > Rough\ Shifting$

11. The reasoning of U_5

$U5$: $He\ was\ sick\ and\ furious\ at\ being\ woken\ up\ so\ early.$

$Construct: Cf$: $a.<he = TERRY>\ or\ b.<he = TONY>$

Cb: $<he = TERRY> or <he = TONY>$

Cp: $<he = TERRY> or <he = TONY>$

$Filter$: $by, he\ refers\ to\ Cb;$

$Classify$:

1) Cb: $he = TERRY$

$so,$ $Cb(U5) = Cb(U4), Cb(U5) = Cp(U5)$

$i.e.$ $Continuing$

2) Cb: $he = TONY$

$so,$ $Cb(U4) \eta Cb(U3), Cb(U4) = Cp(U4)$

$i.e.$ $SmoothShifting$

$Select$: $Continuing\ by\ 5,\ which$
$says\ that\ Continuing > Smooth\ Shifting$

As we see, this mechanism works well in the first four utterances. It can even resolve ambiguities when two pronouns appear in one utterance. But the last one seems doubtful. We cannot say the predication is wrong. Yet, $smoothshifting$ is not wrong either in actual languages. Further, $smoothshifting$ seems to be a more reasonable choice. In BFP (Brennan, Friedman et al. 1987), they also notice a similar problem where an utterance after a $retaining$, gets $continuing$ based on the algorithm but actually should be more like a $smooth-shifting$. Their proposal to the solution is to add a constraint to the computation system as a retention may be a signal of an impending shift. This constraint doesn't apply in this case. Nonetheless, we may add a distance constraint to reduce the priority of $continuing$ here.

Up to now, centering theory is still an active model used in coreference resolution. Recently, experiments have shown that narrowing search space by adding constraints is the most important element when investigating the preference for coreference resolution. Yet, the search space is still large for centering approach. Further, many details need specifications, such as how to rank the order of antecedents, what is an utterance and how to determine utterance boundaries. These details are varied across different languages.

2.2.1 Givenness Hierarchy

Gundel, Hedberg and Zacharski [1] propose Givenness hierarchy theory to explain the distribution and interpretation of noun phrase forms in natural language discourse. A prime assumption of this work is that "different determiners and pronominal forms signal different cognitive statuses (information about location in memory and attention state)". They propose six cognitive statuses relevant for explicating the form of referring expressions in natural discourse which are implicationally related in the Givenness hierarchy shown in Figure 2-2.

in focus	activated	familiar	uniquely identifiable	referential	type identifiable
{it}	*that* *this* *this N*	{that N}	{the N}	{indefinite this N}	{a N}

Table 2.2: Givenness Hierarchy

The pronominal or determiner forms are intended to signal that the referent of the nominal expression is assumed by the speaker or writer to have a particular cognitive status (memory and attention state) for the addressee. Each status on the hierarchy is a necessary and sufficient condition for appropriate use of different pronoun or determiner. It is an implicational scale, which means that whenever the speaker uses a specific linguistic form he/she would be implicating all other states which rank lower in the hierarchy. In addition, the Givenness Hierarchy interacts with Grice's maxims of quantity [2] by virtue of the fact that the Givenness Hierierarchy is an implicational scale.

- Q1: Make your contribution as informative as possible
- Q2: Do not make your contribution more informative than necessary

The interactions of the maxims of quantity with the hierarchy often give rise to scalar implicatures [3], whereby use of a form that explicitly encodes a lower status implicates that the status encoded by a stronger (entailing) form does not hold (much as use of 'some; often implicates 'not all')

Again take sample two for illustration. There are two persons $Terry$ and $Tony$ who are mentioned. In (b) ,(c) and (d) , the Givenness hierarchy predicts that the pronouns he and

his refer to Terry given that it is reasonable to assume that the readerÕs attention is focused on *Terry* at this point in the discourse, as he was introduced in subject position in (a) and has been mentioned in each of the previous sentences. The second pronoun *him* in (d) is likely to refer to Tony, because Tony was also introduced in a relatively prominent syntactic position (object) in the previous sentence, and is therefore potentially in focus, and *him* and *he* cannot corefer due to binding conditions.

The Givenness Hierarchy framework does not predict a unique interpretation in all cases, consistent with the fact that interpretation of natural language (and reference/pronoun resolution in particular) is often ambiguous, and is consistent with more than one possible interpretation. An adequate linguistic theory must be able to predict when such ambiguities/indeterminacies arise; it cannot deterministically resolve them in all cases, any more than people can. In this case, assuming that Terry and Tony are both in focus at the point when the reader encounters sentence (d), either of the pronouns can refer to either *Terry* or *Tony*. If the subject *He* refers to *Terry* , then the object *him* should refer to *Tony* and vice versa (again, given binding conditions). The interpretation where *He* refers to *Terry* and *him* refers to *Tony* is probably the preferred interpretation, based on two factors: (a) since *Terry* is more strongly in focus because *he* has been mentioned in all the previous sentences in this paragraph, *Terry* is the more likely interpretation of the subject pronoun *he* as this is the the first pronoun encountered in sentence (d), leaving *him* to refer to *Tony*, and (b) *Terry* is the more pragmatically plausible interpretation for the subject pronoun *he*, since it is more likely that *Terry* would have called *Tony* than vice-versa, given the semantic content of the previous sentence.

In an automatic system, heuristic strategies or probabilistic models with the help of world knowledge can thus help in resolving such ambiguities to some degree.

Despite some limitations, the Givenness Hierarchy has been widely used in computational models of discourse and reference resolution as it provides insight into the cognitive basis for distribution and interpretation of different referring forms, including pronouns and yields correct predictions in many cases [4].

McCoy and Strube [5] proposes to build an S-List Ranking which is based on information structural concepts for coreference resolution. Though Strube's information structural concepts are not based on the Givenness hierarchy, they can be regarded as a variation of the

Givenness hierarchy. In his paper, he distinguishes between three different sets of expressions, hearer-old discourse entity (*OLD*), mediated discourse entity (*MED*), and hearer-new discourse entity (*NEW*). They are ranked by givenness shown as the following figure.

Figure 2.2: S-list Ranking and Familiarity

As we see, there are some subcategories for OLD and NEW. Under OLD are two subsets E (evoked) and U (unused); under MED are three subsets I (inferrables), I^C(containing inferrables) and BNA (anchored brand-new); and under NEW is BN (brand-new). These categories roughly correspond to the Givenness hierarchy classes as follows.

OLD			MED			NEW
E		U	I	I^C	BN^A	BN
In focus	Activated		maybe Familiar	Uniquely identifiable	Referential	Type identifiable

Table 2.3: Correspondence between GH and SList

We can see that there are not one-to-one correspondences between them two. $Evoked$ in S-List may be status of in focus or activated. There is no correspondent category for unused in the Givenness hierarchy. Unused may be discourse-new though it may be hearer-old. Inferable may be familiar but not always. Indeed, the Givenness hierarchy and S-List ranking have some essential differences. Firstly, statuses in the Givenness hierarchy entail others while categories in S-List are mutually exclusive; secondly, there are no differences between activate and in focus in S-List; thirdly, S-List distinguishes extra-linguistic categories while the Givenness hierarchy only includes categories of cognitive statuses.

(Is this paragraph related to the Givenness hierarchy or Strube's algorithm?) Besides the ranking, there are a few other ranking constraints. They are defined as a 3-tuple (x, uttx, posx) where x is a discourse entity which is evoked in utterance uttx at the text position posx. With respect to any two discourse entities (x,uttx,posx) and (y, utty, posy), uttx and utty specify the current utterance Ui or the preceding utterance Ui-1. The following is the table for the constraints (Strube 1998).

(1)	if $x \in OLD\ and\ y \in MED, then\ x \prec y$ if $x \in OLD\ and\ y \in NEW, then\ x \prec y$ if $x \in MED\ and\ y \in NEW, then\ x \prec y$
(2)	if $x, y \in OLD\ or\ x,\ y \in MED, then\ x,\ y \in NEW$ then if $utt_x \succ utt_y, then\ x \prec y$ if $utt_x = utt_y\ and\ pos_x < pos_y, then\ x \prec y$

Table 2.4: Ranking Constraints on the S-List

Then, the algorithm which processes a text from left to right goes like the following:

1. If a referring expression is encountered,
   (a) If it is a pronoun, test the elements of the S-list in the given order until the test succeeds;
   (b) Update S-list; the position of the referring expression under consideration is determined by the S-list-ranking criteria which are used as an insertion algorithm.
2. If the analysis of utterance U is finished, remove all Des from the S-list, which are not realized in U.

Based on this algorithm, we can build a table to see how the above coreference is resolved.

The analysis for sample 2 is given in Table 2.5 Sample with S-List Algorithm. The preferences for pronouns are given by the S-List immediately above them. The pronoun *he* in *b* is resolved to $Terry$ since $Terry$ has the status of $Evoked$ as in focus in the Givenness hierarchy which has higher status than $a\ beautiful\ day$. In *c*, *He* is resolved to $Terry$ for the same reason. Then, him is still resolved to be $Terry$ since $Tony$ is $Unused$ which is lower than *he* which is $evoked$. Like the Givenness hierarchy, for *d* and *e*, ambiguities arise.

a	Terry really goofs sometimes S: [$TERRY_U$: Terry]
b	Yesterday was a beautiful day and he was excited about S: [$A-BEAUTIFUL-DAY_{BN}$, $TERRY_E$:he] trying out his new sailboat S: [$A-BEAUTIFUL-DAY_{BN}$, $TERRY_E$:his]
c	He wanted S:[$TERRY_E$:he] Tony S: [$TERRY_E$:he, $TONY_U$:Tony] to join him S: [$TERRY_E$:him,$TONY_U$:him]
d	He called S: [$TERRY_E$:He, $TONY_E$:He] him at 6AM S:[$TERRY_E$:him,$TONY_E$:Him]
e	He was sick and furious at being waken up so early S:[$TERRY_E$:He,$TONY_E$:He]

Table 2.5: Sample with S-List Algorithm

Both $Terry$ and $Tony$ are $evoked$. In principle, He in d and e can be either $Terry$ or $Tony$. Correspondingly, him in d can be either $Tony$ or $Terry$.

Illustrated from the above example, we can see how the Centering Theory and the Giveness hierarchy interact and complement with each other. Both theories contribute to coreference resolution. Establishing dependency models between them and embedding them into a coreference resolver hold the potential to improve the performance of a coreference resolver.

2.3 Machine Learning Approaches

Machine learning approaches have been applied to a variety of NLP tasks. Coreference resolution is no exception. Machine learning provides statistical models to either train classifiers or cluster data. Any machine learning approach consists of two main tasks. Task one focuses on creating the data off which the learner will deduce the statistical distributions. Task two consists of training the model on the data and testing it on a held-out set.

2.3.1 Predictions of Antecedents as Random Variables

Using statistical models, we can predicate unknown variables with a series of known variables, observed variables or constants. For coreference resolution, the unknown random variable can be the candidate antecedent for a given pronoun. Ge, Hale and Charniak (Ge, Hale et al. 1998) introduced Naive Bayes in coreference resolution to make such a prediction. The probabilistic model includes several syntactic and semantic features which affect pronoun resolution. Following is the description of the features:

- Distance between the pronoun and the candidate antecedent (closer ones are preferred)
- Syntactic structure. When resolving binding constraints, such as the contra-indexing constraints described in Hobbs' algorithm, we need to know the syntactic structures.
- Agreement constraints such as gender, number and animacy constraints. They can be implemented based on the actual words that occur.

- Mention count. Noun phrases that occur repeatedly get more preference. Probability that a proposed antecedent is correct given that it occurs a certain number of times, is computed.

Mention count as a feature was firstly used in centering theory, according to which, a continued topic is the highest ranked referent candidate for a pronoun. However, among different transitions, locality and preference may not be directly modeled here. In order to extract syntactic structure and distance, a modified version of the Hobbs algorithm is used to compute distance between a pronoun and a proposed antecedent. The Hobbs algorithm also provides the antecedents for which the probability of the antecedent being the correct antecedent for the pronoun is computed.

Another probabilistic model, conditional random fields (CRFs), has also been used for the inference and computation of coreference. Specifically, three models based on CRFs are proposed by McCallum and Wellner (McCallum and Wellner 2005). The first one does not restrict the dependency structure. It considers the coreference decisions and the attributes of entities as random variables, conditioned on the entity mentions and the feature functions depend on the coreference decisions y, the set of attributes, as well as the mentions of the entities, x. In the second model, the dependencies of the coreference variable, y, are replaced with a binary valued random variable, Y_{ij} for every pair of mentions. In addition, the clique potentials are restricted to only pairs of mentions. Further, in order to avoid cyclic coreference errors, some term is added. The third model does not include attributes as a random variable. It is very similar to the second model. It is reported that it behaves a little better than Ng and Cardie's (Ng and Cardie 2002) approach. The F1 results on NP coreference of the MUC-6 dataset are about 73%.

Since CRFs are a time-series model which evolved from HMMs, it can take care of transitive dependencies. For example, if a mention "Mr. Powell" and another mention "Powell" are coreferring, then the chances of "Powell" and "she" coreferring will be very low. To assist in this process, an additional term is included in the conditional for y that considers all possible triangle relations, with very high negative weights.

The inference problem is analogous to graph partitioning, with an unknown number of partitions. The Correlation Clustering algorithm (Bansal, Blum et al. 2002) is used to approximate the graph partitioning problem, which works by measuring the inconsistency incurred

by including a node in a partition and minimizing the disagreements. In Sutton and McCamllum (Sutton and McCamllum 2006), a skip-chain CRF is introduced for the purpose of coreference on proper names, as a part of an information task. An improved approach is presented in Finkel et al. (Finkel, Grenager et al. 2005), which uses long-distance features and Gibbs' sampling for inference. According to Wellner et al. (Wellner, Macallum et al. 2004), CRFs present a natural framework to integrate named entity extraction and coreference resolution of proper names. However, it is not clear if an integrated model will be useful for resolution of other types of noun phrases.

2.3.2 A Pairwise Classification Task

Coreference resolution can be cast as a pairwise classification task. Since McCarthy and Lehnert (Mccarthy and Lehnert 1996) first applied decision trees algorithm to it, coreference resolution has made great progress under the framework of pairwise classification. The classification question is whether two markables corefer or not. That is, those markables which corefer belongs to one class and those which do not corefer to another class. A markable could be a noun phrase or a pronoun. All possible markables are identified during preprocessing steps. Then, a separate clustering mechanism then coordinates the possibly contradictory pairwise classifications and constructs a partition on the set of NPs.

Figure 2-4 illustrates what the classification framework looks like. In that figure, bracket numbers represent steps of the classification cycle. Step 1 collects the subjects of the coreference resolver. These inputs are mentions which have been preprocessed. Related preprocessing includes sentence boundary marking, POS tagging, named entities recognition, nested noun phrase recognition etc. Step 2 generates training and testing data from all mentions. In this process, we need to consider the way to generate both positive and negative samples. Three methods have been developed in this aspect. McCarthy and Lehnert (Mccarthy and Lehnert 1996) put mentions which are not in the same coreference chain as negatives while putting mentions which are in the same coreference chain as positives. A disadvantage of this way is that a huge amount of training data will be generated where negatives are much more than positives, thus resulting in an unbalanced training dataset.

Therefore, many investigators pursued alternative approaches. Soon, et al. (Soon, Ng et al. 2001) put two nearest mentions i and j in the coreference chain as positives while other mentions between them two which are not in the coreference chain as negatives. This way,

there will be much less training data and the locality of coreference resolution has been taken into consideration. The third way is that of Ng and Cardie (Ng and Cardie 2002). Different from Soon's method, when creating positives, for a mention j which is in the coreference chain, if j is a pronoun, the positive pair is formed by including the one in the coreference chain which is the nearest antecedent to j. If j is not a pronoun, the positive pair is formed by including the one in the coreference chain which is the nearest antecedent but which is not a pronoun. The way to create the negative pair is the same as Soon's. In testing, usually we can compose a pair by randomly selecting two mentions from a text or we may set up some preference conditions to filter some pairs.

Step 3 deals with feature extraction. In a pair classification framework, how to design selected features plays a key role in the robustness of the coreference resolver. NLP is actually a strongly ill-posed problem. Only large amount of constraints including knowledge and various rules can make ill-posed problems become well-posed and solvable. Natural languages are full of uncertainty. Therefore, processing human languages is an ill-posed problem which can only be solvable with rich constraints such as knowledge, contexts or experiences. During training decision-trees are used to learn rules based on different features computed on pairs of markables. Similar features to variable predictions are employed. Besides, features other than syntactic information or features which, though have the same name, have different definitions are extracted as well.

- Distance feature: Distance between the two markables in terms of number of sentences is used.
- Agreement features: gender and number agreement
- Type of markable: The grammatical category of the markable, namely, demonstrative noun phrase, definite noun phrase, pronoun, reflexive pronoun, and proper noun.
- Semantic class agreement: The semantic class agreement feature which basically checks if the semantic class of the two markables, agree according to the WordNet hierarchy.
- Alias feature: Two markables have a positive alias feature if they share the same name, or if one markable has just the last name and the other has the complete name, or if one is the acronym of the other.

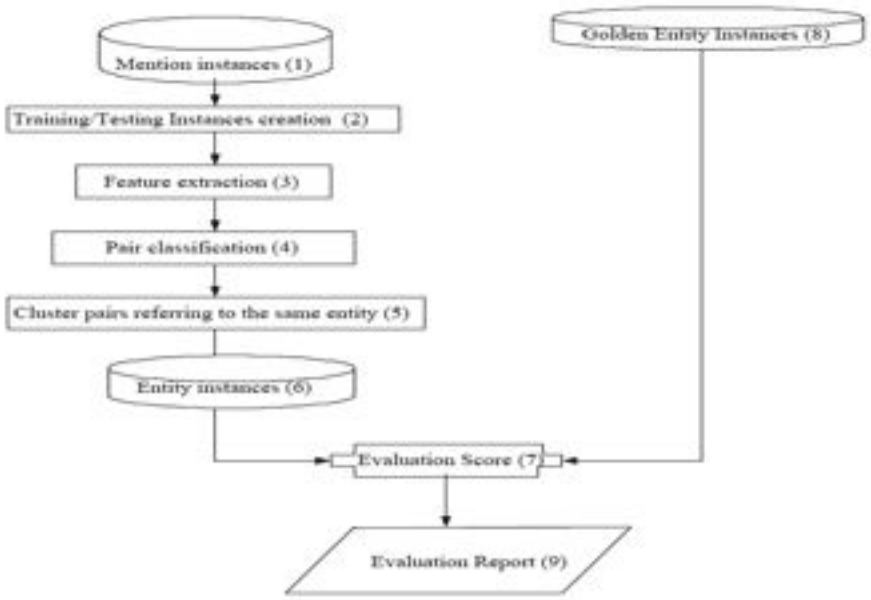

Figure 2.3: The classical framework of pair classification for coreference resolution

McCarthy and Lehnert (Mccarthy and Lehnert 1996) employed 8 simple features while Soon, et al. (Soon, Ng et al. 2001) employed 12 features. Most of those features were acquired by simply processing character strings. Ng and Cardie (Ng and Cardie 2002) extend the feature set and include features that consider the grammatical role of the NPs (subject/object) and a lot of heuristics. About 41 features were added. Later, Ng and Cardie (Ng and Cardie 2003) further extend the model by including a separate classifier for determining if a noun phrase is anaphoric or not. The classifier is a maximum entropy model. The overall coreference classifier uses the same decision-tree learning framework. However, an increase in the number of features does not necessarily translate into a substantial performance gain. In fact, if the corpus is too small, more features do not mean better results since more features with smaller corpus lead to insufficient training of related parameters in the feature space and overtraining. Consequently, it leads to data sparsity and leads to good testing results in the closed data but unsatisfactory results in an open testing data. Hoste and Daelemans (Hoste and Daelemans 2005) proved that feature filtering is useful.

Step 4 is the core of the classification. Various machine learning models (discussed above) have been used. Their characteristics are to train weights for features, parameters

or preferences after extracting features and forming a feature vector. Step 5 groups together pairs which refer to the same entity. There are three main ways to reach this goal. Nearest clustering is to regard the nearest candidate as the antecedent of current pro-forms (Soon, Ng et al. 2001; Strube, Rapp et al. 2002); optimum clustering is to regard the one which has the highest probability as the antecedent (Ng and Cardie 2002; Iida, Inui et al. 2003); maximum clustering is to cluster all candidates and regard all of them as antecedents (Mccarthy and Lehnert 1996). After clustering, we can get the entities which will be used for evaluation. That is what (6) shows. The next two steps are the evaluation (7) and the evaluation results (8). In the above framework, training samples are generated from the corpus automatically. The positive samples are generated for immediate adjacent noun phrase pairs while negative samples are generated by using pairs that are not marked as coreferent.

2.4 Challenges with the Current Coreference Resolution Models

Despite the variety of approaches to coreference resolution, we can summarize the mainstream algorithm in two steps. Step one uses the classic classification model to calculate the coreference probability of two entities. Step two clusters coreference chains based on various methods. Most of the systems focus on discovering whether two mentions in the text refer to the same entity. In such a pair classification framework, discriminative models are used to train a resolver with feature vectors, such as distance, shallow syntactic analysis (Soon, Ng et al. 2001; Ng and Cardie 2002). Although these methods yield good results, there are two main outstanding issues.

2.4.1 Mistakes and Transitivity Errors due to the Pair Classification Framework

The recognition of referring expressions is potentially determined by a pair classification and thus may easily lead to errors including false negatives. If the output coefficient is higher than a threshold, the pair is judged as coreference pair. Otherwise, the pair is judged as non-coreferential. Two problems may arise then:

- the system may mistakenly point an antecedent to an entity description of a non-coreferring expression;

- the system may not carry out a coreference resolution for an entity description of a referring expression

For identity coreference, it is easy to make such errors as clustering $Mr.Powell$ and She. Coreference resolution is essentially a process of clustering or dividing mentions into different coreferent pairs. For identity coreference, if we have a coreference pair <a, b> and another coreference pair <b, c>, then a, c will be a coreference pair as well. However, classic clustering uses a greedy algorithm to cluster coreference pairs from left to right. Then, if $Mr.Powell$ and $Powell$ are a pair and $Powell$ and She are a pair, then in clustering, $Mr.Powell$ and She will be a pair as well. This is incorrect apparently.

2.4.2 Linguistic and World Knowledge

Although the exploration of new mathematical models is quite important, equally important is to employ discriminative features based on sound scientific theories. Yet, since middle 90s, research on coreference resolution has focused exclusively on "knowledge-poor" methods. An exception to this is the Hobbs Algorithm and the Centering theory. However, they focus on one aspect of language analysis. Recent approaches are machine-learning oriented without much linguistic information. The reason leading to such a trend is due to the difficulty in acquiring deep linguistics knowledge. Undoubtedly, some Òknowledge-poor" systems show good results with shallow morphological analysis modules. Kehler, et al (Keller 1988; Kehler, Appelt et al. 2004) emphasize that we have to make use of deep linguistics knowledge to advance the system quality of coreference resolution. It is not hard to imagine that not all coreference can be determined by string matching or syntactic parsing techniques. In the following examples, a pronoun resolution system must determine what the pronoun his refers to:

- John needs his friend.
- John needs his support.

In (a), $John$ and his may corefer. In (b), his refers to some other, perhaps previously evoked entity. But for a traditional pronoun resolution, systems are not designed to distinguish between these cases. They lack the world knowledge required in the second instance

Ð the knowledge that a person does not usually explicitly need his own support. Evidently, we cannot resolve the coreference without deeper understanding of the language itself.

2.4.3 Linguistics Model and Machine Learning Models

Another big problem for various approaches is the separation between linguistic approaches and machine learning approaches. Without doubt, machine learning models have employed linguistic features since middle of 1990s. Yet, linguistic models are not really merged with machine learning models very well. Two reasons lead to this situation: firstly, deeper language processing is hard, and secondly, linguistic models and machine learning models are often incompatible. The first difficulty has been demonstrated in last section. Languages are full of ambiguity in each linguistic level. The second one seems to reflect the different thinking patterns between linguists and statistician.

Chapter 3

Factorial HMM for Coreference Resolution

3.1 Introduction

Pronoun anaphora resolution is the task of finding the correct antecedent for a given pronominal anaphor in a document. It is a subtask of coreference resolution, which is the process of determining whether two or more linguistic expressions in a document refer to the same entity. Adopting terminology used in the Automatic Context Extraction (ACE) program [6], these expressions are called mentions. Each mention is a reference to some entity in the domain of discourse. Mentions usually fall into three categories – proper mentions (proper names), nominal mentions (descriptions), and pronominal mentions (pronouns). There is a great deal of related work on this subject, so the descriptions of other systems below are those which are most related or which the current model has drawn insight from.

Pairwise models [7; 8] and graph-partitioning methods [9] decompose the task into a collection of pairwise or mention set coreference decisions. Decisions for each pair or each group of mentions are based on probabilities of features extracted by discriminative learning models. The aforementioned approaches have proven to be fruitful; however, there are some notable problems. Pairwise modeling may fail to produce coherent partitions. That is, if we link results of pairwise decisions to each other, there may be conflicting coreferences. Graph-partitioning methods attempt to reconcile pairwise scores into a final coherent clustering, but they are combinatorially harder to work with in discriminative approaches.

One line of research aiming at overcoming the limitation of pairwise models is to learn a mention-ranking model to rank preceding mentions for a given anaphor [10] This approach results in more coherent coreference chains.

Recent years have also seen the revival of interest in generative models in both machine learning and natural language processing. Haghighi and Klein [11], proposed an unsupervised non-parametric Bayesian model for coreference resolution. In contrast to pairwise models, this fully generative model produces each mention from a combination of global entity properties and local attentional state. Ng [12] did similar work using the same unsupervised generative model, but relaxed head generation as head-index generation, enforced agreement constraints at the global level, and assigned salience only to pronouns.

Another unsupervised generative model was recently presented to tackle only pronoun anaphora resolution [13]. The expectation-maximization algorithm (EM) was applied to learn parameters automatically from the parsed version of the North American News Corpus [14]. This model generates a pronoun's person, number and gender features along with the governor of the pronoun and the syntactic relation between the pronoun and the governor. This inference process allows the system to keep track of multiple hypotheses through time, including multiple different possible histories of the discourse.

Haghighi and Klein [15] improved their non-parametric model by sharing lexical statistics at the level of abstract entity types. Consequently, their model substantially reduces semantic compatibility errors. They report the best results to date on the complete end-to-end coreference task. Further, this model functions in an online setting at mention level. Namely, the system identifies mentions from a parse tree and resolves resolution with a left-to-right sequential beam search. This is similar to Luo [16] where a Bell tree is used to score and store the searching path.

In this chapter, we present a supervised pronoun resolution system based on Factorial Hidden Markov Models (FHMMs). The success of recent unsupervised approaches may raise the question about the value of new supervised generative pronoun resolution systems. However, this model is distinct from most other work in pronoun resolution. Above all, while not attempting to model psycholinguistic processing explicitly, this system is motivated by human processing concerns, by operating incrementally and maintaining a limited short term memory for holding recently mentioned referents. Second, like Morton [17], the current system essentially uses prior information as a discourse model with a time-series

manner, using a dynamic programming inference algorithm. Third, the FHMM described here is an integrated system, in contrast with [15]. The model generates part of speech tags as simple structural information, as well as related semantic information at each time step or word-by-word step. While the framework described here can be extended to deeper structural information, POS tags alone are valuable as they can be used to incorporate the binding feature (described below).

Although the system described here is evaluated for pronoun resolution, the framework we describe can be extended to more general coreference resolution in a fairly straightforward manner. Further, as in other HMM-based systems, the system can be either supervised or unsupervised. However, these extensions are left for future work.

The final results are compared with the mention-ranking model [10] and systems compared in their paper. The FHMM-based pronoun resolution system does a better job than the global ranking technique and other approaches. This is a promising start for this novel FHMM-based pronoun resolution system.

3.2 Theoretical Foundations

A new system of coreference resolution will be proposed in this thesis. The system aims at building a coreference resolver based on a referential semantic hierarchic hidden Markov model (RSHHMM). The resolver will merge linguistic models and also absorb various linguistic features. In order to make the model be feasibly established, I plan to start from HMMs and basic linguistic features and incrementally expand it into a final RSHHMM. In this section, I will introduce the main theories behind my proposal.

3.2.1 N-Grams Models

N-grams is a probabilistic model to represent word sequences. It can be used to compute the probability of an entire sentence or to give a probabilistic predication of what the next word will be in a sequence. As we know, sentences are composed of string of words which go together by grammatical rules. Namely, there are rules to determine which word goes with which word. Then, an intuition is that some words co-occur more frequently with some words. Therefore, we can calculate the probability of a word sequence given another word

sequence. That is how an N-gram model is derived starting from the calculation of the probability of a whole sentence or a whole string of words. Let us represent it as w1...wn or as wn1 . If we consider each word occurring in its location as an independent event, we can represent this probability as P(wn1). Then, use chain rule of probability and we can decompose this probability as:

$$P(w_1^n) = P(w_1)P(w_2|w_1)P(w_3|w_1^2)...P(^{n-1}w_n) = \prod_{k=1}^{n} w_k w_1^{k-1} \tag{3.1}$$

In this formula, a problem is how to compute probabilities like P(wn|wn?11). This is not easy if we compute it directly. Whereas, a simplification is that one word may only depend-dend on its closer words. Namely, a string of words may be counted as a Markov process. If we only consider the previous word, it is called a bigram model. If we consider previous n words, it is called a n-gram model. This way, P(wn|wn?11) converts to P(wn|wn?1). Take the sentence fragment Mary who is always proud thinks that she is as an example. We want to predict what next word is. In this context, smart should be a more reasonable word than thinks.

Only based on the sentence, we need to compute P(smart|Mary who is always proud thinks that she is) and P(thinks|Mary who is always proud thinks that she is). But with bigram model, we only need to compute P(smart|is) and P(thinks|is). This simplification not only simplifies the computation. Intuitively, it is more reasonable to compute this way unless we have a huge corpus which includes all possible string of words. Evidently, such a corpus is so large that it is impossible to set it up and further, impossible to do such computations either. With a bigram model, the computation of a probability for a sentence with n words can be converted as follows:

$$P(w_1^n) \approx \prod_{k=1}^{n} P(w_k|w_{k-1}) \tag{3.2}$$

Then, the probability of the whole sentence Mary thinks that she is smart can be simply computed as follows:

$$\begin{aligned} P(Mary\ think\ that\ she\ is\ smart) = P(Mary|<s>)P(thinks|Mary) \\ P(that|thinks)P(she|that)P(is|she)P(smart|is) \end{aligned} \tag{3.3}$$

We can see that the bigram model is actually a first order Markov process. If we build it into broader graph model which involves observed states and hidden states, a hidden Markov model will be generated.

3.2.2 Hidden Markov Models

Hidden Markov Models (HMM) is a generative model which has been widely used in speech recognition. It is a time-series model. In speech recognition, it takes spectral input from frame to frame. Each frame is an observed node. The whole spectral sequences are observed sequences ot. Based on the observed sequences, HMMs predict the sequences of hidden states ht . A most likely sequence of hidden states h1..T can then be hypothesized given any sequence of observed states o1...T. The predication is made based on Bayes' Law and Markov independence.

$$P(o_{1...T}|h_{1...T}) = \prod_t \hat{P}_{\Theta A}(o_t|h_t) \tag{3.4}$$

After we get the full probability, we can select the corresponding label of the probability value and get the most likely sequence of phones.

$$P(h_{1...T}) = \prod_t \hat{P}_{\Theta A}(h_t|h_{t-1}) \tag{3.5}$$

$$\begin{aligned}\hat{P}_{1...T} &= argmax_{h_{1...T}} P(h_{1...T}|o_{1...T}) \\ &= argmax_{h_{1...T}} P(h_{1...T}) \cdot P(o_{1...T}|h_{1...T}) \\ &\doteq argmax_{h_{1...T}} \prod_{t=1}^{T} \hat{P}_{\Theta A}(h_t|h_{t-1}) \cdot \hat{P}_{\Theta B}(o_t|h_t)\end{aligned} \tag{3.6}$$

A sequence of phones is in essence the same as a sequence of words. As we see, the Markov independence assumption makes us multiply a chain of transition model. This is a bigram model itself. For such a simple HMM, it can be applied to a variety of task Ð recognizing phones, part-of-speech (POS) tagging, named entity recognition and even coreference resolution as well. In my thesis, I will use a simple HMM (with factored hidden states including POS tagging and word bigrams) to resolve coreference. Its result will be used as a baseline. More complicated model will be developed.

3.3 Model Description

This work is based on a graphical model framework called Factorial Hidden Markov Models (FHMMs). Unlike the more commonly known Hidden Markov Model (HMM), in an FHMM the hidden state at each time step is expanded to contain more than one random variable (as shown in Figure 3.1). This allows for the use of more complex hidden states by taking advantage of conditional independence between substates. This conditional independence allows complex hidden states to be learned with limited training data.

3.3.1 Factorial Hidden Markov Model

Factorial Hidden Markov Models are an extension of HMMs [18]. HMMs represent sequential data as a sequence of hidden states generating observation states (words in this case) at corresponding time steps t. A most likely sequence of hidden states can then be hypothesized given any sequence of observed states, using Bayes' Law (Equation 3.8) and Markov independence assumptions (Equation 3.9) to define a full probability as the product of a Transition Model (Θ_H) prior probability and an Observation Model (Θ_O) likelihood probability.

$$\hat{h}_{1..T} \stackrel{\text{def}}{=} \underset{h_{1..T}}{\operatorname{argmax}} \mathsf{P}(h_{1..T} \mid o_{1..T}) \tag{3.7}$$

$$\stackrel{\text{def}}{=} \underset{h_{1..T}}{\operatorname{argmax}} \mathsf{P}(h_{1..T}) \cdot \mathsf{P}(o_{1..T} \mid h_{1..T}) \tag{3.8}$$

$$\stackrel{\text{def}}{=} \underset{h_{1..T}}{\operatorname{argmax}} \prod_{t=1}^{T} \mathsf{P}_{\Theta_T}(h_t \mid h_{t-1}) \cdot \mathsf{P}_{\Theta_O}(o_t \mid h_t) \tag{3.9}$$

For a simple HMM, the hidden state corresponding to each observation state only involves one variable. An FHMM contains more than one hidden variable in the hidden state. These hidden substates are usually layered processes that jointly generate the evidence. In the model described here, the substates are also coupled to allow interaction between the separate processes. Thus, the transition model expands the left term in (3.9) to (3.10) if hidden states include three sub-states as Figure 3.1 shows.

$$\begin{aligned} \mathsf{P}_{\Theta_T}(h_t \mid h_{t-1}) \stackrel{\text{def}}{=} & \mathsf{P}(op_t \mid op_{t-1}, pos_{t-1}) \\ & \cdot \mathsf{P}(cr_t \mid cr_{t-1}, op_{t-1}) \cdot \\ & \mathsf{P}(pos_t \mid op_t, pos_{t-1}) \end{aligned} \tag{3.10}$$

The observation model expands from the right term in (3.9) to (3.11).

$$\mathsf{P}_{\Theta_O}(o_t \mid h_t) \stackrel{\text{def}}{=} \mathsf{P}(o_t \mid pos_t, cr_t) \tag{3.11}$$

The observation state depends on more than one hidden state at each time step in FHMMs. Each hidden variable can be further split into smaller variables. What these terms stand for and the motivations behind the above equations will be explained in the next section.

3.3.2 Modeling a Coreference Resolver with FHMMs

FHMMs in our model, like standard HMMs, cannot represent the hierarchical structure of a syntactic phrase. In order to partially represent this information, the head word is used to represent the whole noun phrase. After coreference is resolved, the coreferring chain can then be expanded to the whole phrase with NP chunker tools.

In this system, hidden states are composed of three main variables: a referent operation (*OP*), coreference features (*CR*) and part of speech tags (*POS*) as displayed in Figure 3.1. The transition model is defined as Equation 3.10.

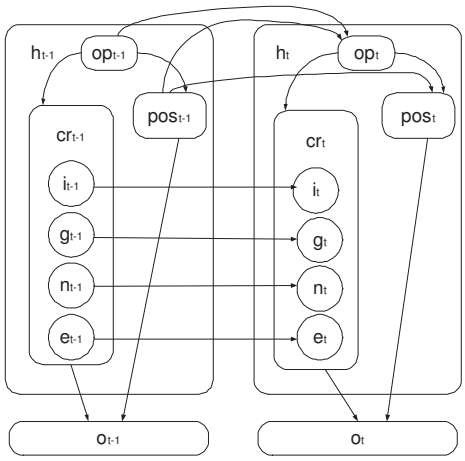

Figure 3.1: Factorial HMM CR Model

The starting point for the hidden state at each time step is the *OP* variable, which determines which kind of referent operations will occur at the current word. Its domain has three possible states: *none, new* and *old*.

The *none* state indicates that the present state will not generate a mention. All previous hidden state values (the list of previous mentions) will be passed deterministically (with

probability 1) to the current time step without any changes. The *new* state signifies that there is a new mention in the present time step. In this event, a new mention will be added to the entity set, as represented by its set of feature values and position in the coreference table. The *old* state indicates that there is a mention in the present time state and that this mention refers back to some antecedent mention. In our pronoun resolution system, all pronouns are defined as old while all other mentions are defined as new. After we expand the model to a general coreference resolution system, we will release the constraints to assign old to non-anaphor mentions. In such a case, the list of entities in the buffer will be reordered deterministically, moving the currently mentioned entity to the top of the list.

Notice that op_t is defined to depend on op_{t-1} and pos_{t-1}. This is sometimes called a *switching* FHMM [19]. This dependency can be useful, for example, if op_{t-1} is *new*, in which case op_t has a higher probability of being *none* or *old*. If pos_{t-1} is a verb or preposition, op_t has more probability of being *old* or *new*.

One may wonder why op_t generates pos_t, and not the other way around. This model only roughly models the process of (new and old) entity generation, and either direction of causality might be consistent with a model of human entity generation, but this direction of causality is chosen to represent the effect of semantics (referents) generating syntax (POS tags). In addition, this is a joint model in which POS tagging and coreference resolution are integrated together, so the best combination of those hidden states will be computed in either case.

3.3.3 Coreference Features

Coreference features for this model refer to features that may help to identify co-referring entities.

In this chapter, they mainly include index (*I*), named entity type (*E*), number (*N*) and gender (*G*). The index feature represents the order that a mention was encountered relative to the other mentions in the buffer. The latter three features are well known and described elsewhere, and are not themselves intended as the contribution of this work. The novel aspect of this part of the model is the fact that the features are carried forward, updated after every word, and essentially act as a discourse model. The features are just a shorthand way of representing some well known essential aspects of a referent (as pertains to anaphora resolution) in a discourse model.

Features	Values
I	positive integers from 1…n
G	male, female, neutral, unknown
N	singular, plural, unknown
E	person, location, organization, GPE, vehicle, company, facility

Table 3.1: Coreference features stored with each mention.

Unlike discriminative approaches, generative models like the FHMM described here do not have access to all observations at once. This model must then have a mechanism for jointly considering pronouns in tandem with previous mentions, as well as the features of those mentions that might be used to find matches between pronouns and antecedents.

Further, higher order HMMs may contain more accurate information about observation states. This is especially true for coreference resolution because pronouns often refer back to mentions that are far away from the present state. In this case, we would need to know information about mentions which are at least two mentions before the present one. In this sense, a higher order HMM may seem ideal for coreference resolution. However, higher order HMMs will quickly become intractable as the order increases.

In order to overcome these limitations, two strategies which have been discussed in the last section are taken: First, a switching variable called *OP* is designed (as discussed in last section); second, a memory of recently mentioned entities is maintained to store features of mentions and pass them forward incrementally.

OP is intended to model the decision to use the current word to introduce a new referent (*new*), refer to an antecedent (*old*), or neither (*none*). The entity buffer is intended to model the set of 'activated' entities in the discourse – those which could plausibly be referred to with a pronoun. These designs allow similar benefits as longer dependencies of higher-order HMMs but avoid the problem of intractability. The number of mentions maintained must be limited in order for the model to be tractable. Fortunately, short term memory faces effectively similar limitations and thus pronouns usually refer back to mentions not very far away. Even so, the impact of the size of the buffer on decoding time may be a concern.

Since the buffer of our system will carry forward a few previous groups of coreference features plus *op* and *pos*, the computational complexity will be exorbitantly high if we keep high beam size and meanwhile if each of features interacts with others. Luckily, we have successfully reduced the intractability to a workable system in both speed and space with following methods. First, we estimate the size of buffer with a simple count of average distances between pronouns and their antecedents in the corpus. It is found that about six is enough for covering 99.2% of all pronouns. Secondly, the coreference features we have used have the nice property of being independent from one another. One might expect English non-person entities to almost always have neutral gender, and thus be modeled as follows:

$$\mathsf{P}(e_t, g_t \mid e_{t-1}, g_{t-1}) = \mathsf{P}(g_t \mid g_{t-1}, e_t) \cdot \mathsf{P}(e_t \mid e_{t-1}) \tag{3.12}$$

However, a few considerations made us change the idea. First, exceptions are found from the corpus. Personal pronouns such as *she* or *he* are used to refer to country, regions, states or organizations. Second, existing model files made by Bergsma [20] include a large number of non-neutral gender information for non-person words. We employ these files for acquiring gender information of unknown words. If we use Equation 3.12, sparsity and complexity will increase. Further, preliminary experiments have shown models using an independence assumption between gender and personhood work better. Thus, we treat each coreference feature as an independent event. Hence, we can safely split coreference features into separate parts. This way dramatically reduces the model complexity. Thirdly, our HMM decoding uses the Viterbi algorithm with A-star beam search.

The probability of the new state of the coreference table $\mathsf{P}(cr_t \mid cr_{t-1}, op_t)$ is defined to be the product of probabilities of the individual feature transitions.

$$\begin{aligned} \mathsf{P}(cr_t \mid cr_{t-1}, op_t) = \mathsf{P}(i_t \mid i_{t-1}, op_t) \cdot \\ \mathsf{P}(e_t \mid e_{t-1}, i_t, op_t) \cdot \\ \mathsf{P}(g_t \mid g_{t-1}, i_t, op_t) \cdot \\ \mathsf{P}(n_t \mid n_{t-1}, i_t, op_t) \end{aligned} \tag{3.13}$$

This supposes that the features are conditionally independent of each other given the index variable. Each feature only depends on the operator and the corresponding feature at the previous state, with that set of features re-ordered as specified by the index model.

3.3.4 Feature Passing

Equation 3.13 is correct and complete, but in fact the switching variable for operation type results in three different cases which simplifies the calculation of the transition probabilities for the coreference feature table.

Note the following observations about coreference features: i_t only needs a probabilistic model when op_t is *old* – in other words, only when the model must choose between several antecedents to re-refer to. g_t, e_t and n_t are deterministic except when op_t is *new*, when gender, entity type, and number information must be generated for the new entity being introduced.

When op_t is *none*, all coreference variables (entity features) will be copied over from the previous time step to the current time step, and the probability of this transition is 1.0. When op_t is *new*, i_t is changed deterministically by adding the new entity to the first position in the list and moving every other entity down one position. If the list of entities is full, the least recently mentioned entity will be discarded. The values for the top of the feature lists g_t, e_t, and n_t will then be generated from feature-specific probability distributions estimated from the training data. When op_t is *old*, i_t will probabilistically select a value $1 \ldots n$, for an entity list containing n items. The selected value will deterministically order the g_t, n_t and e_t lists. This distribution is also estimated from training data, and takes into account recency of mention. The shape of this distribution varies slightly depending on list size and noise in the training data, but in general the probability of a mention being selected is directly correlated to how recently it was mentioned.

With this understanding, coreference table transition probabilities can be written in terms of only their non-deterministic substate distributions:

$$\begin{aligned} \mathsf{P}(cr_t \mid cr_{t-1}, old) = \quad & \mathsf{P}_{old}(i_t \mid i_{t-1}) \cdot \\ & \mathsf{P}_{reorder}(e_t \mid e_{t-1}, i_t) \cdot \\ & \mathsf{P}_{reorder}(g_t \mid g_{t-1}, i_t) \cdot \\ & \mathsf{P}_{reorder}(n_t \mid n_{t-1}, i_t) \end{aligned} \tag{3.14}$$

where the *old* model probabilistically selects the antecedent and moves it to the top of the list as described above, thus deciding how the reordering will take place. The *reorder* model actually implements the list reordering for each independent feature by moving the feature value corresponding to the selected entity in the index model to the top of that feature's list.

The overall effect is simply the probabilistic reordering of entities in a list, where each entity is defined of a label and a set of features.

$$\begin{aligned} \mathsf{P}(cr_t \mid cr_{t-1}, new) = \quad & \mathsf{P}_{new}(i_t \mid i_{t-1}) \cdot \\ & \mathsf{P}_{new}(g_t \mid g_{t-1}) \cdot \\ & \mathsf{P}_{new}(n_t \mid n_{t-1}) \cdot \\ & \mathsf{P}_{new}(e_t \mid e_{t-1}) \end{aligned} \tag{3.15}$$

where the *new* model probabilistically generates a feature value based on the training data and puts it at the top of the list, moves every other entity down one position in the list, and removes the final item if the list is already full. Each entity in i takes a value from 1 to n for a list of size n. Each g can be one of four values – *male*, *female*, *neuter* and *unknown*; n one of three values – *plural*, *singular* and *unknown* and e around eight values.

Note that pos_t is used in both hidden states and observation states. While it is not considered a coreference feature as such, it can still play an important role in the resolving process. Basically, the system tags parts of speech incrementally while simultaneously resolving pronoun anaphora. Meanwhile, pos_{t-1} and op_{t-1} will jointly generate op_t. This point has been discussed in Section 3.3.2.

Importantly, the pos model can help to implement binding principles [21]. It is applied when op_t is old. In training, pronouns are sub-categorised into personal pronouns, reflexive and other-pronoun. We then define a variable loc_t whose value is how far back in the list of antecedents the current hypothesis must have gone to arrive at the current value of i_t. If we have the syntax annotations or parsed trees, then, the part of speech model can be defined when op_t is *old* as $\mathsf{P}_{binding}(pos_t \mid loc_t, s_{loc_t})$. For example, if $pos_t \in reflexive$, $\mathsf{P}(pos_t \mid loc_t, s_{loc_t})$ where loc_t has smaller values (implying closer mentions to pos_t) and $s_{loc_t} = subject$ should have higher values since reflexive pronouns always refer back to subjects within its governing domains. This was what [22] did and we did this in training with the REUTERS corpus [23] in which syntactic roles are annotated. We finally switched to the ACE corpus for the purpose of comparison with other work. In the ACE corpus, no syntactic roles are annotated. We did use Stanford parser to extract syntactic roles from the ACE corpus. But the result is largely affected by the parsing accuracy. Again, for a fair comparison, we extract similar features to Denis and Baldridge [10], which is the model we mainly compare with. They approximate syntactic contexts with POS tags surrounding the pronoun. Inspired by this idea, we successfully represent binding features with POS tags before

anaphors. Instead of using $\mathsf{P}(pos_t \mid loc_t, s_{loc_t})$, we train $\mathsf{P}(pos_t \mid loc_t, pos_{loc_t})$ which can play the role of binding. For example, suppose buffer size is 6 and $loc_t = 5$, $pos_{loc_t} = noun$. Then, $\mathsf{P}(pos_t = reflexive \mid loc_t, pos_{loc_t})$ is usually higher than $\mathsf{P}(pos_t = pronoun \mid loc_t, pos_{loc_t})$, since the reflexive has higher probability of referring back to the noun located in 5 than the pronoun.

In future work expanding to coreference resolution between any noun phrases we intend to integrate syntax into this framework as a joint model of coreference resolution and parsing.

3.4 Observation Model

The observation model that generates an observed state is defined as Equation 3.11. To expand that equation in detail, the observation state, the word, depends on its part of speech and its coreference features as well. Since FHMMs are generative, we can say part of speech and coreference features generate the word.

In actual implementation, the observed model will be very sparse since cr_t will be split into more variables according to how many coreference features it is composed of. In order to avoid the sparsity, we transform the equation with Bayes' law as follows.

$$\mathsf{P}_{\Theta_O}(o_t \mid h_t) = \frac{P(o_t) \cdot \mathsf{P}(h_t \mid o_t)}{\sum_{o'} P(o')\mathsf{P}(h_t \mid o')} \tag{3.16}$$

$$= \frac{P(o_t) \cdot \mathsf{P}(pos_t, cr_t \mid o_t)}{\sum_{o'} P(o')\mathsf{P}(pos_t, cr_t \mid o')} \qquad if\ op_t = OLD\ or\ NEW \tag{3.17}$$

$$= \frac{P(o_t) \cdot \mathsf{P}(pos_t \mid o_t)}{\sum_{o'} P(o')\mathsf{P}(pos_t \mid o')} \qquad if\ op_t = COPY \tag{3.18}$$

We define *pos* and *cr* to be independent of each other, so we can further split the above equation as:

$$\mathsf{P}_{\Theta_O}(o_t \mid h_t) \stackrel{\text{def}}{=} \frac{P(o_t) \cdot \mathsf{P}(pos_t \mid o_t) \cdot \mathsf{P}(cr_t \mid o_t)}{\sum_{o'} P(o') \cdot \mathsf{P}(pos_t \mid o') \cdot \mathsf{P}(cr_t \mid o')} \tag{3.19}$$

where $\mathsf{P}(cr_t \mid o_t) = \mathsf{P}(g_t \mid o_t)\mathsf{P}(n_t \mid o_t)\mathsf{P}(e_t \mid o_t)$ and $\mathsf{P}(cr_t \mid o') = \mathsf{P}(g_t \mid o')\mathsf{P}(n_t \mid o')\mathsf{P}(e_t \mid o')$

One problem which needs solution is how to calculate the denominator. In Equation 3.19 o' represents observed words. We need to sum up all $P(o') * P(h_t|o')$. Namely, we need to calculate all mentions which have gender, number, entity, syntactic roles and other features values. If words are not mentions, only their pos will be considered.

In FHMM, we cannot obtain the results of the denominator incrementally since all observed words cannot come out all at once. Therefore, we have to put this calculation into the processing steps and generate a file and read it in as all other trained models.

This change transforms the FHMM to a hybrid FHMM since the observation model no longer generates the data. Instead, the observation model generates hidden states, which is more a combination of discriminative and generative approaches. This way facilitates building likelihood model files of features for given mentions from the training data. The hidden state transition model represents prior probabilities of coreference features while this observation model factors in the probability given a pronoun.

3.4.1 Unknown Words Processing

If an observed word was not seen in training, the distribution of its part of speech, gender, number and entity type will be unknown. In this case, a special unknown words model is used.

The part of speech of unknown words is estimated using a decision tree model. This decision tree is built by splitting letters in words from the end of the word backward to its beginning. The decision tree makes decisions by considering the morphological features of words trained from the corpus. This method is about as accurate as the approach described by Klein and Manning [24].

Next, a similar model is set up for estimating $\mathsf{P}(n_t \mid w_t = unkword)$. Most English words have regular plural forms, and even irregular words have their patterns. Therefore, the morphological features of English words can often be used to determine whether a word is singular or plural.

Gender is irregular in English, so model-based predictions are problematic. Instead, we follow Bergsma and Lin [20] to get the distribution of gender from their gender/number data and then predict the gender for unknown words.

3.5 Evaluation and Discussion

3.5.1 Experimental Setup

In this research, we used the ACE corpus (Phase 2) [1] for evaluation. The development of this corpus involved two stages. The first stage is called EDT (entity detection and tracking) while the second stage is called RDC (relation detection and characterization). All markables have named entity types such as FACILITY, GPE (geopolitical entity), PERSON, LOCATION, ORGANIZATION, PERSON, VEHICLE and WEAPONS, which were annotated in the first stage. In the second stage, relations between named entities were annotated. This corpus include three parts, composed of different genres: newspaper texts (NPAPER), newswire texts (NWIRE) and broadcasted news (BNEWS). Each of these is split into a *train* part and a *devtest* part. For the train part, there are 76, 130 and 217 articles in NPAPER, NWIRE and BNEWS respectively while for the test part, there are 17, 29 and 51 articles respectively. Though the number of articles are quite different for three genres, the total number of words are almost the same. Namely, the length of NPAPER is much longer than BNEWS (about 1200 words, 800 word and 500 words respectively for three genres). The longer articles involve longer coreference chains. Following the common practice, we used the *devtest* material only for testing. Progress during the development phase was estimated only by using cross-validation on the training set for the BNEWS section. In order to make comparisons with publications which used the same corpus, we make efforts to set up identical conditions for our experiments.

The main point of comparison is Denis and Baldridge [10], which was similar in that it described a new type of coreference resolver using simple features.

Therefore, similar to their practice, we use all forms of personal and possessive pronouns that were annotated as ACE "markables". Namely, pronouns associated with named entity types could be used in this system. In experiments, we also used *true* ACE mentions as they did. This means that pleonastics and references to eventualities or to non-ACE entities are not included in our experiments either. In all, 7263 referential pronouns in training data set and 1866 in testing data set are found in all three genres. They have results of three different systems: SCC (single candidate classifier), TCC (twin candidate classifier) and RK (ranking). Besides the three and our own system, we also report results of emPronouns, which is an

[1] See `http://projects.ldc.upenn.edu/ace/annotation/previous/` for details on the corpus.

unsupervised system based on a recently published paper [13]. We select this unsupervised system for two reasons. Firstly, emPronouns is a publicly available system with high accuracy in pronoun resolution. Secondly, it is necessary for us to demonstrate our system has strong empirical superiority over unsupervised ones. In testing, we also used the OPNLP Named Entity Recognizer to tag the test corpus.

During training, besides coreference annotation itself, part of speech, dependencies between words and named entities, gender, number and index are extracted using relative frequency estimation to train models for the coreference resolution system. Model files are created as conditional probability tables (CPTs). But HHMM model, as a time-series model may crash in running if the training models are too sparse. Namely, some variables given their conditions have no values in their CPTs since these values have never been seen in training data. However, these values may exist in testing data. These unseen values will lead zero probability mass and thus the process will crash. In order to avoid such a crash or break, smoothing is employed. That is, all possible values are assigned some probabilities. Add-one smoothing and simple good turing are used for smoothing. If the number of values of a variable is smaller than five, add-one smoothing is used. Otherwise, we use simple good turing.

Inputs for testing are the plain text and the trained model files. The entity buffer used in these experiments kept track of only the six most recent mentions. The result of this process is an annotation of the headword of every noun phrase denoting it as a mention. In addition, this system does not do anaphoricity detection, so the antecedent operation for non-anaphora pronoun *it* is set to be *none*. Finally, the system does not yet model cataphora, so we filter out 10 cataphoric pronouns from the data.

3.5.2 Results

The performance was evaluated with the success metric, the ratio of the number of correctly resolved anaphors over the number of all anaphors. All the standards are consistent with those defined in Charniak and Elsner [13].

During development, several preliminary experiments explored the effects of starting from a simple baseline and adding more features. BNEWS are employed in these development experiments. The baseline only includes part of speech tags, the index feature and and syntactic roles. Syntactic roles are extracted from the parsing results with Stanford parser.

The success rate of this baseline configuration is 48%. This low accuracy is partially due to the errors of automatic parsing. With gender and number features added, the performance jumped to 65%. This shows that number and gender agreements play an important role in pronoun anaphora resolution. For a more standard comparison to other work, subsequent tests were performed on the gold standard ACE corpus (using the model as described with named entity features instead of syntactic role features). As shown in Denis and Baldridge [10], they employ all features we use except syntactic roles. In these experiments, the system got better results as shown in Table 3.2. The result of the first one is obtained

System	BNEWS%	NPAPER%	NWIRE%
emPronouns	58.5	64.5	60.6
SCC	62.2	70.7	68.3
TCC	68.6	74.7	71.1
RK	72.9	76.4	72.4
FHMM	**74.9**	**79.4**	**74.5**

Table 3.2: Accuracy scores for emPronouns,SCC, TCC, the ranker and FHMM

by running the publicly available system emPronouns[2] . It is a high-accuracy unsupervised system which reported the best result in Charniak and Elsner [13].

The results of the other three systems are those reported by Denis and Baldridge [10]. As Table 3.2 shows, the FHMM system gets the highest average results.

emPronouns got the lowest results partially due to the reason that we only directly run the existing system with its existing model files without training. But the gap between its results and results of our system is large. Thus, we may still say that our system probably can do a better job even if we train new models files for emPronouns with ACE corpus.

With almost exactly identical settings, why does our FHMM system get the highest average results? The convincing reason is that FHMM strongly cares about the sequential dependencies. The ranking approach ranks a set of mentions using a set of features, and it also maintains the discourse model, but it is not processing sequentially. The FHMM system always maintain a set of mentions as well as a first-order dependencies between part of speech and operator. Therefore, context can be more fully taken into consideration. This is the main

[2] the available system in fact only includes the testing part. Thus, it may be unfair to compare emPronouns this way with other systems.

reason that the FHMM approach achieved better results than the ranking approach.

From the result, one point we may notice is that NPAPER usually obtains higher results than both BNEWS and NWIRE for all systems while BNEWS lower than other two genres. In last section, we mention that articles in NPAPER are longer than other genres and also have denser coreference chains while articles in BENEWS are shorter and have sparer chains. Then, it is not hard to understand why results of NPAPER are better while those of BNEWS are poorer.

In Denis and Baldridge [10], they also reported new results with a window of 10 sentences for RK model. All three genres obtained higher results than those when with shorter ones. They are 73.0, 77.6 and 75.0 for BNEWS, NPAPER and NWIRE respectively. We can see that except the one for NWIRE, the results are still poorer than our system. For NWIRE, RK model got 0.5 higher. The average of RK is 75.2 while that of FHMM system is 76.3 which is still the best.

In dataset section, we provided detailed information about the ACE corpus. The size of the corpus should be large enough to guarantee the results are statistically significant. But a hypothesis test may be still necessary to make sure if the final result can reflect the true prediction of each article. Specifically, we suppose that the corpus follows a normal distribution. Employed is the t-test where μ_0 is the final result and the sample mean $\bar{X}$ and the sample variance S^2 are the mean and the variance of the results of all articles respectively. We use S^2 to replace the actual σ^2. Given $\alpha = 0.05$, the statistical level $u_a = 1.96$. The statistical levels can be calculated with equation $U = \frac{\bar{X}-\mu_0}{\frac{\sigma_0}{\sqrt{n}}}$. The result is reported in 3.3.

	bnews	napper	nwire
U	0.739	0.066	1.58

Table 3.3: statistical levels of each genre of FHMM

This shows that the training models and the test results are valid since the statistical level of each genre is lower than 1.96.

In order to confirm the result of FHMM is better than other systems, it is necessary to make statistical comparisons. However, we don't have results of each individual article for RK, TCC and SCC. Luckily, emPronouns is a publicly available system. We have obtained results for each article. We still suppose that the results of both FHMM and emPronouns

comply with normal distributions. Our goal is to show that there are markable differences between their respective predictions, namely, between their means. Two steps are involved. First, we need to show that the variances of each genre is close and second, we need to test whether μ_{FHMM} is equal to $\mu_{emPronouns}$. What we need is that the first is true and the second is false.

The first one is calculated with the formula $F = \frac{S^2_{FHMM}}{S^2_{emPronouns}}$. F abides by a F-distribution with the degree of freedom, $n_{articles} - 1$. Given $\alpha = 0.05$, we can determine that $P(F < F_{FHMM}) = P(F > F_{emPronouns}) = 0.025$ and then for bnews, $F_{FHMM} = 0.53$ and $F_{emPronouns} = 1.88$, for npaper, $F_{FHMM} = 0.38$ and $F_{emPronouns} = 2.62$ and for nwire $F_{FHMM} = 0.47$ and $F_{emPronouns} = 2.11$.

The second one is calculated with the formula $T = \frac{\bar{X}-\bar{Y}}{\sqrt{S_1^2+S_2^2/n}}$. T is a T-distribution with $2n-2$ degree of freedom. Given $\alpha = 0.05$, we can determine that $P(|T_{FHMM}| > t_\alpha) = 0.05$ and then for bnews, $t_{\alpha(51)} = 2.021$, for npaper, $t_{\alpha(18)} = 2.045$ and for nwire, $t_{\alpha(29)} = 2.101$

	bnews	npaper	nwire
F	0.558	0.583	0.529
T	4.115	4.217	5.744

Table 3.4: hypothesis test of FHMM and emPronouns

We can see from table 3.4 that all values of F are between their ranges and all values of T are above the limits respectively. Therefore, statistical comparisons confirm that the results of FHMM and those of emPronouns are remarkably different in predictions. FHMM performs better than emPronouns in resolving pronouns of ACE corpus.

3.5.3 Error Analysis

After running the system on these documents, we checked which pronouns fail to catch their antecedents. There are a few general reasons for errors.

First, pronouns which have antecedents very far away cannot be caught. Long-distance anaphora resolution may pose a problem since the buffer size cannot be too long considering the complexity of tracking a large number of mentions through time. During development, estimation of an acceptable size was attempted using the training data (including the REUTERs corpus). It was found that a mention distance of fourteen would account for every

case found in this corpus, though most cases fall well short of that distance. Future work will explore optimizations that will allow for larger or variable buffer sizes so that longer distance anaphora can be detected.

A second source of error is simple misjudgments when more than one candidate is waiting for selection. A simple case is that the system fails to distinguish plural personal nouns and non-personal nouns if both candidates are plural. This is not a problem for singular pronouns since gender features can tell whether pronouns are personal or not. Plural nouns in English do not have such distinctions, however. Consequently, *demands* and *Israelis* have the same probability of being selected as the antecedents for *they*, all else being equal. If *demands* is closer to *they*, *demands* will be selected as the antecedent. This may lead to the wrong choice if *they* in fact refers to *Israelis*. This may require better measures of referent salience than the "least recently used" heuristic currently implemented.

Third, these results also show difficulty resolving compound noun phrases due to the simplistic representation of noun phrases in the input. Consider this sentence: *President Barack Obama and his wife Michelle Obama visited China last week. They had a meeting with President Hu in Beijing*. In this example, the pronoun *they* corefers with the noun phrase *President Barack Obama and his wife Michelle Obama*. The present model cannot represent both the larger noun phrase and its contained noun phrases. Since the noun phrase is a compound one that includes both noun phrases, the model cannot find a head word to represent it.

Finally, while the coreference feature annotations of the ACE and Reuters corpora are valuable for learning feature models, the model training may give some misleading results because of our small training corpus. We used both add-one smoothing and deleted interpolation in training models. But sparsity is still existent despite the transformation in the generation order of the observation model. In the future, we plan to incorporate REUTERs, MUC6, MUC7 and ACE into a single large training set.

3.6 Conclusion and Future Work

This chapter has presented a pronoun anaphora resolution system based on FHMMs. This generative system incrementally resolves pronoun anaphora with an entity buffer carrying

forward mention features. The system performs well and outperforms other available models. This shows that FHMMs and other time-series models may be a valuable model to resolve anaphora.

Chapter 4

Extension of FHMM CR Resolver

In last chapter, a high-performance FHMM CR resolver is illustrated. However, from the error analysis section, we can also see that defects still exist in that model. In particular, we only make use of limited features as salience features, POS tagging, gender, number and named entity. They are good features to catch most of anaphora and their antecedents. However, only these features and their combinations still miss many instances of anaphora. Actually, from linguistic perspective, different anaphoric expressions exhibit different patterns of resolution and are sensitive to different factors. The binding principle, for example is one of the well-known patterns seen in pronoun reference. In last chapter, we did make ways to implement this principle. Nonetheless, the implementation without the involvement of syntactic roles in last chapter is only a close approximation. Taking these into consideration, we extend the new FHMM CR system with these components added.

But there is a bottleneck in that model to limit us to extend the system to include more features, the slow speed of processing. As discussed in the introduction of last chapter, the system works by incrementally processing input texts word by word. This is why the system suffers from the slow speed since a system built on a HMM framework needs to do beam search within the each lattice which involves hundreds of thousands of search at each time state. The HHMM parser created by Schuler [] implements the search with A-star. Yet, even so, the large volumes of search make the processing intolerably slow and cannot be used for real-world applications.

The word-by-word fashion is based on the cognitive hypothesis that human brains may

follow the incremental working mechanism. But carefully observations on human understanding of language input reveal that this fashion may not fully correct. It is true that we human beings make use of short-term memory for holding information recently coming in and use them to help process information coming later. Yet, the information held in the short-term memory may not be really words or may be be language fragments ordered not necessarily word by word. We don't really know what information is kept there. It is more a complex combination of semantic, syntactic and even audio or video fragments. In real processing, the information in the short-term memory will be integrated into new information just coming into the brain and also the world-knowledge saved in the long-term memory in the brain. Probably, some cache in the brain will join together to help resolve the pronouns and so on.

Further, it is found that the dependencies between words may not be really so helpful in resolving anaphors, especially, if both of the neighboring words are not mentions. In this case, the operation states of the two neighboring words are *copy*. Therefore, it may not affect the resolution result if we remove these states. Based on these considerations, a revised model is formed and it is found the processing speed is dozens of faster than the original one. Thanks to the much higher speed, it is possible to build a new system which can integrate much more features without worrying much about the intractability. In this chapter, we will describe such a general resolution system.

4.1 Introduction

In this extension, we propose the general resolution system.

Corefernece features can have two dependencies with their previous states in the buffer. One is to depend on its closest neighbors and the other may depend on its potential antecedent. Besides coreference features which are directly related to the resolution of corefernce, other features may indirectly help the resolution. The neighboring parts-of-speech of the mentions are such an example. As parts of contexts of mentions, the role of mentions are determined by them to some degree. Therefore, these features are also essential and will be incorporated into the model.

After describing the general resolver system, I will show what changes we have made for the system described in last chapter. Next, I will describe the implementation of binding

principles with syntactic roles in the new system. Thirdly, we discuss how givenness hierarchy reflect the cognitive status and how these statuses can be reflected in different demonstratives and pronouns. We also discuss how the centering theory models the relationships among focus of attention and choice of referring expressions and how the centering theory can be used to resolve pronouns. These two theories are closely related while they reveal the nature of referring expressions from different angles. Both theories can potentially be employed in pronoun resolution.

4.2 A fast-speed Mention-based FHMM pronoun resolution system

As introduced in the beginning of the section, the idea of creating a fast-speed FHMM pronoun resolution system is inspired practically by the observations that many words in an article are in fact not mentions and meanwhile, mentions do not depend on these words as well. Therefore, we can remove the $COPY$ operator from the operation state set. Then, only two effective operation states are left except in the initial state and the end state, we need a dummy operator.

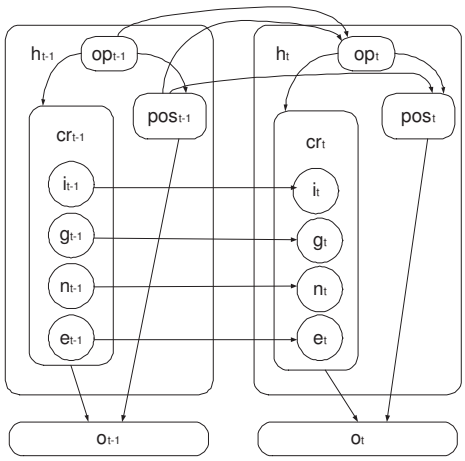

Figure 4.1: Factorial HMM CR Model

With the innovation, the difference from the model of Chapter 3.6 (we may call it word-based FHMM CR) is made on the PosModel. As we have seen, the PosModel is defined as $p(pos_t|op_t, pos_{t-1})$. And OpModel is defined as $p(op_t|op_{t-1}, pos_{t-1})$. In word-based

FHMM CR, pos_{t-1} is the pos of the immediate previous word. These two models can reflect the context dependencies between neighboring words. For example, when the previous word is $verb$ and thus its state is $COPY$, then, the probability of the next word being a NEW or OLD is high if the next word is $noun$ or $pronoun$. But these models cannot display the dependencies between two mentions since two neighboring mentions are usually blocked by a few non-mention words.

A corollary to this adjustment is that the dependencies between neighboring words no longer exist – only those between neighboring mentions are kept. However, the graphical model remains the same, except that o_t is now a mention rather than a word. The new model can be called mention-based FHMM CR system as shown in Figure 4.1. When we created word-based FHMM resolver, the pos dependency model may only consider words within the mention if the mention include more than one word. For example, in the sentence $He\ likes\ the\ US\ president$, the head word of the mention $the\ US\ president$ is $president$. Following the definition of PosModel of word-based FHMM CR, the pos of $president$ depends on its op which should be NEW and pos of its previous word which is US. Such a model seems to be not so useful. In contrast, if the pos of $president$ depends on both op and pos of words outside the mention, such as $likes$, this should be more useful.

In contrast, the fast system (let's call it mention-based FHMM CR) can reflect the dependencies in pos between two mentions. The extra words between two mentions are skipped. In this sense, it is similar to skip-chain CRF [25]. Nonetheless, it seems that the context information coming from neighboring words is missing since they are skipped. Fortunately, we can keep the context information by simply creating such a dependency model without difficulty in the new system.

We can train such a model by calculating how many times the mention $the\ US\ president$ goes with the verb (not necessarily the specific verb $likes$). In light of this, our new system include two models related to part of speech besides OpModel. Let's call them two as MentionPosModel and BigramPosModel. The probability function for MentionPosModel should be similar to the word-based FHMM resolver as $p(pos_t|op_t pos_{t-1})$. The difference lies in the fact that op_t can never be $COPY$ any more and thus both pos_t and pos_{t-1} are pos of head words of two neighboring mentions rather than those of two neighboring words. The probability function for BigramPosModel should be $p(pos_t|pos_{bigram})$ where pos_{bigram} is really the pos of neighboring words of the current mention. Following the same spirit, all features

can be trained into models of the two kinds. Thus, the new system can easily incorporate local features and long-distance features.

But if we keep two pos models, this violates the principle of Markov Blanket [26] which says that in a graph model, the Markov Blanket of a node A includes its parents, children and the other parents of all its children. That is, if we denote $MB(A)$ as the Markov blanket of the node A and all other nodes as B, then the conditional probability of $Pr(A|MB(A),B) = Pr(A|MB(A))$. But we cannot split the conditional probability into smaller one than $MB(A)$. In our present model, we should unify MentionPosModel and BigramPosModel into as one since the node considered in both models is the same one as pos_t. Namely, we should have $p(pos_t|pos_{t-1}, op_t, pos_{bigram})$.

Taking the nature of Markov Blanket into consideration, we train a new pos model. Let us call it MenBiPosModel. After the model is trained from the training data, we still need to smooth it. Since the new pos model has three dependency variables (four-dimensional CPT is created), it involves much more values than the old one. Therefore, we switch to use $MaxEnt$ to smooth the model. The advantage of $MaxEnt$ lies in that when doing smoothing, it take global relationship into consideration rather than only consider each conditional probably table alone like Add-one or Good Turing smoothing. This way, the smoothing will assign more reasonable probability mass to unseen variables.

4.3 Implementation of the Binding Principle with the Incorporation of Syntactic Roles

In order to implement binding principles completely, syntactic roles are needed. Besides, the syntactic role itself is a good feature which reflects the context information of mentions. Plus other four features, the number of hidden variables becomes five now. Like other variables, the syntax feature is supposed to be independent of all other coreference features. Based on this design, a new diagram is drawn as

With the addition of the syntax feature, the probability of $P(cr_t|cr_{t-1}, op_t)$ is updated as

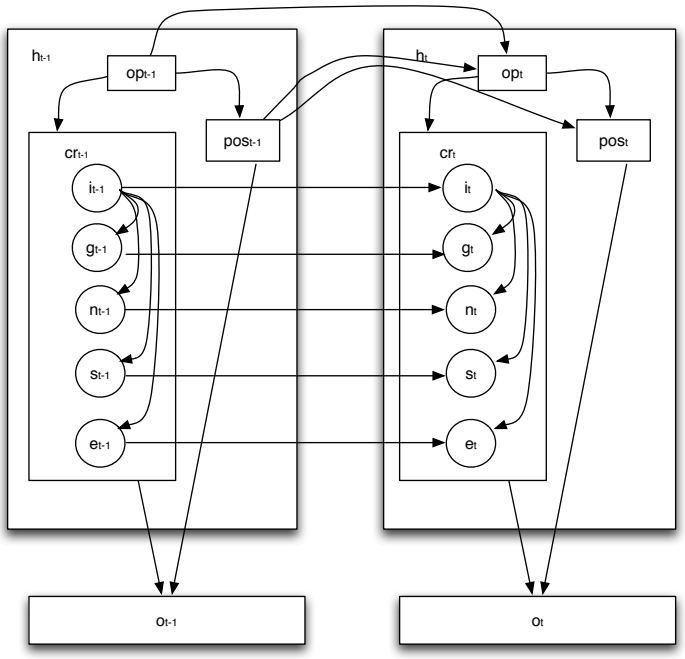

Figure 4.2: Factorial HMM CR Model 2

follows.

$$\begin{aligned} \mathsf{P}(cr_t \mid cr_{t-1}, op_t) = \mathsf{P}(i_t \mid i_{t-1}, op_t) \cdot \\ \mathsf{P}(e_t \mid e_{t-1}, i_t, op_t) \cdot \\ \mathsf{P}(g_t \mid g_{t-1}, i_t, op_t) \cdot \\ \mathsf{P}(n_t \mid n_{t-1}, i_t, op_t) \cdot \\ \mathsf{P}(s_t \mid s_{t-1}, i_t, op_t) \end{aligned} \tag{4.1}$$

Syntactic roles, in general, includes $subject, predicate, object, attributive, appositive, adverbial$ and $complement$. That is what we make use of in the present model. It should be helpful to categorize syntactic roles in more fine-grained way as follows (based on Stanford dependent parsing) though.

Yet, after a few trials, we finally give up the fine-grained categories. It is found that with the increase values of the variable, the computation complexity increase exponentially so that the whole model becomes intractable.

Now, let us illustrate how binding principle can be implemented with the new design. In

syntactic roles	advmod amod appos ccomp conj_and conj_but conj_or csubj dep dobj iobj npadvmod nsubj nsubjpass pobj poss prep_about prep_against prep_along_with prep_as prep_at prep_behind prep_beyond prep_by prep_for prep_from prep_in prep_in_front_of prep_out_of prep_over prep_through prep_to prep_with rcmod xcomp prep_of prep_on prep_between

Table 4.1: Syntactic Roles

Chapter 3.6, when OP is OLD, we approximate binding principle with $P(pos_t|loc_t, pos_{loc_t})$ where loc_t represents which antecedent that the current pronoun refer back to. We don't know what syntactic role of pos_{loc_t}. This method itself complies with statistical principle. It is effective when syntactic roles are not available. But if syntactic roles are known, we can now employ s_{loc_t} rather than pos_{loc_t}. This should work better than the pos approximation. But syntax features and POS tags are not independent of each other. The binding feature is defined as $P(pos_t|loc_t, s_{loc_t})$. For example, if $pos_t = reflexive$, $P(pos_t|loc_t, s_{loc_t})$ where loc_t has larger values (implying closer mentions to pos_t). As before, loc_t is in fact one value of i_t which is the index feature vector. In addition, $s_{loc_t} = subject$ should have higher values since reflexive pronouns always refer back to subjects within its governing domains. This change brings the model subtle changes as well, shown in figure 4.3.

That is, rule-based principles are integrated into the statistic process. Since rule-based is always definite, this integration should make the prediction more definite as well. However, since ACE corpus doesn't have syntax annotations, instead, we use Stanford parser to do this job, the result will be affected by the accuracy of the parser. From the result we have got, it seems that the parser enjoy high accuracy and thus this integration really boost the performances.

4.4 Implementation of Givenness Hierarchy in Hidden Model

The Givenness hierarchy itself doesnÕt aim at resolving coreference between expressions. Instead, it tries to determine the appropriateness and possible interpretation of a referring expression depending on the cognitive status of the referent of that expression, as determined by its antecedent. The theory aims to capture the mechanisms or strategies that the

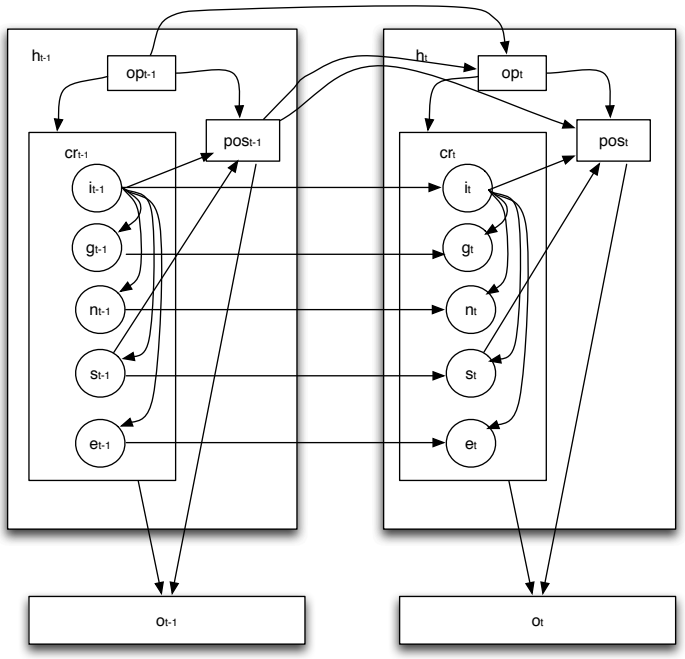

Figure 4.3: Factorial HMM CR Model 3

hearer makes use of in restricting the set of possible antecedents , and thus determining cognitive status for referents should play an important roles in resolving coreference in human communication. The principles that Gundel et al [27] propose to link referents of pronouns to their antecedents can be implemented in a coreference resolution system. In my work, I integrate these into the FHMM CR system and achieve good results. Since my system, for now, focuses on resolving pronouns, IN-FOCUS, which is the highest status on the hierarchy, and the one that is proposed to be coded by unstressed personal pronouns in English, is the relevant status here. According to the coding protocol for cognitive status, Þve conditions are sufficient for bringing an entity into focus of attention.

1. It is introduced in a syntactically prominent position in the immediate preceding sentence, such as subject, syntactic topic, focus of cleft or existential sentence, etc., where syntactic topics include topicalized or dislocated phrases and topic marked phrases.

2. It is part of the interpretation of a previous part of the same sentence.

3. It is a higher level topic that is part of the interpretation of the preceding sentence

(whether it is overtly mentioned there or not).

4. It is part of the interpretation of each of the two immediately preceding clauses.

5. It is the event denoted by the immediately preceding sentence.

For the first condition, we need to distinguish whether the mention is in the previous sentence or clause or in the current sentence or clause if its syntactic role is subject. If the mention is in the current sentence or current clause, the pronoun cannot refer to it while the reflexive must refer to it according to the binding principle.

Based on the above principle, we can train a focus model from the training data by the statistics of the relationship between the coreferring mentions. Formally, the trained model looks like the following,

$$P(focus_t|pos_t, syn_{loc_t}). \tag{4.2}$$

where $focus_t$ is a binary variable. If current mention is $IN-FOCUS$, $focus_t = 1$. Otherwise, $focus_t = 0$. pos_t is the part of speech of the current mention and syn_{loc_t} refers to the syntactic roles of the antecedent.

We only care about pronouns. Therefore, pos_t is always $pronoun$. But empirically, pronouns of different persons or genders or animacy have different distributions and different referring nature as well. Further, it is found that in the I2B2 corpus, second person pronouns refer to the patient in most cases. First person pronouns are ambiguous and refer to either the patient or the care-providing doctor. Hence we divide pronouns (as represented in the pos model) into finer categories such as non-personal pronouns or 1st-/2nd-/3rd-person pronouns. Training the pronoun resolution model at this granularity yields intuitive empirical information that the system may make use of.

4.5 Centering with Optimality Theory in FHMM

From discussion of related work, we know that if we know sentence boundaries, we know how many mentions in a sentence and we know the parts-of-speech of those mentions, we may employ centering to determine which pronoun refers to which antecedent in previous sentence. All of the knowledge can be obtained from the hidden buffer where features about

previous mentions are stored. Yet, it may not be so straightforward to assign forward and backward centers in hidden buffer. Mentions, in our system are employed as input from observation states. Hence, we hold that it is better to implement centering theory in the observation models.

Further, the hidden state may be so involved that the computation may be hard to tract. In addition, we integrate givenness hierarchy inside the hidden state. Givenness hierarchy and centering may compensate each other. So, givenness hierarchy play some roles in filtering out some bad candidates in the hidden states. Then, centering may continue to filter out some more ineligible ones in the observation states.

Similar to the implementation of givenness hierarchy, we need to know the sentence boundary. In observation state, this is not hard to obtain. Since my work is on text processing, we can simply check where the period is (the work can be used in speech as well. But I didn't extend it to speech yet). One similar thing to the integration of givenness hierarchy should be done in the implementation. Namely, we need some data structure to store current sentence and the previous sentence. As is known, the centering theory supposes that we know sentence boundary and also, we need two complete sentences. This is a big constraint to prevent it from the use of it in the FHMM pronouns resolution system since the system processes documents word by word. Beaver [28] proposed an alternative of centering theory, which convert BFP to COT. Although COT in essence still works within the framework of sentence processing, we think that we can use it in a word-by-word fashion. In the following part, I illustrate how to process pronoun resolution with the new theory.

4.5.1 Basics of Centering Optimality Theory

Two levels of representation are employed in optimality theory. They are an input and an output. The input is a representation of logic form (LF) or argument structure, and the output is a representation of surface structure. Relative to a given fixed input, the constraints are used to find the optimal output [29]. According to COT [28], COT can be generalized as bellows.

In COT, the input represents the surface form and the output represents the meaning. The surface form is a partially syntactically analyzed sentence, and the meaning is a mapping from referring noun phrases (NPs) in the sentence to their referents. Given some inputs to an OT model, a series of constraints established offer a way to select an optimal interpretation

from a set of candidates. These constraints can be boolean with respect to candidates and some are not. It is possible that while two candidates both violate the same constraint while one candidate involves more violations. Candidate A is superior to candidate B if and only if there is some constraint x such that (i) A has no more violations then B of each constraint higher ranked than x, and (ii) A has strictly fewer violations of x than B.

Beaver [28] renames $backward-lookingcenter$ to $topic$ and redefined it as

Definition 1 *The topic of a sentence is the entity referred to in both the current and the previous sentence, such that the relevant referring expression in the previous sentence was minimally oblique. If there is no such entity, the topic is undefined.*

Six constraints are developed roughly corresponding to transition rules and filter rules in BFP. The operation corresponding to constraints takes place following the order in the following list.

- AGREE Anaphoric expressions agree with their antecedents in terms of number and gender.
- DISJOINT Co-arguments of a predicate are disjoint.
- PRO-TOP The topic is pronominalized.
- FAM-DEF Each definite NP is familiar. This means both that the referent is familiar, and that no new information about the referent is provided by the definite.
- COHERE The topic of the current sentence is the topic of the previous one.
- ALIGN The topic is in subject position.

$AGREE$ and $DISJOINT$ are the top two constraints. They reflect the syntactic requirements. The second one is in fact the application of Principle B from binding theory. In BFP, these constraints are made use of in the construction and filtering stage, respectively. $PRO-TOP$ is the mirror of Centering's Rule 1 which says that "$if\ there\ are\ pronouns\ in\ the\ sentence,\ then\ one\ of\ them\ refers\ to\ the\ backward-looking\ center\ of\ the\ current\ sentence$". But different from Centering's Rule 1, there is not if clause in $PRO-TOP$. Therefore, the constraint in COT is defeasible. Namely, if there is a pronoun, COT functions identically with rule 1

of BFP. Otherwise, $PRO-TOP$ is irrelevant. This way avoids absoluteness. Actually, this is a typical feature of OT which constraints can always be cancelled if not applicable, thus adding its flexibility.

$FAM-DEF$ defined that the definites should be familiar, including pronouns, definite descriptions and proper names. This is not defined in BFP and derived from the classic formalization of the notion of familiarity using file-change semantics [30].

Although constraints in OP are always violated and thus it seems that there are not absolute constraints, the combinations of constraints can have the effect of producing absolute ones. The interaction of $PRO-TOP$ and the lower ranked $FAM-DEF$ has such an effect that makes the rule 1 of BFP indefeasible. The example given in [28] is a convincing one. In a sentence, a possible interpretation is given to some proper name or definite description and anaphoric pronouns. The proper name is supposed to refer to the topic and yet some anaphoric pronouns in the current sentence refer to discourse entities rather than the topic. The combination of $PRO-TOP$ and $FAM-DEF$ will lean against such an interpretation because this reading breaks $PRO-TOP$ and not the lower ranked FAM-DEF. Consequently, this interpretation is not an optimal one. There must be alternative interpretations which violates $FAM-DEF$ by allowing the proper name to refer to a novel entity and meanwhile the topic can be identified with the referent of some pronoun so that $PRO-TOP$ will be abided by. This interpretation can be regarded to more optimal one.

$COHERE$ and $ALIGN$ are the two constraints corresponding to the transition types in BFP. The condition for $COHERE$ is that topic is defined and unchanged.

In order to have a clearer picture on how COT works, I creates an example to illustrate it. The example I use is still the once used in Chapter 2. The example is cited again in the following.

1. (a) $Terry_i$ really goofs sometimes.
   (b) Yesterday was a beautiful day and he_j was excited about trying out his_k new sailboat.
   (c) He_l wanted $Tony_m$ to join him_n on a sailing expedition.
   (d) He_o called him_p at 6 AM.
   (e) He_q was sick and furious at being woken up so early.

Staring from the second sentence, we can build tableaus to show how optimal candidates are selected out.

Example 2	AGREE	DISJOINT	PRO-TOP	FAM-DEF	COHERE	ALIGN
☞ j=i k=i					*	
j=i k!=i k!=j			*	*	*	
j!=i k!=i k!=j			*	*	*	*
j!=i k=i				*	*	*
j!=i k=j				*	*	*

Table 4.2: the ranking tableau for 1b

In Example (1b), there are two pronouns, he_j and his_k which both refer to the same person, $terry$ in (1a). Probably, a detailed interpretation of the table may make clear which option is the optimal one and how COT works.

Two pronouns and one antecedent comprise 5 different possible results. The first one is the two pronouns and the antecedent all refer to the same person $TERRY$; the second one, he_j refers to $terry_i$ while his_k doesn't refer to $terry_i$; the third one, he_j doesn't refer to $terry_i$ while his_k doesn't refers to either $terry_i$ or he_j; the fourth choice is that he_j doesn't refer to $terry_i$ and his_k refers to $terry_i$; the last one is he_j doesn't refer to $terry_i$ and his_k refers to he_j.

Now, let us have a look at how each constraint is abided by or violated.

1. AGREE: Each of these candidates doesn't violate agreement since no gender or number conflicts happens.

2. DISJOINT: No candidates violates this constraint since the predict "was excited" doesn't involve the second pronoun analyzed.

3. PRO-TOP: The constraint is violated in the second candidate and the fifth since either one of the pronouns or the two pronouns don't involve anaphoric references. And thus no topic is defined.

4. FAM-DEF: all of candidates except the first one violate this constraint since at least one of the two pronouns is not interpreted as anaphoric.

5. COHERE: All of candidates violate this constraint since the topic of the previous sentence was not defined.

6. ALIGN: the lower three cases violates this one because the topic in current sentence is not the subject among the three candidates.

Evidently, the first candidate has fewer violations than others. Therefore, it is the optimal one and in the tableau, it is dignified with a "☞".

For the example (**??**c), three mentions are involved. Yet, we need to consider two of them since only He_l and him_n are pronouns. Further, since according to above derivation, both he_j and his_k refer to $Terry_i$, we can use index i to refer to previous mentions. Nonetheless, $Tony_m$ should be considered as a possible antecedent of him_n so as to take intra-sentential coreference into considerations. For intra-sentential coreference, COT [28] defines a saliency principle to take care of it. In fact, BFP, though in the beginning, it only took inter-sentential coreference into consideration, also started to extended to handle intra-sentential one in Kameyama [31]. According to COT, RECENCY is one element to bring about saliency, which says that one discourse entity is more salient than another if the first was referred to in a later clause. This rule capture partially the idea that salience declines over time and open up the possibility of a treatment of intra-sentential anaphora. Based on the above discussion, the ranking tableau is constructed as Table 4.3.

Example 2	AGREE	DISJOINT	PRO-TOP	FAM-DEF	COHERE	ALIGN
☞ l=i n=i						
l=i n=m		*				
l=i n!=i n!=m				*		
l!=i n=i				*	*	*
l!=i n!=i n=m		*	*	**	*	*
l!=i n!=i n!=m			*	**	*	*

Table 4.3: the ranking tableau for 1c

In Table 4.3, all of them do not violate $AGREE$ since gender and number are all $male$ and $singular$. The first candidate doesn't violate any of the constraints. Thus, it is the optimal candidate. The second candidate violates $DISJOINT$ due to the choice to let $n = m$. They two are in the same governing domain and the choice, $n = m$ violates binding principle or $DISJOINT$ constraint in COT. The third candidate violates only $FAM-DEF$ since n is not pronominized. The fourth candidate violates $FAM-DEF$ since n is not interpreted as anaphoric. The fifth candidate violates all constraints except $AGREE$ and the last candidate violates all constraints except the first two. The fifth supposes that $n = m$, violating $DISJOINT$. Both the fifth and the last don't define the topic because there is no anaphoric reference at all. Consequently, all constraints that make reference to the topic, including $PRO-TOP$, $COHERE$ and $ALIGN$ are violated simply because there is no topic. Finally, both candidates violate $FAM-DEF$ twice since both n and m are not interpreted as anaphoric.

Example 2	AGREE	DISJOINT	PRO-TOP	FAM-DEF	COHERE	ALIGN
☞ o=i p=m						
o=p=i		*				
o=m p=m					*	*
o=p=m		*			*	
o=i p not ∈ {i, m}				*		
o=m p not ∈ {i, m}				*	*	
o not ∈ {i, m} p=i				*		*
o not ∈ {i, m} p=m				*	*	*
o,p not ∈ {i, m} o!=p			*	**	*	*
o=p not ∈ {i, m}		*	**	*	*	*

Table 4.4: the ranking tableau for 1d

Similar to Table 4.3, $AGREE$ is not violated by all candidates. As regards to $DISJOINT$, as long as the two pronouns, both arguments of the predicate "called", are resolved to the same mention, violations occur then. The lowest two cases do not define topics. Hence, $PRO-TOP$ is violated. Lowest six candidates violate $FAM-DEF$. In all of them, at least one of the two pronouns is not interpreted as anaphoric. When o is not interpreted as i,

that implies that the current topic is not identical to the previous topic. Likewise, when the current topic is not in the subject, $ALIGN$ is violated. After counting all violations, we find that the first one is the optimal one.

Example 2	AGREE	DISJOINT	PRO-TOP	FAM-DEF	COHERE	ALIGN
q=i						
☞ q=m					*	
q not ∈ {i,m}			*	*		*

Table 4.5: the ranking tableau for 1e

The last only involves one pronoun. The tableau only include three lines, namely, $q = i$, $q = m$ and neither. Then, we can see the result is the same as that obtained with BFP.

4.5.2 implementation of COT in FHMM

As is seen in Chapter 3.6, FHMM, like common HMM, involves both hidden states and observation states. In our work, observation states are words and hidden states include three main components, OP, POS and CR.

Hidden states take care of transition probabilities between time steps. For coreference resolution, the goal is to find antecedents for anaphors. Therefore, in last section, dependencies between anaphors and antecedents are constructed as transition models in the FHMM. COT, as a theory of resolving coreferences by means of constructing the relations between mentions of current sentences and those of previous sentences, is naturally integrated into hidden states. In our work, we only consider pronoun resolution. But we distinguish common pronouns and reflexives.

For COT, like the givenness hierarchy, the key thing here is to convert the relations to conditional probability tables so that we can build a COT model in the system. In previous work, Donna Byron [32] implements centering as tableau ranking with optimality theory. But she doesn't convert COT to a statistical model. Yet, it is found that there are corresponding probability models to each constraint.

There are six constraints among COT. Nonetheless, we do not need to implement all of them as probability models since three of them are existent in our probability models.

Agree constraint can in fact be split into both mNumModel and mGenModel. Disjoint is the binding model implemented in our system as $p(pos_t|loc, syn_{loc_t})$. $FAM-DEF$ is similar to the OP model in our system. Therefore, the OP model can be used as an approximation of $FAM-DEF$.

The constraints we need to implement include Pro-Top, Cohere and Align. Pro-Top is the dependency between topics and part-of-speech and thus can be expressed as $p(topic_t|pos_t)$. Cohere is the dependency between the current topic and the topic which is coreferring with the current topic. We can express it as $p(topic_t|loc_t, topic_{loc_t})$ where $topic_t$ is the topic at current state, loc_t is the index of the mention which current mention is coreferring and $topic_{loc_t}$ is the topic value of the coreferring mention.

Align can be expressed as $p(topic_t|syn_t)$. It refers to the probability that the current mention is a topic given its syntactic role.

So, this way, one thing we only need to do is to determine what topic is. According to the definition, $topic$ is the entity which are referred in both current sentence and previous sentence. If there are two or more, the most outstanding one should be the $topic$.

Two observations here make us further improve the topic model. Firstly, $topic$ in COT is in fact the same concept as that in givenness hierarchy. Secondly, the principle of Markov Blanket requires that three probability models about topics and the model of giveness hierarchy should be merged into one. Accordingly, the whole topic model can be generalized as follows,

$$P(topic_t|pos_t, loc_t, syn_t, syn_{loc_t}, topic_{loc_t}) \tag{4.3}$$

That is, we can merge all topic models into a six-dimensional conditional probability table. Following this spirit, the graphic model will change from Figure 4.3 to Figure 4.5.

Again, in order to overcome the sparsity, we make use of $MaxEnt$ to train the topic model.

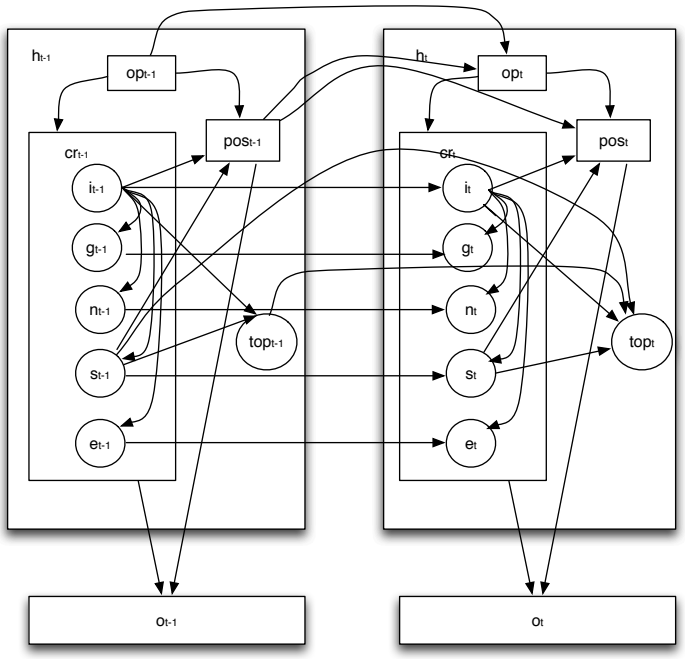

Figure 4.4: Factorial HMM CR Model with COT

4.6 Evaluations

After we made the extensions, a series of evaluations are made with the new model. The features used in the system of last chapter are used as baselines. Syntax, bigram-pos and the implementation of the binding principle together form the second stage of experiment. Finally, the implementation of givenness hierarchy (GH) and COT for the third stage. In order to test the robustness of our system, we train the model files on both ACE corpus and medical corpus and test mainly on clinical corpus called I2B2 2011 coreference corpus for the moment. Hence, results reported in this thesis are all from I2B2 2011 coreference corpus.

4.6.1 Measure Metrics

In the challenge, system performance was measured using MUC [33], B-CUBED (B3) [34] Entity-based CEAF [35], and BLANC [36], similar to SEMEVAL-2010 [36] and CoNLL-2011 [37].

MUC metric is based on links between mentions and, thus, it does not concern mentions with no link and does not take into account the length of chain segments, or numbers of

mentions, connected by a link.

B-CUBED metric is based on mentions, and mitigates the issues of MUC. Similar to MUC, however, gold-standard chains and predicted chains can be mapped each other one-to-many or many-to-one, lacking an entity-level view. CEAF metric is based on entities, where gold-standard and predicted chains are mapped one-to-one. In particular, entity-based CEAF normalizes a similarity score of each pair of mapped chains. BLANC is a recent metric that measures the similarity between gold-standard chains and predicted chains based on pairs of mentions in them. For official evaluation, an average of B-CUBED, CEAF, and MUC (mention, entity, and link average (MELA)) [38] was used, without including BLANC. Scores of the three metrics were averaged with equal weights, in the same manner as CoNLL 2011. The reported performance measures were calculated using the Python script provided by the challenge organizers.

4.6.2 I2B2 2011 coreference corpus

I2B2 2011 coreference corpus is a collection of a few clinical data which consists of three sets from three different institutions; Partners HealthCare, Beth Israel Deaconess Medical Center, and University of Pittsburgh. The data of University of Pittsburgh have two types of notes, i.e., discharge and progress notes. All Protected Health Information (PHI) is fully de-identified. In the training set, gold standard markables and chains are manually annotated. The training set contains total 492 notes (Partners: 136, Beth: 115, Pittsburgh: 119 discharge and 122 progress notes) and the test set contains total 322 notes (Partners: 94, Beth: 79, Pittsburgh: 77 discharge and 72 progress notes)

The data itself is used for I2B2 2011 challenge on coreference resolution.

The above evaluation metrics is for the general coreference resolution rather than for pronoun resolution. Namely, the results reported with the above metrics include nominal resolution. In order to see how good pronominal resolution is, I still report success rate as in last chapter. Table 4.6 reports the success rate of the pronoun resolution and Table 4.7 reports the standard rates including precision, recall and F-measure with four measures.

From the success rate, we can see the baseline is around 59.75%. This is much lower compared with the success rate given in Chapter 3.6. But this is understandable since the baseline is done after the $copy$ state is removed and thus, the context information is lost. In order to see how poorer the result is without $copy$ operator, I reran the old model on I2B2 corpus

System	Beth%	Partners %	Discharge %	Progress %	average %
copyIncluded	74.3	70.5	69.3	64.9	69.75
copyRemoved	62	60.5	57.5	59	59.75
syntax, pos-bigram plus binding	68	66.5	64	65.5	66
cot plus GH	74	73.5	70	72.5	72.5

Table 4.6: success rate of pronoun resolution

System	Average %	B^3 P|R|F %	MUC P|R|F %	BLANC P|R|F %	CEAF P|R|F %
copyIncluded	77.7	89.4|90.4|89.9	54.7|76.8|63.9	94.7|71.2|78.8	79.4|94.7|71.2
copyRemoved	75.7	88.2|90.5|89.3	51.6|68.3|58.8	89.1|65.9|72.7	73.8|84.6|78.9
syntax	80.2	88.5|92.7|90.6	68.3|71.5|69.9	90.4|69.6|76.4	78.9|81.3|80.1
cot	82.1	88.8|93.5|91.1	78.3|72.1|75.1	90.7|98.3|94.2	82.6|77.8|80.2

Table 4.7: metrics with nominal resolution results added

as well. That is the results of the second row. We call the system now as $copyIncluded$. It is found that the loss without $copy$ is large, as shown in Table 4.6. Taking the speed into considerations and also taking the potentiality of feature addition, it may still be worthwhile. Indeed, we have attempted to add more feature variables to old models. It turned out that the huge search space leads the computation intractable. For the simplified model, instead, we can add more though the speed is going slower and slower.

In addition, it is found that many errors come from the type mismatching between named entity types and pronouns. One of the reasons that leads to the mismatching is that due to the distance limit (6 is still used as the maximum distance), some pronouns cannot find their antecedents. In this case, as discussed in Chapter 3.6, in order to avoid the crash of the whole process, smoothing is done to assign small probabilities to unseen variables. When encountering this case, the smoothing works so that pronouns may refer to antecedents with different types. Typical examples include personal pronouns which may refer to medical terms. In addition, the domain difference is also one of the reasons that lead to the lower performance. If we look at the result including nominal, we can see that the precisions for all metrics are quite high while the recalls are not so good.

In coreference metrics, we know that the precision refers to the correct number of links or mentions from the key compared to the system output while the recall refers to the correct number of links or mentions from the system output compared to the key. The fact that recalls are not good indicate that for each key chain, links or mentions are misjudged into other key chain. In contrast, low precision indicates that for each system chain, not many true mentions or links are on it. The mismatching types wrongly assign mentions to wrong chains. That leads the lower correct numbers of mentions or links in each true chain. But the precision is still good. Then, it means that the system may link mentions which don't belong to the same chain together. Therefore, the precision is not affected much. A simple example can illustrate this more clearly.

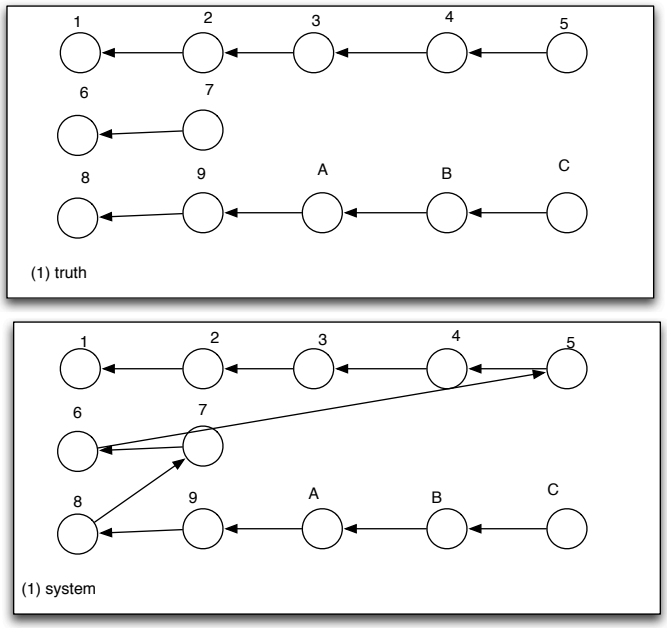

Figure 4.5: the key vs the system output

Let's suppose that in the data, there are three chains as shown in the above diagram. But the system output connect three chains sequentially as one chain. If we count mentions, there are 12 mentions while if count links, there are 11. According to MUC, the recall is 100% since there are 9 correct links from system. Then, $recall = 9/9 = 100\%$. But the precision is only 81.5% since there are 9 correct links from the truth out of 11 in the system. Namely,

$precision = 9/12 = 81.5\%$

In order to overcome this mismatching, I release one constraint on pronouns: allow pronouns be assigned as NEW. The probability mass is still trained from the corpus. In 3.6, we removed cataphora cases from the training and thus guarantee that pronouns which appear in the beginning of the text is ignored. Now, since we allow pronouns are assigned as new, we don't need to ignore the cataphoric. On the contrary, cataphora can be cared about now. That is, the pronoun which refers forward to other mentions will be assigned as new. Then, it is waiting for another pronoun referring back to it.

Along with the release of the constraint, we did the second experiment in which syntax, pos-bigram and binding are implemented. These brought above 6.25% increase in the average success rate. This implies that added features reduce the error to a large degree. For the complete results, the improvements are even larger. More than 10% improvement is seen for the average. The different between success rate and the chain metrics can be understood as that, the reduction of mismatching make the wrong link broken and thus, precision increases much.

For the last experiments, we added COT and givenness hierarchy. We see another 5.5% increase for the success rate. COT and givenness hierarchy have been successful in rule-based coreference resolution system. This is the first time to use them in the probability models. This improvement confirms the validity of these linguistics theories from the perspective of the statistics and meanwhile it shows that the combination of statistical models and linguistics theories is a feasible method in NLP research. If we turn to the chain metrics, we only find only 2% increase. This may indicate that the result of these chains may reach an extremity so that it may be hard to get better results. In fact, for coreference resolution, the final results of the chain metrics have reached the state-of-art though these results are still not high enough compared with POS tagging and parsing. How to make the results even better is our future work.

Chapter 5

Brief Literature Review on Relation Detection

Various approaches are taken to tackle this problem. Based on machine learning methods employed, we can still classify them into supervised and unsupervised discovery approaches.

5.1 Supervised Methods

5.1.1 Pipelined methods

Miller, S. and H. Fox, et al. (2000) started to employ semantic parse tree. Their system makes all relation, entity and syntax decision at once using a generative probability model. Pipelined method was proposed by McDonald, R. 2005. The primary motivation is the observation that most relation extraction systems are pipelined. Usually entity tagging and other syntactic tagging such as part-of-speech and parse tree generation are done beforehand using separately trained models. The output of these models are then used as input to the relation extraction model. The relation extraction model is based on a first-order Markov assumption.

But with these methods, creating training data is non-trivial. Meanwhile, Markov assumption make the model unable to incorporate long-range features into relation decisions.

Generative models cannot easily represent a rich set of dependent features in a computationally tractable manner.

5.1.2 Joint Detection of Entities and Relations

Roth, D. and W. Yih (2002) proposed a probabilistic frame work for recognizing entities and relations together. Classifiers are first trained separately for entities and relations, and then their output (conditional probabilities for each entity and relation) is used together with constraints induced between relations and entities, such as selectional restrictions of verbs established in terms of types of entities, in order to make global inferences for the most probability assignment for all entities and relations under consideration.

However, properties like semantic types of phrases (i.e., class labels, such as people, locations and relations among them are more difficult to acquire. Furthermore, the integration of named entity recognition and relation detection is ideally a good model. In fact, since named entity recognition has reached a high level up to now. The result of named entity recognition can be done independently and then used as input for relation detection. Namely, we can focus on relation detection rather than care them both.

5.1.3 Distant Supervision Methods

One interesting supervised model which worth mentioning is the one presented by Mintz et al, 2009. Essential assumptions of their system is that if two entities participate in a relation, any sentence that contain those two entities might express that relation. Thanks to this assumption, the system has the ability to combine information from many different mentions of the same relation. They used Freebase to get data as 1.8 million instances of 102 relations connecting 940,000 entities;

5.1.4 Kernel methods

All previous methods need extraction of language features. Types of NEs are used as features. Besides, the sequence of words between the two entities; the pos tags of these words, a flag indicating which entity came first in the sentence; a window of k words to the left of entity 1 and their pos tags; and a window of k words to the right of entity 2 and their pos tags are also

used as features. Syntactical features include a dependent path between the entities and for each entity, one 'window' node that is not part of the dependency path.

But besides above features, new features are hard to discover. Then, kernel methods can be a good choice. The idea of kernel methods is to compute the similarity of two objects directly. The key problem is how to represent and capture structured information in complex structures.

NEs involving relation types are usually close to each other and within one sentence. So, strong correlation should exist between two NEs, texts and syntactical components between them. Tree kernel methods were employed in Zhao (2005) and Zhang (2008). The tree refers to the parse tree of a sentence. Specifically speaking, there are four kinds of trees: compressed path-enclosed tree; bottom-attached tree; entity-attached tree and top-attached tree. Kernels are defined as the number of common sub-trees as the syntactic structure similarity between two parse trees. T_1 and T_2.

$$K_C(T_1, T_2) = \sum_{n_1 \in N_1 n_2 \in N_2} \bigwedge(n_1, n_2) \tag{5.1}$$

Then, SVMs are used to train classifiers and testing it on testing data.

5.2 Unsupervised Methods

Unsupervised models strive for shaking off the dependencies of annotated corpus and for improving the ability of processing large volume data.

5.2.1 Clustering relation types with similar semantic and syntactic dependencies

Gamallo, P. and M. Gonzalez, et al. (2002) employ an unsupervised strategy for clustering semantically similar syntactic dependencies, according to their selectional restrictions. A set of interpretation rules are then applied to classify the syntactic dependencies in order to extract semantic relations. The Semantic relations are organized according to a hierarchical structure similar to the one used in WordNet.

5.2.2 Bootstrapping

Bootstrapping is also widely used in relation extraction in recent years. It uses a weak learner to learn a model based on a small set of labeled data. Then, the weak leaner is used on a large set of unlabeled examples. Iteratively, using the output of the learner as training data for the next iteration.

Co-training and Yarowksky are the two main types of bootstrapping algorithm.

- Co-training algorithm

  Co-training uses two or more learners, each with a separate view of the unlabeled data. The output of one is then used as the input for others during the next iteration o training.

- Yarowsky algorithm

  Yarowsky algorithm uses just one trainer, taking the highest confidence examples on each iteration as training for the next iteration.

  Selectivity is a key in this algorithm. It refers to the confidence in the classifiers ability to generate precise training examples for future iterations. Coverage is the system ability to generate new (or all) labeled examples. If the classifier routinely generates false positives, then its accuracy will decrease every iteration, until it becomes of no use whatsoever. A classifier that is overly selective will not introduce any new examples and the system will terminate without significantly expanding its seed set.

The main problem with Yarowsky style bootstrapping algorithms according to Yangarber is that the patterns that the system extracts degrade with every iteration since ultimately some errors will be introduced to the system. With co-training or counter-training algorithms, the multiple classifiers play a role in preventing this from happening by constraining each other through the different views of the data. Some of them relied on partial or full syntactic analysis.

Chapter 6

Initial Experiments on relation detection with LDA models

[39] constructed a system which uses a NER system to identify pairs of entities and then cluster them based on the types of the entities and the words appearing between the entities. Inspired by this approach, we decided to take the same one to acquire input data. In contrast with them, we employed two learning models: latent Dirichlet Allocation and SVM.

6.1 LDA

As described previously, we regard texts between two NEs as an instance of document. LDA, as a model which is good at topic categorization seems to be natural choice.

In LDA, each document is regarded to be mixture of topics though each document dominantly belongs to one topic. It defines three basic concepts as word, document and corpus. A word is the basic unit of discrete data, from vocabulary indexed by $1, ..., V$. The v-th word is represented by a V-vector w such that $w_v = 1$ and $w_u = 0$ for $u \neq v$. A document is a sequence of N words denoted by $d = w_1, w_2, ..., w_N$. A corpus is a collection of M documents denoted by $D = d_1, d_2, ..., d_M$.

According to [40], LDA is a graphical model displayed as Figure 6.1, where the shaded circle w in the plate inside is the individual word. That plate is the individual document and N inside that plate is the number of words. The outside plate is the corpus which is composed of documents with the number of D. Five non-shaded circles there represent different

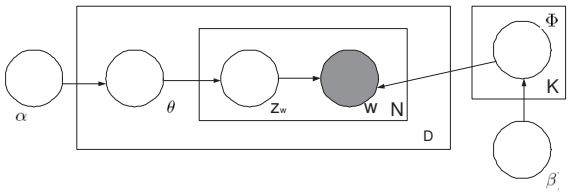

Figure 6.1: Graphical model of LDA

parameters. α and β are two hyperparameters to launch the sampling. θ and ϕ depend on them respectively. The conditional distribution θ given α is chosen as Dirichlet. It is a convenient design since Dirichlet is in the exponential family, has finite dimensional sufficient statistics, and is conjugate to the multinomial distribution. As the figure shows, θ generates z; z and ϕ jointly generate w. They are both multinomial distribution. Among them, θ represents the topic distribution of each document and ϕ and z represent the topic distribution of each word. So, together, given the parameters α and β, the joint distribution of a topic mixture θ, a set of N topics z, and a set of N words w is given by:

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = p(\theta|\alpha) \prod_{n=1}^{N} p(z_n|\theta) p(w_n|z_n, \beta) \tag{6.1}$$

Integrating over θ and summing over z, we obtain the marginal distribution of a document:

$$p(w|\alpha, \beta) = \int p(\theta|\alpha) (\prod_{n=1}^{N}) \sum_{z_n|theta} p(w_n|z_n, \beta) d\theta \tag{6.2}$$

6.2 Drawback of LDA in Predicting Relation Types

But after we tried LDA, we found that LDA often learns some topics that are hard to interpret and those topics cannot find exact correspondence to relation types of ACE corpus.

Table **??** is a sample output of theta values after we ran LDA on ACE corpus. Seven unknown topics are involved. The row numbers are instance numbers which represent numbers of each document. As a mixture topic model, LDA assigns probabilities of topics to each instance. Therefore, each instance has seven probabilities. The topic with the largest probability can be regarded to represent the dominant topic in that instance. Presumably, the topic is supposed to correspond to the relation type of that instance.

θ value	1	2	3	4	5	6	7
1	0.052	0.052	0.263	0.052	0.368	0.157	0.052
2	0.047	0.142	0.142	0.047	0.523	0.047	0.047
3	0.047	0.047	0.047	0.047	0.047	0.714	0.047
4	0.043	0.043	0.043	0.130	0.043	0.043	0.652
5	0.052	0.052	0.052	0.157	0.578	0.052	0.052
6	0.047	0.047	0.142	0.142	0.523	0.047	0.047
7	0.428	0.428	0.028	0.028	0.028	0.028	0.028
8	0.058	0.058	0.176	0.058	0.529	0.058	0.058
9	0.263	0.368	0.052	0.157	0.052	0.052	0.052
10	0.043	0.043	0.043	0.043	0.043	0.043	0.739
11	0.04	0.04	0.12	0.04	0.36	0.04	0.36
...	...						

Table 6.1: A sample LDA θ assignments for ACE relation data.

	Prec%	Rec%	F%
SVM	53.2	35.2	40.3
LLDA	28.3	51.6	36.6
MEDLDA	**57.8**	**53.2**	**55.4**

Table 6.2: Overall performance of the 3 systems

But the actual situation is that the dominant topic learned this way does not correspond to relation types.[1]

What we want is to cluster named entity (NE) pair which has similar relation types together. This is similar to the task of predicting a movie rating from the words in its review. Intuitively, good predicative topics will differentiate key words for each relation type and also be able to differentiate relation types. Topics estimated from an unsupervised model may correspond to key words from different instances. But they cannot tell us what relation types these topics correspond to. Based on such intuition, we decided to take Labeled LDA as our learning model.

6.3 Labeled LDA

LLDA is proposed by [41]. It aims at predicting credit attribution in multi-labeled corpora. Namely, in their corpora, each document has multi-labels. Label information is observed.

Like Latent Dirichlet Allocation, Labelled LDA (LLDA) models each document as a mixture of underlying topics and generates each word from one topic. However, unlike LDA, LLDA incorporates supervision by simply constraining the topic model to use only those topics that correspond to a document's (observed) label set.

Their model can be described as follows. Let each document d be represented by a tuple consisting of a list of word indices $\mathbf{w}^d = (w_1, ..., w_{N_d})$ and a list of binary relation type presence/absence indicators $\Lambda^{(d)} = (l_1, ..., l_K)$ where each $w_i \in \{1, ..., V\}$ and each $l_k \in \{0, 1\}$. N_d is the document length, V is the vocabulary size and K the total number of unique labels (the number of topics as well) in the corpus.

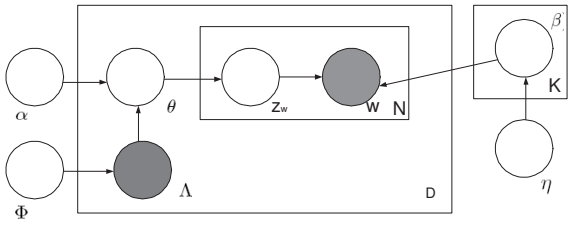

Figure 6.2: Graphical model of LLDA

[1] we have files to compare output of LDA and relation types of ACE. They cost too much space. So, we didn't put them into the report.

Steps 1 and 2 is to draw the multinomial topic distributions over vocabulary β_k for each topic k, from a Dirichlet prior η. The traditional LDA model then draw a multinomial mixture distribution $\theta^{(d)}$ over all K topics, for each document d, from a Dirichlet prior α. LLDA would restrict θ^d to be defined only over the topics that correspond to its label $\Lambda^{(d)}$. Since the word-topic assignments z_i are drawn from this distribution, this restriction ensures that all the topic assignments are limited to the document's label.

this objective is done with the following steps. Use a Bernoulli coin toss for each topic k, with a labelling prior probability Φ_k. Next, define the vector of document's labels to be $\lambda^d = \{k|\Lambda_k^{(d)} = 1\}$. This allows us to define a document-specific label projection matrix $L^{(d)}$ of size $M_d \times K$ for each document d, where $M_d = |\Lambda^{(d)}|$. Note, M is the number of documents. But M_d is the number when the document is assigned 1 for a topic.

For each row $i \in \{1,...,M_d\}$ and column $j \in \{1,...,K\}$:

$$L_{ij}^{(d)} == \begin{cases} 1 & \text{if } \lambda^{(d)} = j, \\ 0 & \text{if } otherwise, \end{cases}$$

Namely, the i^{th} row of L^d has an entry of 1 in column j if and only if the i^{th} document label $\lambda_i^{(d)}$ is equal to the topic j and zero otherwise. L^d matrix is used to project the parameter vector of the Dirichlet topic prior $\alpha = (\alpha_1,...,\alpha_{\mathbf{k}})^{\mathbf{t}}$ to lower dimensional vector $\alpha^{(d)}$ as follows:

$$\alpha^{(d)} = L^{(d)} \times \alpha = (\alpha_{\lambda_1}(d),...,\alpha_{\lambda_{M_d}}(d))^T \tag{6.3}$$

Then, $\theta^{(d)}$ is drawn from a Dirichlet distribution with parameters $\alpha^{(d)} = L^{(d)} \times \alpha = (\alpha_2, \alpha_3)^T$.

For example, suppose $K = 4$ and that a document d has labels given by $\Lambda^{(d)} = \{0,1,1,0\}$ which implies $\lambda^{(d)} = \{2,3\}$, then $L^{(d)}$ would be:

$$\mathbf{L^{(d)}} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Then, $\theta^{(d)}$ is drawn from a Dirichlet distribution with parameters $\alpha^{(d)} = L^{(d)} \times \alpha = (\alpha_2, \alpha_3)^T$.

6.4 Adapting LLDA for Relational Discovery

As we can see, the only difference between LLDA and LDA lies in the labels. Lables in LLDA are observed while in LDA are unknown. In our work, our label set are relation types. But

unlike LLDA defined in last section, only one relation type is involved in most instances. So, we don't need a Bernulli to sample the labels. We actually treat lambda as fully observed in our experiments. Due to this difference, we can drop the Φ out. Only a definite Λ is left as displayed in figure 6.3.

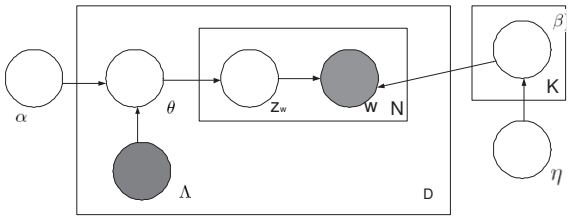

Figure 6.3: Graphical model of LLDA-R

As the common LDA, our goal it to estimate θ and z. And also like common LDA, estimation of posterior distribution of θ and z is intractable. Therefore, approximation is still a must. We employed Gibbs sampling (Griffiths and Steyvers, 2004). The full conditional equation used for sampling individual z_i values from the posterior is given by

$$P(z_i = v|\mathbf{z}_{-i}, \mathbf{w}, \alpha, \beta) \propto \left(\frac{n^d_{-i,v} + \alpha}{\sum_u^T (n^d_{-i,u} + \alpha)}\right)\left(\frac{n^{w_i}_{-i,v} + \beta}{\sum_{w'}^W (\beta + n^{w'}_{-i,v})}\right) \tag{6.4}$$

where $n^d_{-i,v}$ is the number of times topic v is used in document d, and $n^{w_i}_{-i,v}$ is the number of times word w_i is generated by topic v. The *-i* notation signifies that the counts are taken omitting the value of z_i. [41] implements their LLDA with Gibbs sampling as well. But they sample the label based on the $\alpha^{(d)}$ which is sampled from a Bernulli sampling of Φ. $\alpha^{(d)}$ can be zero or one thereafter. Namely, if $\alpha^{(d)}$ is 1, θ_d is fully sampled and subsequently, z_d is sampled. Otherwise, z_d will not be sampled.

In our labelled LDA, labels have only one unique value. If we follow LLDA of [41], only one label is sampled each time for one instance and thus most of labels will not be sampled. This will lead to insufficient sampling for θ and z then. As a result, in our experiment, sampling this way led to poor results. As an adaptation, we use λ to denote the indicator set of labels. We made revisions on original LLDA as follows. Let us use q_{iv} to represent above equation. We can rewrite the above equation as:

$$P(z_i = v|\mathbf{z}_{-i}, \mathbf{w}, \alpha, \beta) \propto q_{iv}\lambda(v \in C^i) \tag{6.5}$$

Next, we relaxed the constraint by using a sigmoid function to assign some ratios to where λ is 0.

The sigmoid function is in nature a kind of logistic regression with the equation as follows.

$$\lambda = \frac{e^z}{e^z + 1} = \frac{1}{1 + e^z} \tag{6.6}$$

After the topic model are learned from the training set, we can perform inference on unseen document still using Gibbs sampling restricted to its tags, to determine its per-word label assignment $\mathbf{z}$. Meanwhile, the posterior distribution θ over topics can be computed by appropriately normalizing the topic assignments $\mathbf{z}$.

Chapter 7

Entity relation detection using supervised topic models with maximum margin learning

7.1 Introduction

In this work, we explore how the latent semantics of the text can help in detecting entity relations. Specifically, we are interested in underlying topic distributions. Intuitively, topics such as marriage or birth could be indicative of a Social relationship between the participating entities. Similarly, topics such as nationality or recruitment could be indicative of an Affiliation relation between participating entities. Thus in this paper, we investigate if hidden topic distributions indicative of entity relations can be effectively learned.

In order to achieve this, we adapt the Latent Dirichlet Allocation (LDA) approach to solve the ERD task. There are a number of challenges in employing the LDA framework for ERD. Primarily, the basic LDA model is unsupervised, and hence the discovered topics may not help classification. This problem has been solved in supervised models such as Labeled LDA (LLDA) [41] and supervised LDA (sLDA) [42]. While LLDA and sLDA are powerful generative models that capture the underlying semantics of texts pertinent to classes of interest, they have trouble discovering marginal classes and easily employing rich feature sets, both of which are important for ERD.

In order to incorporate the desired capabilities, we build our ERD system, *ERD-MEDLDA*, based on Maximum Entropy Discriminant Latent Dirichlet Allocation (MEDLDA) from [43]. MEDLDA is a supervised extension of LDA that combines the capability of capturing latent semantics with maximum margin learning. Specifically, MEDLDA is a combination of sLDA and support vector machines (SVMs); thus, it integrates maximum likelihood estimation (MLE) and maximum margin estimation (MME). Further, in order to employ rich and heterogeneous features we introduce a separate exponential family distribution for each feature, similar to [44], into our ERD-MEDLDA model.

The relation detection task is formulated within the topic model framework in ERD-MEDLDA as follows. Occurrences of pairs of NE mentions[1] in a document and the text between them is considered as a *mini-document*. Each mini-document has a relation type (analogous to the response variable in the supervised topic model). The supervised topic model discovers a latent topic representation of the mini-documents and a response parameter distribution. The topic representation is discovered with observed response variables during training, which influences topic discovery towards the response variables. During prediction, the topic distribution of each mini-document can form a prediction of the relation types.

We carry out experiments to measure the effectiveness of our approach and compare it to SVM-based and LLDA-based models, as well as to a previous work using the same corpora. We also measure and analyze the discovered topics and the effectiveness of incorporating different features in our model relative to other models.

Our approach exhibits better overall precision, recall and Fmeasure than baseline systems, and shows better overall performance than state-of-the-art kernels. We also find that the ERD-MEDLDA shows consistent capability for incorporation and improvement due to a variety of heterogeneous features.

The rest of the paper is organized as follows. We describe the proposed model in Section **??** and the features that we explore in this work in Section 7.4. Section 7.3 describes the data, Section 7.5 presents the experiments and Section 7.6 presents analyses. We discuss the related work in Section 7.7 before concluding in Section 7.8.

[1] Adopting terminology used in the Automatic Context Extraction (ACE) program [6], specific NE instances are called *Mentions*.

7.2 ERD-MEDLDA

ERD-MEDLDA is based on the principle of hierarchical Bayesian models. The basic LDA is an unsupervised model and the resulting topics may not help with classification tasks. However, with the explicit addition of supervised information (such as response variables), the resulting topic models have good predictive power for classification and regression. MEDLDA model from [43] is one such extension of LDA, where the class information is added to the model. Further, MEDLDA integrates max margin learning and topic models by optimizing a single objective function with a set of expected margin constraints.

In this section, we explain the development of ERD-MEDLDA as follows: The MEDLDA model from Zhu et al. [43] and the modifications to it are explained in Section 7.2.1. Section 7.2.2 describes further modifications that allow for the incorporation of heterogeneus features, Section 7.2.3 describes inference and estimation procedures and finally Section 7.2.4 describes how relation detection is performed under the model.

The following notation is used in this paper:

- K - the total number of topics
- $\alpha_{1:K}$ - K-dimensional parameters of a Dirichlet distribution
- $\beta_{1:K}$ - Parameters for K component distribution over the words
- $\theta_{1:K}$ - K-dimensional parameters topic distribution variables over a document
- N - the total number of words and features in a document
- $Z_{d1:dN}$ - A finite set of random variables which represents a sequence of topics in a document
- $W_{d1:dN}$ - A finite set of observed variables and $w_{d1:dn}$ or simply $\mathbf{w}$ represent a specific document
- D - a collection of documents denoted by $D = \mathbf{w_1}, \mathbf{w_2}, ..., \mathbf{w_D}$
- C - the total number of relation types
- $\eta_{1:C}$ - Parameters for C component distribution over the relation types

- $Y_{1:C}$ - A finite set of observed variable which represents relation types

For simplicity we would use bold case symbols for vectors and drop the subscript where dimensionality is unambiguous.

7.2.1 MEDLDA

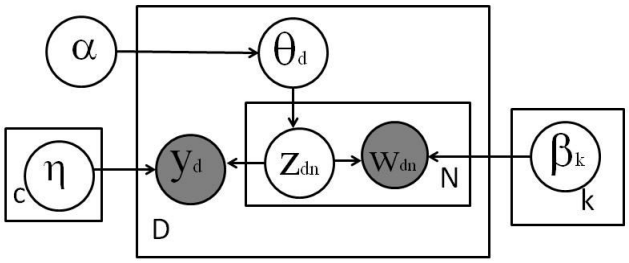

Figure 7.1: MEDLDA

The MEDLDA model described in [43] is illustrated in Figure 7.1. In a collection of documents D, each document $w_{1:N}$ is generated from a sequence of topics $z_{1:N}$. θ is a k-dimensional topic distribution variable, which is sampled from a Dirichlet distribution $Dir(\alpha)$. Like common LDAs, MEDLDA uses independence assumption for a finite set of random variables $Z_1, ..., Z_n$ which are independent and identically distributed, conditioned on the parameter, θ. Like sLDA, MEDLDA is a supervised model. A response variable Y connected to each document is added for incorporating supervised side information. The supervised side information is expected to make MEDLDA topic discoveries more interpretable. Zhu, Ahmed and Xing's [43] MEDLDA model can be used in both regression and classification. Concretely, Y is drawn from $\eta_{1:c}$, a $c \times k$-dimensional vector and the topic distribution $z_{1:N}$. Note that the plate diagram for MEDLDA is quite similar to sLDA [42]. But there is a difference – sLDA focuses on building regression models, and thus the response variable Y in sLDA is generated by a normal distribution. In regression, similar to sLDA, a normal distribution is used for generating Y while in classification, MEDLDA uses the max-margin principle to directly generate Y.

Based on the plate diagram, the joint distribution of latent and observable variables for

our MEDLDA-based relation detection is given by

$$p(\theta, \mathbf{z}, \mathbf{w}, \mathbf{y} | \alpha, \beta_{1:k}, \eta_{1:c}) = \prod_{d=1}^{D} p(\theta_d|\alpha) \times \Big(\prod_{n=1}^{N} p(z_{dn}|\theta_d) p(w_{dn}|z_{dn}, \beta_{1:k}) \Big) \times p(y_d | z_{d1:dN}, \eta_{1:c}) \quad (7.1)$$

Another important difference from sLDA lies in the fact that MEDLDA does joint learning with both MME and MLE. The joint learning for classification is done in two stages, unsupervised topic discovery and multi-class classification. During training, EM algorithms are utilized to infer the posterior distribution of the hidden variables θ, $\mathbf{z}$ and η. In testing, the trained models are used to predict relation types $\mathbf{y}$.

7.2.2 Fine Mixed Membership MEDLDA

MEDLDA is already a mixed membership model and although the MEDLDA model described above can be applied to the detection and classification tasks, we felt a few modifications are necessary before it can be effective in predicting relation types. In particular, MEDLDA is defined for using a homogeneous component distribution and we required it to use heterogeneous features.

As we can see from the plate model in Figure 7.1, each of $w_{n\in 1:N}$ is assumed to be generated from one of the discrete component distributions. In each document, bag of words are the same type of objects. The set of distributions remain the same across all words. Thus LDA is designed to handle data points with homogeneous features such as words. But previous work in relation detection has shown that it is important to incorporate part-of-speech tags, named entities, grammatical dependencies and other linguistic features. We achieve this by introducing a separate exponential family distribution for each feature similar to [44]. Our MEDLDA-based relation detection model is really a mixed-member Bayesian network. Figure 7.2 illustrates our model with this extension.

Figure 7.2 is very similar to Figure 7.1. There are two differences: one is that the topic component number k is now kN and the other is that there is another component η_0 before the response parameter η. The first one is the revision we made to incorporate heterogenous features. Note that now we have β^d_{ni} rather than only β^d_i since we have drawn separate distributions for each word (or feature) n.

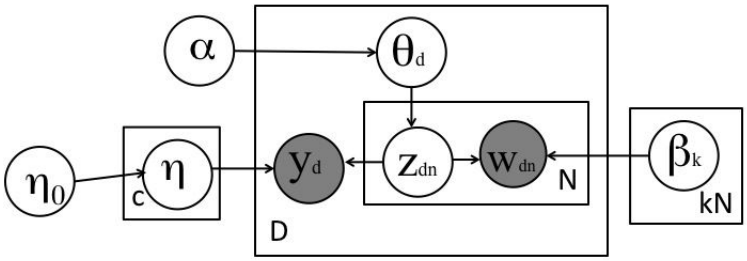

Figure 7.2: Fine Mixed Membership MEDLDA

In MEDLDA, Zhu and Xing [43] have in fact introduced the idea that η is sampled from a prior $p_0(\eta)$. We still follow the idea and draw it as a hyper-parameter in Figure 7.2 for clarity. Like MEDLDA, assuming that there are c classes and k topics. In our work, c is the number of relation types. Different from MEDLDA, the response parameter $\eta_{1:c}$ in our model is a matrix with c k-dimensional softmax parameters as the row.

The generative process for each document in this model is similar to that given in [44] and is as follows:

1. Sample a component proportion $\theta_d \sim$ **Dirichlet**(α),
2. For each feature like word, part-of-speech, named entity in the document,
   (a) For $n \in \{1, ..., N\}$, sample $z_{dn} = i \sim$ **Discrete**(θ_d)
   (b) For $n \in \{1,...,N\}$, sample $w_{dn} \sim P(w_{dn}|\beta^d_{ni}, z_{dn})$
3. Sample the relation type label for a document from a softmax distribution , two steps are involved,
   (a) For $j \in \{ 1, ... C\}$, sample $\eta_j \sim \mathbf{N}(0, \eta_0)$

(b) For $j \in \{1, \ldots C\}$, $y_d \sim softmax(\frac{exp(\eta_j^T \bar{z})}{\sum_{j=1}^{c} exp(\eta_j^T \bar{z})})$

In step 2, index i is the number of the topic component which ranges from $1 : K$. $P(w_{dn}|\beta_{ni}^d, z_{dn})$ in 2(b) is an exponential family distribution.

Like the original MEDLDA, we make use of a prior $p_0(\eta)$ where $p_0 = N(0, \eta_0)$ to sample the response parameter η. That is, η is taken as a variable rather than a hyper-parameter like α and β. Instead, η_0 is the hyperparameter for η. It will be sampled c times. Namely, for each relation type, a vector of response parameters will be sampled and they will in turn be used to sample relation types.

For the sampling in step 3b, a softmax distribution is employed. Softmax is an variation of logistic regression. Logistic regression is usually used for binary classification while softmax is for multi-class classification. Softmax is a mature statistical method for classifications, easy to compute and is appropriate for handling missing features. The input for the softmax distribution is $\bar{z}$, namely, the mean of z for all words/features.

Note that we do not directly use maximum margin principle for classification. This is another revision we made in the model. In [43], their model for classification include two separate parts – the first part is an unsupervised LDA which does MLE for modeling topics and the second part is an SVM which does MME for classification. Zhu and Xing's motivation of doing so is that calculations of partition factors or normalization factors are too hard and too slow. High speed can be achieved. However, we hold that this separation cannot make full used of advantages of integration of MME and MLE. And meanwhile, in the learning process, topic discoveries biased by supervised side information should be more helpful than topics discovered learned in unsupervised fashion. Though the learning of normalization factor for classification may be slower, it is not intractable and the speed can be improved by selecting suitable sampling methods and by good optimizations and approximations.

Thus, a joint distribution can be written as:

$$p(\theta, \mathbf{z}, \eta_{1:c}, \mathbf{w}, \mathbf{y}|\alpha, \beta_{1:k})) = p_0(\eta)\, p(\theta, \mathbf{z}, \mathbf{w}, \mathbf{y}|\alpha, \beta_{1:k}, \eta_{1:c}) \tag{7.2}$$

The second term is the same as that defined in Equation 7.1. The first term, i.e. the prior distribution of η is defined as a a normal distribution.

The density function for $\mathbf{w}, \mathbf{y}$ after θ and $\mathbf{z}$ are integrated out is given by

$$p(\mathbf{w},\mathbf{y}|\alpha,\beta_{1:k},\eta_{1:c}) =$$
$$\prod_{d=1}^{D}\int_{\theta_d} p_0(\eta)p(\theta_d|\alpha)\prod_{n=1}^{N}\sum_{k=1}^{K}(p(z_{dn=k}|\theta_d)p(w_{dn}|\beta_{nk},z_{dn})))p(y_d|\mathbf{z}_d,\eta_{1:c})d\theta_d \quad (7.3)$$

Since the relation detection only involves token features though these features are heterogenous in nature, ranging from bag of words, parts-of-speech, chunk types and so on, they are discrete symbols. Hence, before we find continuous features, such as numerical values, all features have discrete distributions.

With this extension, the distribution for generating w_{dn} not only depends on z_{dn}, but also depends on what kind of features employed. Therefore, by choosing an appropriate exponential family distribution for each feature (in our relation detection, all features involve discrete distributions with diverse parameters), our ERD-MEDLDA model can integrate diverse features of different types or the same features with different parameters.

7.2.3 Inference and Estimation

In [43], MEDLDA integrates a Bayesian sLDA and a support vector machine for both MLE and MME. In our work. we follow the same approach. But as shown in our generative process, instead of only using unsupervised LDA for topic modeling, we use fully supervised MEDLDA for both topic modeling and the final classification. Yet, the inference is slower because of the normalization factor in the probability distribution of y and also because we draw different distributions for each word. To remedy this, we make use of a strategy similar to Shan et al. [44]. Namely, for features of the same type, we sample them with the same exponential distributions and also we sample them with parameters averaged from the training data which makes the learning much faster and efficient.

In our model, the learning task is to obtain an optimal set of parameters α, β and $p(\eta)$ such that the likelihood of observing the whole feature set and the relation types $p(\mathbf{w},\mathbf{y}|\alpha,\beta,p(\eta))$ are maximized. However, the calculation of the likelihood function 7.3 is intractable. Following the generative process, parameter estimation and inferences can be made with either Gibbs sampling or EM-based variational methods. We use variational methods since we adapt MEDLDA package[2] to mixed-membership MEDLDA and train relation detection

[2] this package is downloaded from http://www.cs.cmu.edu/ junzhu/MEDLDA.htm

models.

Specifically, to obtain a tractable lower bound, we consider an entire family of parameterized lower bounds with a set of free variational parameters, and pick the best lower bounds by optimizing the lower bound with respect to the free variational parameters.

Variational Inference

The EM-based variational methods involves E-step and M-step. In the E-step, the latent variable distributions are computed while in the M-step, parameter estimation is done by maximizing the expectation of the complete likelihood distribution. As we know, in order to use MEDLDA to make predications on relation type, the key inferential problem that we need to solve is that of computing the posterior distribution of the hidden variables given a document,

$$p(\theta, \mathbf{z}, \eta_{1:c}|\mathbf{w}, y, \alpha, \beta_{1:k}) = \frac{p(\theta, \mathbf{z}, \eta_{1:c}, \mathbf{w}, y|\alpha, \beta_{1:k})}{p(\mathbf{w}, y|\alpha, \beta_{1:k}, \eta_{1:c})} \tag{7.4}$$

where, the denominator is Equation 7.3 which is the probability of observed variables given all parameters and the numerator is the probability of all observed variables plus sampled parameters given hyper-parameter for one document .

The partition function obtained from 7.3 and 7.4 is intractable, thus cannot be computed in closed form. Hence, the same approximation as in [43] is made. The upper bound (formalized as $L^{bs}(q)$) of the negative log-likelihood $-log\ p(\mathbf{w}, y|\alpha, \beta, \eta)$ is given,

$$\begin{aligned} L^{bs}(q) &= -E_q[log\ p(\theta, \mathbf{z}, \eta, \mathbf{y}, \mathbf{w}|\alpha, \beta, p_0(\eta))] - H(q(\theta, \mathbf{z}, \eta)) \\ &= KL(q(\eta)||p_0(\eta) + E_{q(\eta)}[L^s] \end{aligned} \tag{7.5}$$

where L^s refers to the likelihood function of supervised LDA and $KL(q(\eta)||p_0(\eta) = E_{p_0\eta} log(p_0\eta/q(\eta))$ is the Kullback-Leibler (KL) divergence. Based on Figure 7.2, for each data point $\mathbf{w}_d, y_d$, the joint probability of L^s can be factored as,

$$\begin{aligned} &E_q[log\ p(\theta, \mathbf{z}, \eta, \mathbf{y}, \mathbf{w}|\alpha, \beta, p_0(\eta))] \\ &= E_q[log\ p(\theta_d|\alpha)] + E_q[log\ p(z_d|\theta_d)] + E_q[log\ p(w_d|z_d, \beta)] + E_q[log\ p(y_d|\bar{z}, q(\eta) \end{aligned} \tag{7.6}$$

The second term of equation 7.5, $E_{q(\eta)} L^s = q(\theta, z, \eta|\gamma, \phi)$ is the variational form the original

log-likelihood form. The expanded form of that term is,

$$E_{q(\eta)}L^s = q(\eta)\prod_{d=1}^{D} q(\theta_d|\gamma_d)\prod_{n=1}^{N} q(z_{dn}|\phi_{dn}). \tag{7.7}$$

where γ_d is a K-dimensional vector of Dirichlet parameters, and each ϕ_{dn} is a categorical distribution over K topics. Namely, γ_d is the approximation of θ, and ϕ_{dn} is the approximation of z_{dn}. Denoting the lower bound for each data point $\mathbf{w}_d, y_d$ with $L(\gamma_d, \phi_d; \alpha, \beta, q(\eta))$, Equation 7.5 can be expanded as,

$$\begin{aligned} L(\gamma_d, \phi_d; \alpha, \beta, q(\eta)) = {} & E_q[log\ p(\theta_d|\alpha)] + E_q[log\ p(z_d|\theta_d)] + E_q[log\ p(w_d|z_d, \beta)] \\ & - E_q[log\ q(\theta_d|\gamma_d)] - E_q[log q(\mathbf{z_d}|\phi_d)] + E_q[log\ p(y_d|\bar{z}, q(\eta))] \end{aligned} \tag{7.8}$$

Then, with exactly the same idea of integration of MLE and MME as that in [43], we can define the integrated training of ERD-MEDLDA as:

$$\begin{aligned} & min_{q,q(\eta),\alpha,\beta,\xi} E_{q(\eta)}[L^s] + KL(q(\eta)||p_0\eta) + C\sum_{d=1}^{D}\xi_d \\ & s.t.\ d, y \neq y_d : E[\eta^T(f(y_d, \bar{z}) - f(y, \bar{z}))] \geq 1 - \xi_d; \xi_d \geq 0, \mu_d \end{aligned} \tag{7.9}$$

where μ is lagrange multiplier, ξ is the slack variable tolerating errors in training data. y_d is the true label while y is the prediction. So, $f(y_d, \bar{z}_d) - f(y, \bar{z}_d)$ is the difference between truth and prediction which we call expected margin. Namely, the former is the average boundary composed of true labels and the latter is the average boundary of predicated labels. The smaller the difference is, the closer to the true labels.

That is essentially the advantage of MEDLDA over other MLE or max-entropy models. Meanwhile, since MEDLDA also employs regular MLE for data generated from sampling, it enjoys advantages of both kinds of learning. That is, it takes care of sufficient statistics as well as handles examples that are around the decision boundary with support vectors. When lagrange multipliers are not zeros, terms related to them act as a regularizer, biasing the model towards discovering a latent representation. Thus, more accurate predictions will be obtained on difficult examples located in decision boundaries. These latent representations are fixed for words in a document and therefore, yield much more discriminant powers.

Then, the Lagrangian L of equation 7.9 is identical to the one for classification as in [43] except that the first term is supervised rather than unsupervised. This equation is reproduced below for readability.

$$L = L(q)^s + KL(q(\eta||p_o(\eta))) + C\sum_{d=1}^{D}\xi_d - \sum_{d=1}^{D}\upsilon_d\xi_d - \sum_{d=1}^{D}\sum_{y\neq y_d}\mu_d(y)(E[\eta^T\Delta f_d(y)]) + \xi_d - 1 - \sum_{d=1}^{D}\sum_{i=1}^{N}c_{di}(j=1)^K\phi_{dij} - 1) \quad (7.10)$$

where $\Delta f_d(y) = f(y_d, \bar{z}_d) - f(y, \bar{z}_d)$ and the last term is from the normalization condition $\sum_{j=1}^{K}\phi_{dij} = 1, \forall_i, d$.

Parameter estimation

In last section, we constructed variational parameters for approximating hidden variables θ, $\mathbf{z}$ and η. Then, following this line, the EM algorithm will iteratively solve the approximation problem with two steps:

1. $E-step$: infer the posterior distribution of the hidden variables θ, $\mathbf{z}$ and η, where, for θ and $\mathbf{z}$, inferring the posterior distributions of them are in fact to fit the variational parameters ϕ and γ while for η, more complex issues will be involved. More discussion is given in next section

2. $M-step$: estimate the unknown model parameters α, β

The update rules are in fact done by sequentially taking partial derivative of Equation 7.10 over related variables. These rules are summarized below for readability:

- Optimize L over γ
- Optimize L over ϕ
- Optimize L over $q(\eta)$
- Optimize L over α
- Optimize L over β

Since the constraints in equation 7.10 are not on θ (its variational γ), α or β, the update rules are the same as LDA or sLDA. The update of ϕ is the same as MEDLDA in [43] where the product of Lagrangian multiplier μ and $E[\eta^T(f(y_d,\bar{z})-f(y,\bar{z}))]$ makes the update rule of ϕ has the maximum margin nature explained above. But since we will add more variational parameters to approximate $q(\eta)$, the final update rule of ϕ is different.

The hardest part is the update of $q(\eta)$. This is because η, as the parameter of the response variable y or the relation type y, is coupled with $\bar{z}$. In our general process, the relation type label y_d is sampled from a soft-max or a multi-class logistic regression $softmax(\eta_1^T\bar{z},\eta_2^T\bar{z},...,\eta_c^T\bar{z})$. Namely, y_d is generated from a discrete distribution $[p_1,p_2,...,p_c,\sum_{j=1}^{c}p_j]$ with $p_j=\frac{exp(\eta_j^T\bar{z})}{\sum_{j=1}^{c}(\eta_j^T\bar{z})}$. Hence, the probability mass of relation type y_d given $\mathbf{z}_d$ and η is,

$$p(y_d|\mathbf{z}_d,\eta)=exp(\sum_{j=1}^{c}\eta_j^T\bar{z}_d y_{dj}-log(\sum_{j=1}^{c}exp(\eta_j^T\bar{z}_d)) \tag{7.11}$$

Accordingly, the last term in equation 7.8 is,

$$E_q[log\ p(y_d|\mathbf{z_d},q(\eta))]=\sum_{j=1}^{c}\sum_{k=1}^{K}\eta_{jk}E_q[\bar{z}_{dk}]y_{dj}-E_q[log(\sum_{j=1}^{c}exp(\eta_j^T\bar{z}_d))] \tag{7.12}$$

Even after introducing the variational distribution q above, the above equation cannot be efficiently computed. Consequently, further approximation must be done. Thus, a new variational parameter δ is introduced to obtain a further lower bound for it. Specifically, besides Jensen inequality, another inequality that $-log(x)\geq 1-\frac{x}{\delta}-log(\delta)$ [45] is used here to lower bound the term as,

$$E_q[log\ p(y_d|\bar{z},q(\eta))]\geq\sum_{j=1}^{c}\sum_{k=1}^{K}\eta_{jk}E_q[\bar{z}_{dk}]y_{dj}-\frac{1}{\delta}E_q[\sum_{j=1}^{c}exp(\eta_j^T\bar{z}_d)]+(1-\frac{1}{\delta}-log\delta), \tag{7.13}$$

Thus with the addition of another variational parameter, maximizing the lower-bound lagrange function 7.10 with respect to the variational parameters will now give the update equation of γ_d, ϕ_d, η and δ_d . The update rule of δ_d is similar to γ_d and ϕ_d. A simple partial derivation of equation 7.10 over δ will give its update equation as,

$$\delta_d=1+\sum_{j=1}^{c}\sum_{k=1}^{K}\phi_{dk}exp(\eta_{jk}) \tag{7.14}$$

As we see from Equation 7.13, δ and ϕ are coupled together now. Hence, in the update rule of δ, ϕ is part of it. Consequently, the update rule of ϕ is not separate from δ.

$$\phi_{di} \propto exp(E[log\theta|\gamma]) + E[log\ p(w_{di}|\beta)] + \frac{1}{N}\sum_{y\neq y_d}\mu_d(y)E[(\eta_{yd} - \eta_y)/\delta_d] \tag{7.15}$$

Now, we can obtain the optimization of $q(\eta)$ by setting $\partial L/\partial q(\eta) = 0$

$$q(\eta) = \frac{p_0(\eta)exp(\eta^T\sum_{d=1}^{D}(\mu_d(y))(E[\bar{z}_d/\delta_d]) + \sum_{y\neq y_d}E[\Delta f_d(y)])}{\mathbf{Z}}$$

The lagrange multiplier μ is the optimum solution of the dual problem:

$$D1 : max_\mu - logZ + \sum_{d=1}^{D}\sum_{y\neq y_d}\mu_d(y)$$
$$s.t.\forall d : \sum_{y\neq y_d}\mu_d(y) \in [0, C], \tag{7.16}$$
$$\tag{7.17}$$

As mentioned before, we use standard normal prior for $p_0(\eta) = N(0, I)$. According to the principle of conjugate prior, $q(\eta)$ is a normal distribution with a shifted mean as $q(\eta) = N(\lambda, I)$, where $\lambda = \sum_{d=1}^{D}\sum_{y\neq y_d}(E(\bar{z}_d) + E[\Delta f_d(y)]$. The dual problem D1 can be solved using existing multi-class SVM methods as:

$$max_\mu - \frac{1}{2}||\sum_{d=1}^{D}\sum_{y\neq y_d}\mu_d(y)(E[\bar{z}_d/\delta_d]) + \sum_{y\neq y_d}E[\Delta f_d(y)])||_2^2 + \sum_{d=1}^{D}\sum_{y\neq y_d}\mu_d(y)$$
$$s.t.\forall d : \sum_{y\neq y_d}\mu_d(y) \in [0, C],$$
$$\tag{7.18}$$

7.2.4 Relation Detection

With the generative process, inference and parameter estimation in place, ERD-MEDLDA is ready to perform relation detection. The first step is to perform variational inference given the testing instances.

In classification, we estimate the probability of the relation type given topics and the response parameters, i.e. $p(y_d|z_{d1:dN},\eta_{1:C})$. Using variational approximation described in the previous section, we can derive the prediction rule as $F(y,z_{1:N},\eta)=\eta^T f(y,\bar{z})$ where $f(y,\bar{z})$ is a feature vector. Now, SVM can be used to derive the prediction rule. The final prediction can be generalized exactly the same as Zhu, Ahmed and Xing [43]:

$$\hat{y}=argmax_y E[\eta^T f(y,\bar{z})|\alpha,\beta] \tag{7.19}$$

Recall that we make use of softmax regression in sampling relation types; consequently the derivation rules are given as,

$$E[logp(y=i|w_{1:N},\alpha,\beta,\eta_{1:C})] = \begin{cases} \eta_i^T E[\bar{z}] - E[log(\sum_{i=1}^{C} exp(\eta_i^T \bar{z}))] & [i]_1^C \\ 0 - E[log(\sum_{i=1}^{C} exp(\eta_i^T \bar{z}))] & i=C. \end{cases} \tag{7.20}$$

The term $E[\bar{z}]$, like in model learning, is intractable. Therefore, similar variational distributions are introduced. Namely, we use $E_q(\bar{z})=q(\theta,z_{1:N})$ to approximate $E[\bar{z}]$.

7.3 Data

We use the ACE corpus (Phase 2, 2005) for training and evaluation. The ACE corpus has annotations for both entities and relations. The corpus has six major relations types, 23 subtypes and 7 entity types. In this work, we focus only on the six high-level relation types listed in Table 7.1. In addition to the the 6 major types, we have an additional category – no relation (NO-REL) that exists between entities that are not related.

Named entities within a sentence are paired, and all text in between and including the NEs is considered as a mini document. The gold standard annotation of their ACE relation type, or NO-REL if no relation exists, forms the mini document's label. For instance, consider the following sentence (all NEs are shown in *italics*)

John is married to *Liz* who works in *Xyz Corp* located in *New York*.

All NE pairs, the corresponding mini documents and their labels constructed from this sentence are listed in Table 7.2

Major Type	Definition	Example
ART artifact	User, owner, inventor or manufacturer	the makers of the Kursk
GEN-AFF	citizen, resident, religion,ethnicity and organization-location	U.S. Companies
ORG-AFF (Org-affiliation)	employment, founder, ownership, sports-affiliation, investor-shareholder, student-alumni and membership	The CEO of Siemens
PART-WHOLE	geographical, subsidiary and so on	a branch of U.S bank
PER-SOC (person-social)	business, family and lasting personal relationship	a spokesman for the senator
PHYS (physical)	located or near	a military base in Germany

Table 7.1: Relation types for ACE 05 corpus

NE pair	Mini Document	Annotation
John-Liz	John is married to Liz	PER-SOC
Liz-Xyz Corp	Liz who works in Xyz Corp	ORG-AFF
Liz-New York	Liz who works in Xyz Corp located in New York	GEN-AFF
Xyz Corp- New York	Xyz Corp located in New York	PHYS
John-Xyz Corp	John is married to Liz who works in Xyz Corp	NO-REL
John-NewYork	John is married to Liz who works in Xyz Corp located in New York	NO-REL

Table 7.2: NE pairs, Mini documents and labels for a sample sentence"

All relations in the ACE corpus are intra-sentential and hence we do not create NE pairs that cross sentence boundaries. Also, almost all positive instances are within two mentions of each other. Hence, we create NE pairs for only those NEs that have at most 2 intervening NEs in between. This gives us a total of 38,342 relation instances of which 32,640 are negative instances and 5912 are positive relation instances belonging to one of the 6 categories. The distribution of the 6 ACE categories is given in Table 7.3.

Relation Type	Count	Distribution
ART	536	0.09
GEN-AFF	746	0.126
ORG-AFF	1762	0.298
PART-WHOLE	926	0.157
PER-SOC	911	0.15
PHYS	1031	0.174

Table 7.3: Distributions of Relation Types

7.4 Features

We explore the effectiveness of incorporating features into our system as well as the baselines. For this, we construct feature sets similar to Jiang and Zhai [46] and Zhou [47]. Three sets of features are employed: Bag Of Words (BOW), Syntactic (SYN) and Composite (COMP).

7.4.1 Bag of Words Features (BOW)

The Bag of Words (BOW) feature captures all the words in our mini-document. Consider, for example, the following text snippet that reveals an affiliation relation (ORG-AFF).

(1) *X*, the president of the *United States [ACE relation type: ORG-AFF]*

Notice that words such as “of” can be indicative of the ORG-AFF relation. BOW features capture this lexical information. However, compared to traditional classification settings, there is a difference in how ERD-MEDLDA employs these features. BOW features are trained as topics and then the discovered topics will be employed as features. Thus words such as

"of" may fall into topic(s) that, in turn, would eventually contribute to the recognition of AFF-ORG class.

7.4.2 Syntactic Features (SYN)

The SYN features are constructed to capture syntactic, semantic and structural information of the mini-document. Let us consider the following text snippet:

(2) *X*, the president, visited the *United States [ACE relation type: no relation]*

Notice that even though examples 1 and 2 exhibit different relation types, they share a large number of words. Now, observe that the words that indeed differ between the two text snippets, "of" and "visited", also differ in their part of speech: one is a preposition (IN), while the other is a verb (VBD). Thus, we include part of speech (POS) information of the words between two NEs to additionally clue the system to the type of relation that might exist between them.

We observed that the syntactic roles are also indicative of relationships between entities. For instance, in Example 1, *the president* is the subject of *the United States*. We encode dependency features to capture this information.

Other syntactic features employed by the systems include:

- *HM1:* The head word of the first mention
- *HM2:* The head word of the second mention
- *ET1:* Entity type of the first entity
- *M1:* Mention Type of the first entity
- *#MB:* Number of other mentions in between the two mentions under consideration
- *#WB:* Number of words in between the two mentions.

7.4.3 Composite Features (COMP)

The Composite features (COMP) are similar to SYN, but they additionally capture order and dependencies between the features mentioned above. Ordering of words are not captured by BOW or SYN. This feature exchangeability works for models based on random or seeded

sampling (e.g. LDA) – as long as words sampled are associated with a topic, the hidden topics of the documents can be discovered. In the case of ERD, this assumption might work with symmetric relations. However, when the relations are asymmetric, ordering information is important. Besides exchangeability, LDA-based models also assume that words are conditionally independent. Consequently, the system cannot capture the knowledge that some mentions may be included in other mentions.

We overcome these limitations by explicitly encoding these information as COMP features. Composite features comprise of feature pairs indicating which of the two occurs first. These include:

- *HM1HM2:* Head word of mention 1 and head word of mention 2. This encodes what mention head word occurs first.
- *ET12 :* Ordered pair of mention entity type
- *ML12:* combination of mention levels
- *M1InM2 :* flag indicating whether M1 is included in M2. This feature captures mention dependencies
- *M2InM1:* flag indicating whether M2 is included in M1

7.5 Experiments

ERD-MEDLDA is a LDA-based framework that uses max-margin learning. Thus we need to verify if our MEDLDA formulation does better than topic modeling alone and max-margin methods alone. For this, we compare ERD-MEDLDA to both LDA and SVM. Comparison with SVM is straightforward as it is a supervised framework. However, as basic LDA is unsupervised, the discovered topics are not tied to any particular class. Thus, for a fair comparison, we use a labeled variant of LDA, the LLDA [41], as the baseline topic model system.

We use 80% of the instances for training and 20% for testing. The topic numbers and the penalty parameter of the cost function C are first determined for each of the models (wherever applicable) using the training data. Best parameters are determined for the three conditions: 1) BOW features alone *BOW*, 2) BOW plus SYN features (*PlusSYN*) and 3) BOW plus

SYN and COMP features (*PlusCOMP*). All systems achieved their overall best performance with PlusCOMP features (see Section 7.6.1 for a detailed analysis).

7.5.1 ERD-MEDLDA Setup

The number of topics for the LDA-based models are determined using the equation $2K_0 + K_1$ following Zhu, Ahmed and Xing [43] and $K_1 = 2K_0$. K_0 is the number of topics per class and K_1 is the number of topics shared by all relation types. The choice of topics is based on the intuition that the shared component K_1 should use all class labels to model common latent structure while non-overlapping components should model specific characteristics data from each class. The ratio of topics is based on the understanding that shared topics may be more than topics of each class. The specific numbers do not produce much variation in the final results. We experimented with the following number of topics: 20, 40, 70, 80, 90, 100, 110. BOW, PlusSYN, and PlusCOMP configurations obtain the best performance for 90 topics, 80 topics, and 70 topics respectively.

Since SVMs are employed in the ERD-MEDLDA implementation, we need to determine the penalty parameter of the cost function, C. We used 5 fold cross-validation to locate the parameter C. The best values for C are 25, 28, 30 respectively for BOW, PlusSYN and PlusCOMP configurations. We used a linear kernel as it is the most commonly used kernel for text classification tasks. Since ERD-MEDLDA is run by sampling, the result may be different each time. We ran it 5 times for each setting and took the average as the final results.

7.5.2 Baselines

We employ the same features and the same settings for all three models as much as possible.

LLDA

LLDA [41] is a variation of supervised LDA. While the original model was designed for recognizing credit attribution, it can be easily used for relation detection. We recreate the plate diagram from the original paper in Figure 7.3 and briefly explain the LLDA model for relation detection as follows.

The relation types (Λ) generate the topic distribution parameter θ with α. Each document d is represented by a tuple consisting of a list of word indices $w^d = (w_1, ..., w_{N_d})$

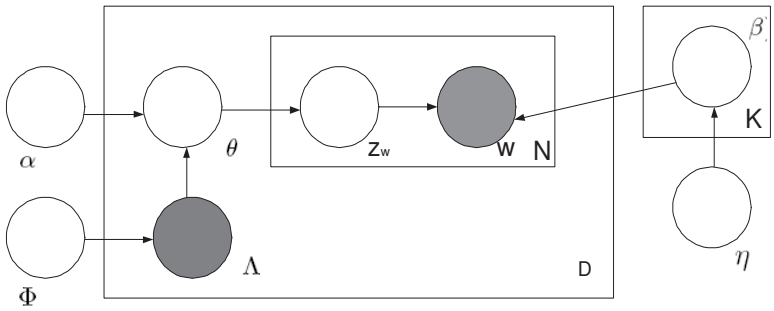

Figure 7.3: *Graphical model of LLDA*

and a list of binary relation type presence/absence indicators $\Lambda^{(d)} = (l_1, ..., l_K)$ where each $w_i \in \{1, ..., V\}$ and each $l_k \in \{0, 1\}$. N_d is the document length, V is the vocabulary size and K the total number of unique topic labels in the corpus.

Drawing the multinomial topic distributions over vocabulary β_k for each topic k, from a Dirichlet prior η is the same as that for traditional LDA. However, LLDA restricts θ^d to be defined only over the topics that correspond to its labels $\Lambda^{(d)}$. Since the word-topic assignments z_i are drawn from this distribution, this restriction ensures that all the topic assignments are limited to the document's label. We refer the reader to the orignial paper for further details.

The setting of topics for LLDA is similar to ERD-MEDLDA. As LLDA is also run by sampling, we ran it 5 times for each setting and took the average as the final results.

SVM

The SVM [48] baseline is a straightforward supervised system. We use the SVMlight implementation from [49]. For for our task, labels are relation types and the training vector comprises of the features discussed in Section 7.4. Binary (presence/absence) features are used to overcome sparcity problems. In SVMlight, a grid search tool is provided to locate the the best value for parameter C. The best C for all three feature conditions, BOW, PlusSYN and PlusCOMP, was found to be 1. All other settings are similar to those of ERD-MEDLDA, including the linear kernel.

7.5.3 Results

	Precision %	Recall %	Fmeasure%
SVM	53.2	35.2	40.3
LLDA	28.3	51.6	36.6
ERD-MEDLDA	**57.8**	**53.2**	**55.4**

Table 7.4: Overall performance of the three systems

Labels	SVM			LLDA			ERD-MEDLDA		
	Pre%	Rec%	F%	Pre%	Rec%	F%	Pre%	Rec%	F%
ART	30	8	14	1.5	33	3	**49**	**36**	**41**
GEN-AFF	**53**	**48**	**50**	3	32	6	40	39	40
ORG-AFF	55	35	43	**59**	58	**59**	53	**59**	56
PART-WHOLE	39	08	14	31	**82**	45	**44**	52	**48**
PER-SOC	50	17	25	7	**92**	13	**73**	76	**75**
PHYS	55	35	**43**	26	**47**	33	**56**	19	29
NO-REL	**90**	**95**	**93**	70	17	27	89	91	90

Table 7.5: Multi-class Classification Results

We present the results of the three systems built using PlusCOMP, as all systems achieved their best overall performance using these features. Table 7.4 reports the precision, recall and Fmeasure of the three systems averaged across all 7 categories (the best numbers for each metric are highlighted in **bold**). Amongst the baselines, SVM has better precision, while LLDA has better overall recall. Here we see that ERD-MEDLDA outperforms LLDA and SVM across all metrics. Specifically, there is a four percentage point improvement in precision, two percentage point improvement in recall, and 15 percentage point improvement in Fmeasure over the best performing baseline. This result indicates that our approach of combining topic model and max-margin learning is effective for relation detection.

Now, looking at the results for each individual relationship category (see Table 7.5; the best numbers for each category and metric are highlighted in **bold**) we see that the Fmeasure for ERD-MEDLDA is better than that for SVM for 4 out of the 6 ACE relation types; and

better than the Fmeasure obtained by LLDA for all relation types except ORG-AFF. Specifically, comparing with the best performing baseline, ERD-MEDLDA produces a Fmeasure improvement of 27 percentage points for ART, 3 percentage points for PART-WHOLE and 50 percentage points for PER-SOC. Also, for four of the six ACE relation types, ERD-MEDLDA achieves the best precision. Even in the cases where ERD-MEDLDA is not the best performer for a relation category, its performance is not very poor (unlike, for example, SVM for PART-WHOLE and LLDA for ART respectively).

Interestingly, the NO-REL category reveals a sharp contrast in the performance of SVM and LLDA. NO-REL is a difficult, catch-all category that is a mixture of data with diverse distributions. This is a category where maximum-margin learning is more effective than maximum-likelihood estimation. Notice that ERD-MEDLDA achieves performance close to SVM for this category. This is because, even though both LLDA and ERD-MEDLDA model hidden topics and then employ the discovered hidden topics to predict relation types, ERD-MEDLDA does joint inference of MLE and MME. This joint inference helps to improve the detection of NO-REL.

Labels	CD'01	AAP	AAPD	TSAAPD-0	TSAAPD-01	ERD-MEDLDA
ART	**51**	49	50	48	47	41
GEN-AFF	9	10	12	11	11	**40**
ORG-AFF	43	43	43	43	45	**56**
PART-WHOLE	30	28	29	30	28	**48**
PER-SOC	62	58	70	63	73	**75**
PHYS	32	**36**	29	33	33	29
Overall (Avg)	38	37	39	38	40	**48**

Table 7.6: F-measures for every kernel and MEDLDA

Finally, we also compare our system's results (using PlusCOMP features) with the results of previous research by Khayyamian, Mirroshandel and Abolhassani [50] on the same corpus. They use similar experimental settings: every pair of entities within a sentence is regarded to involve a negative relation instance unless it is annotated as positive in the corpus. A similar filter (they use a distance filter) is used to sift out unrelated negative instances. Their train/test ratio of data split is also the same as ours.

Khayyamian, Mirroshandel and Abolhassani [50] employ state-of-art kernel methods developed by Collins and Duffy [51] and only report Fmeasures over the six ACE relation types. For clarity, we reproduce their results in Table 7.6 and repeat ERD-MEDLDA Fmeasures from Table 7.5 in the last column. The last row (Overall) reports the macro-averages computed over all relation types for each system. Here we see that overall, ERD-MEDLDA outperforms all kernels. ERD-MEDLDA also performs better than the best kernel for four of the six relation types.

7.6 Analysis

We incorporated an exponential family distribution into ERD-MEDLDA model in order to make use of rich features. In this section we analyze if indeed ERD-MEDLDA effectively utilizes the variety of features listed in Section 7.4 in comparison with baseline methods.

Additionally, as supervised topic models are designed to infer topic distributions indicative of the class, we inspect the topics learned by ERD-MEDLDA to see if the inclusion of supervision has created topics biased towards relation types.

7.6.1 Feature Incorporation

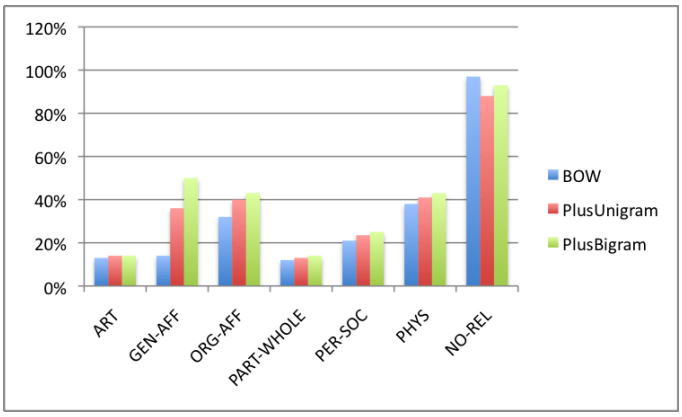

Figure 7.4: SVM Fmeausres for 3 feature conditions

As mentioned previously, all three systems achieved their overall best performance with PlusCOMP features. Here, we analyze if informative features are consistently useful and if the systems can harness the informative features consistently across all relation types. Figures

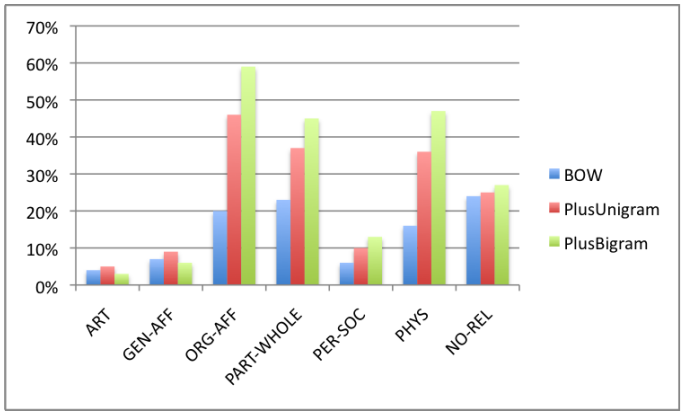

Figure 7.5: LLDA Fmeausres for 3 feature conditions

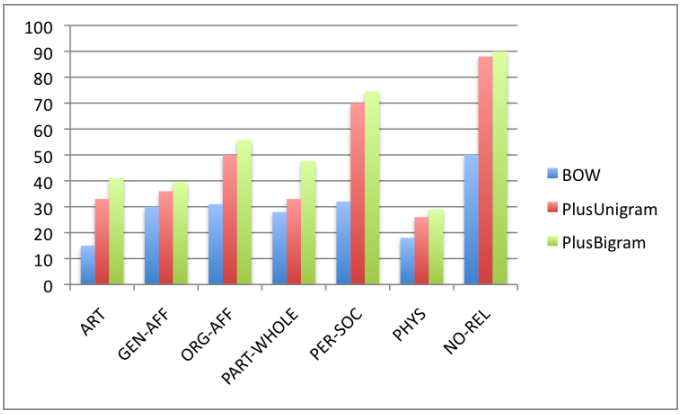

Figure 7.6: MEDLDA Fmeausres for 3 feature conditions

7.4, 7.5 and 7.6 illustrate the F-measures for SVM, LLDA and ERD-MEDLDA respectively for the three conditions: BOW, PlusSYN and PlusCOMP.

Let us first look at the best systems (based on Fmeasure) for each of the six ACE relation types in Table 7.5, and look at what feature set produces the best result for that system and relation. ERD-MEDLDA is the best performer for ART, PART-WHOLE and PER-SOC in Table 7.5. Figure 7.6 reveals that ERD-MEDLDA's best performance for these relation types are obtained using PlusCOMP features. Similarly SVM obtains the best Fmeasure for GEN-AFF and PHYS relations and Figure 7.4 shows that SVM achieves its best performance for these categories using PlusCOMP. We also see a similar trend with LLDA and the ORG-AFF relation

type. These results corroborate intuition from previous research that informative features are important for relation type recognition. The only exception to this is the performance of SVM for NO-REL. This is not surprising, as the features we use are focused on determining true relation types and NO-REL is a mixture of all cases (and features) where relations do not exist.

Further analysis of the figures reveal that even though there is a general trend towards better performance with addition of more informative features, not all systems show consistent improvements across all relation types with the addition of composite features. That is, some systems get degraded performance due to feature addition. For example, in Figure 7.4, we see that the SVM with PlusCOMP features is outperformed by SVM with PlusSYN for ART and SVM with BOW for NO-REL. The gains from features are also inconsistent in the case of LLDA (Figure 7.5). While the LLDA system with PlusSYN features always improves over the one using BOW, the performance drops considerably when using PlusCOMP features for ART and GEN-AFF. On the other hand, ERD-MEDLDA (see Figure 7.6) shows more consistent improvement for all relation types with the addition of more complex features. Also, the gains are more substantial. This is encouraging and opens up avenues for further exploration.

7.6.2 Topic Discovery

Our goal in employing the supervised framework was to guide the topic discovery to topics useful for relation detection. In this section, we examine the topics discovered for each relation type and discuss the effects of varying the number of topics in our model.

Figure 7.7 illustrates the topic distribution in ERD-MEDLDA for the different relation types. The distributions are computed by averaging the expected latent representation of documents in each class. For all 6 ACE relation categories, we observe a sharp, sparse and fast-decaying distribution over topics, indicating an affinity of the relation types for certain indicative topics. The NO-REL class, on the other hand, does not show this characteristic. Here, we see an almost uniform distribution over all topics. This is not surprising, as the NO-REL is a catch-all category comprising of various topics that do not correspond to the ACE relation types.

Notice that Topics 10 and 0 are prominent across some relation types. Specifically, Topic 10 is prominent for all 6 ACE relation categories, and Topic 0 is prominent for GEN-AFF and PHYS categories. When we inspected the top features belonging to Topic 10, we discovered

that it covered person-related and pronominal features such as $I-I$, $I-you$ or $I-my$, he, $he-he$, $they-he$, $them-I$ and $that$. Intuitively, these features could indicate any one of the relation types. Topic 10 is in fact an indicator of a *presence* of an ACE relation – which also explains why Topic 10 is not seen in the NO-REL category.

Most relation types have distributions over additional topics that help distinguishing them. For example, in addition to Topic 10, ART has a substantial percentage of distribution over Topics 14, 15 and 17. Similarly, ORG-AFF has noticeable distributions over Topics 1 and 11. The only exception is the PHYS category. PHYS has prominent distributions only over Topics 0 and 10, both of which are not unique to it alone. Not surprisingly, this affects the recognition of this category, as evidenced by the relatively low Fmeasures in Table **??**. In contrast, PER-SOC has a strong component of Topic 7, which is not seen in any other category. Consequently, it is well distinguished, as evidenced by the results in Table **??**

We observed that increasing the number of topics helps the system to obtain more distinct distributions for the relation types. Figure 7.8 illustrates the topic distribution when 110 topics are used. However, if the number of topics is too large, it can lead to overlapping topic distributions again. Optimal topic numbers for ERD is an avenue for future exploration.

We also inspected the topics discovered by LLDA and basic LDA. LLDA assumes that the topic discovered is the relation type. Here too, supervision helps with tying the topics to the relation types. For example, $family$ and $wife$ were amongst the top features for PER-SOC, and $weapons$ was a top feature for ART. Further, similar to ERD-MEDLDA, LLDA also showed broad topic distribution for NO-REL.

Finally, topics discovered (and consequently, the features for those topics) by the basic LDA model were not clearly interpretable as indicative of the relation types. This is not surprising, as basic LDA does not incorporate supervision.

7.7 Related Work

As discussed in above sections, our work targets at relation detections. However, the exploration of model adaptations and improvements on relation detections is also our significant goals. In this section, we firstly have a quick browse on what progress is being made in relation detection and then we will have a short review on what LDA is and what applications LDA and its variations have been made.

7.7.1 Diverse Approches to Relation Detection

Previous research has explored various methods for relationship detection and mining. Kernel methods have been popularly used for ERD. The main advantage of kernel methods is the ability of exploiting a huge amount of features features without an explicit feature representation. This can be done by computing a kernel function between a pair of linguistic objects, where such function is a kind of similarity measure satisfying certain properties. One simple kernel function used in NLP is the sequence kernel [52], where the objects are strings of characters and the kernel function computes the number of common subsequences of characters in the two strings. Some more involved examples are dependency tree kernels [53], shortest dependency path kernels [54]. Collins and Duffy [51] developed the classical tree kernels to encode grammatical derivations.

ERD is one of the earliest NLP fields in which intensive explorations are made on various kernel functions. Zelenko, D. and Aone, C. and Richardella, A. [55] has exploited some similarity measures over diverse features. More recently, convolution tree kernels [56; 57] context-sensitive convolution tree kernels [58] and dynamic syntax tree kernels [59]. Though more complex, the basic idea is following the same fashion. All tree kernels aims at counting the number of subtree shared by two input trees. If the number of shared subtree is close to each other, they should be counted as similar.

Besides using independent kernel functions, the combinations among different kernel functions have also been explored. Nguyen, Moschitti and Riccardi [60] propose to combine constituent and dependency trees and sequential structures with kernel methods. To fully exploit the potential of dependency tree, they also applied the partial tree kernel proposed by [61], which is a general convolution tree kernel adaptable for dependency structures and investigated the incorporation of dependency structure into rich sequence kernels like word sequence kernels [62].

Rich kernel methods enable us to realize how important the structural information is for ERDs. In the process of development of our ERD-MEDLDA model, we take into full considerations how to represent such information. instead of computing distances between subtrees, we sample topics based on their distributions. The sampling is not only on the (mini) document level, but also on the word level or on the syntactic or semantic level. Our method focuses on addressing the underlying semantics more directly than typical kernel-based methods. Furthermore, the marriage of MLE-based supervised LDA and MME-based

SVM make it possible for our model to directly employing rich kernel functions. The use of kernel functions in ERD MEDLDA is beyond the scope of our present work though it is one of our future efforts.

Chan and Roth [63] employ constraints using an integer linear programming (ILP) framework. Using this, they apply rich linguistic and knowledge-based constraints based on coreference annotations, a hierarchy of relations, syntacto-semantic structure, and knowledge from Wikipedia. In our work, we focus on capturing the latent semantics of the text between the NEs.

Besides employing kernel functions, extraction of features to enhance ERD is always an unignorable approach. A variety of features have been explored for ERD in previous research [47; 64; 46; 65; 66]. Syntactic features such as POS tags and dependency path between entities; semantic features such as Word-Net relations, semantic parse trees and types of NEs; and structural features such as which entity came first in the sentence have been found useful for ERD. Roth and Yih [67] applied a probabilistic approach to solve the problems of named entity and relation extraction with the incorporation of all these features. Kambhatla [66] employed maximum entropy models with diverse features including words, entity and mention types and the number of words (if any) separating the two entities.

We too observe the utility of informative features for this task. However, exploration of the feature space is not the main focus of this work. Rather, our focus is on whether the models are capable of incorporating rich features. A fuller exploration of rich heterogeneous features will be done in our future work.

A closely related task is that of relation mining and discovery, where unsupervised, semi-supervised approaches have been effectively employed [39; 68; 69]. For example, Hasegawa et al. [39] use clustering and entity type information, while Mintz et al. [68] employ distant supervision. Our ERD task is different from these as we focus on classifying the relation types into predefined relation types in the ACE05 corpus.

7.7.2 Topic Models and Natural Language Processing

From previous research on relation detection, it seems that few relation detection work is based on LDA or its variations. However, in the general natural language processing, LDA or called topic models have been extensively employed.

Since the birth of LDA, it has been an active model in machine learning and related fields,

such as image processing, bioinformatics and language processing. Many researchers have explored extensions to the original LDA from Blei et al. [40], such as correlated topic models [70], LLDA [41], sLda [42], discLDA [71]. In NLP, topic models have been employed for quite a few tasks, such as review mining [72; 73], perspective analysis [74], image annotation [75], key phrase extractions [76] and genomic profiling [77].

But so far, no research has been done to use LDA as a prime model for relation detection. The work on relation discovery from Hachey [78] is the only we have found to use LDA in their model. Yet, they only use LDA as a module in his system to reduce feature dimensions only. But it is worth mentioning that they do realize the advantage of LDA over other models such as latent semantic analysis (LSA) [79] or its variation probabilistic LSA (pLSA) which is more close to LDA. They point out that pLSA, as a generative probabilistic version of LSA [80], model each word in a document as a sample from a mixture model, but does not provide a probabilistic model at the document level. LDA addresses this by representing documents as a random mixtures over latent topics [81]. Besides having a clear probabilistic interpretation, LDA provides intuitive graphical representations.

In this work we adapt the MEDLDA topic model to incorporate rich features for the task of relation extraction. We not only develop relation detection into a fully probabilistic model which make use of MLE for maximizing the posterior distributions, but also, we also make use of MME meanwhile for catching the boundary cases of relation types by employing Zhu and et al [82]'s MEDLDA.

7.8 Conclusion and Future Work

In this work, I have presented ERD-MEDLDA, a system for detecting entity relations based on topic models. The approach was motivated by the idea that latent semantics of text, as discovered by LDA-based models, are useful for relation detection. For this, ERD was presented as a topic modeling task. To the best of my knowledge, this is the Þrst work to make full use of topic models for relation detection. MEDLDA and mixed membership models were adapted to the relation detection task. SpeciÞcally, the optimization problem was modified an exponential family distribution was incorporated for each feature. The resulting ERD-MEDLDA model has the advantages of both max margin and maximum likelihood methods and is also able to beneÞt from rich features.

Our experiments show that ERD-MEDLDA achieves better overall performance than SVM-based and LLDA-based approaches across all metrics. Comparing with previous work from [50], ERD-MEDLDA was shown to have better overall performance than state-of-the-art kernels.

We also experimented with different features and the effectiveness of ERD-MEDLDA in harnessing these features as compared to baseline methods. An analysis shows that ERD-MEDLDA is able to effectively and consistently incorporate informative features. Examination of the topic distribution obtained by this system shows that supervision indeed helps the model to learn topics biased towards the relation types. As a model that incorporates maximum-likelihood, maximum-margin and mixed membership learning, ERD-MEDLDA has the potential of incorporating rich kernel functions or conditional topic random Þelds (CTRF) [82]. These are some of the promising directions for our future exploration.

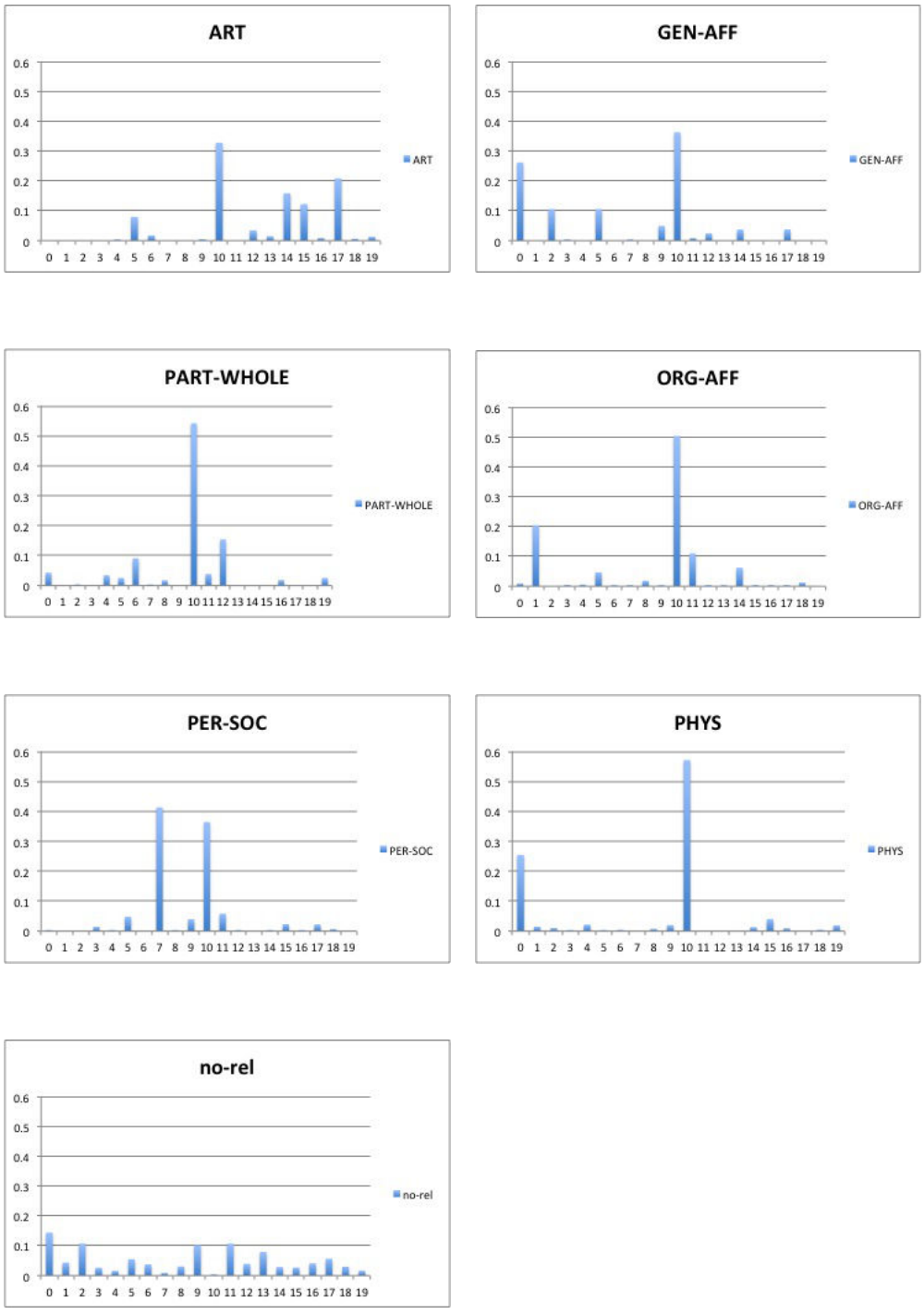

Figure 7.7: Topic distribution for all relation types and NO-REL with 20 topics

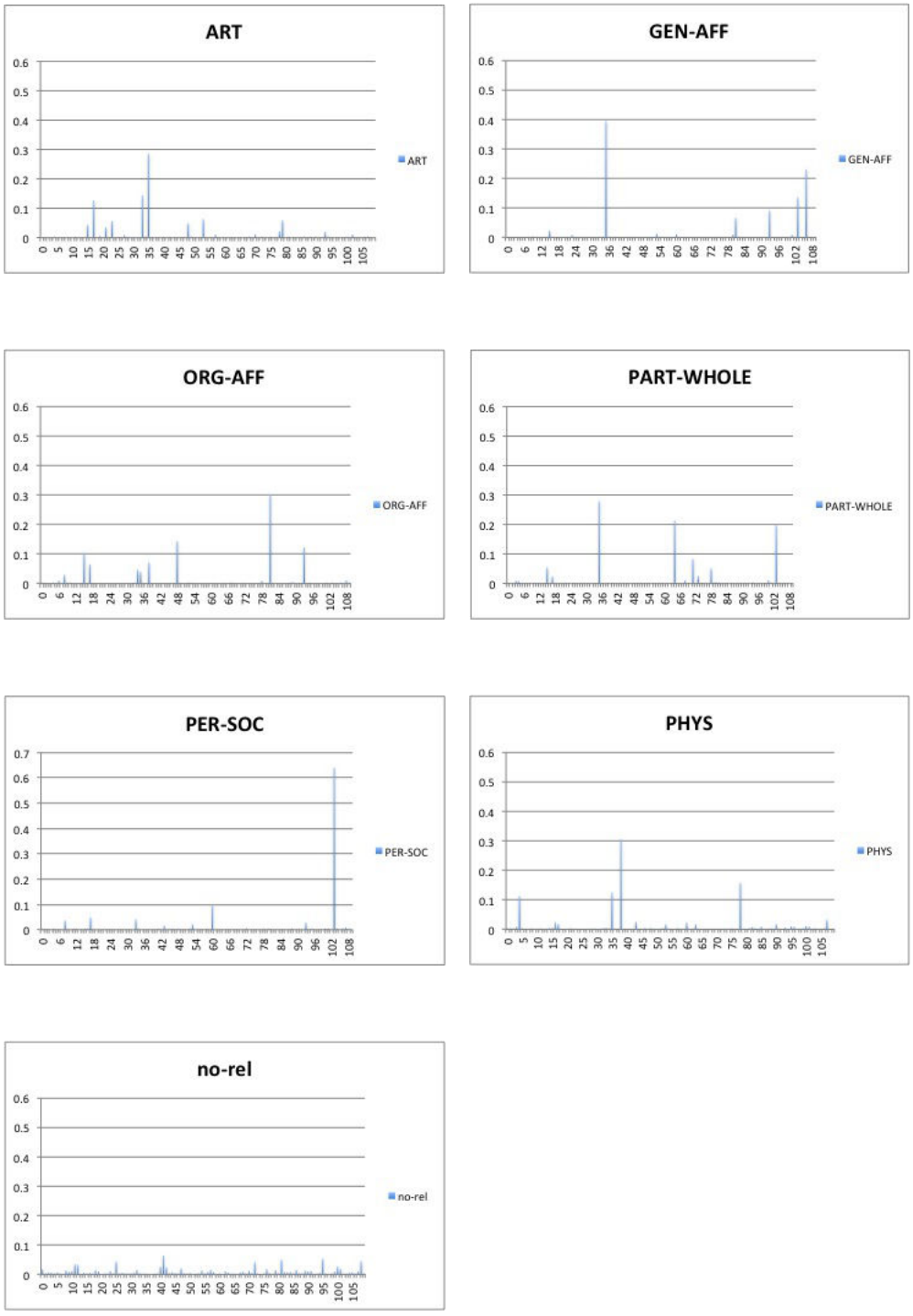

Figure 7.8: Topic distribution for all relation types and NO-REL with 110 topics

Chapter 8

Conclusion and Future Work

Although each chapter on the model development and evaluations has given a conclusion and future work, it cannot be counted as complete without a final chapter on conclusion and future work. This is the motivation to write such a short chapter.

Starting from coreference resolution system, there are various existing coreference resolution systems available, from mention-based, entity-based to ranking models or multi-sieved models. All these models have made successful results. However, none of these models handle coreference resolution incrementally as we human beings do. Hidden Markov models are themselves incremental time-series models. They have been successfully applied in speech-recognition, parsing an tagging. Hierarchical HMMs (HHMMs), because they can reflect the nature of rich levels in human languages, are also favored by NLP researchers. Schuler and et al [83]. This is the main motivation for me to develop a coreference resolution system with FHMMs, a variation of HHMMs. With the robust nature of RHMM, the coreference resolution system bore out impressive results.

After coreference resolution is expanded to include givenness hierarchy and centering theory, we train and test the new model on I2B2 medical corpus. High performance is obtained. This shows that the feature additions contribute better accuracy. It also shows that FHMM-based coreference resolution system has a good mechanism to absorb new features. It lays a good foundation for the future enhancements and adjustments.

Successful conversion of givenness hierarchy and centering theory open a new road to unify and integrate statistical models and linguistic theories. It is known that machine learning models often display strong mathematical soundness. Nonetheless, in the process of

real-world applications, incorrect or inappropriate integrations of domain theories lower the capacity of the model. Therefore, how to combine machine learning models and linguistics theories play a decisive role in improving the specific NLP subtasks.

Due to the time issue, I didn't test the system with only adding GH or with only adding centering theory. It may be interesting to see how large the role for each of them. This is what I am working now. The results will be reported in another paper soon.

In the relation detection part, similar challenges exist. Latent Dirichlte Allocations (LDAs) are themselves good at clustering topics. But topics learned without supervision often lack the interpretation power though LDAs do a good job in the process of automatic topic discoveries. These topics may be good enough for people who only want to browse at today's news to get a fast overview at news stories or even good enough for literary critical writers to write a imaginary comment on who is the writers of an ancient playwrights. However, if our goal is to detect the relationship between entities, these topics may be too random to be so helpful. Hence, we need to select a more supervised model with side information as annotated features. That is the main reason that we gave up a LDA-based relation detection model. Meanwhile, change of document concept from a whole article to a fragment around two related mentions is also an innovation of making the supervised LDA suitable for the task of relation detection. In the short document, words between the two mentions and rich features extracted from these words have good discriminative distributions. With good designs of the supervised LDA, these features can be fully made use of to be trained as powerful models. Besides the topic model itself, the combination of maximum-likelihood estimation and maximum-entropy estimation is another key to enable the ERD system has a good predictive power.

But the two systems are beyond perfect. For FHMM-based coreference resolution system, the speed problem is still a problem with more features added. Namely, once more features are added to the new model, high computation complexity will reduce the processing speed. Consequently, how to do more smart beam search is a bottleneck to build a real-world coreference resolution system. Once the speed can be greatly enhanced, we can make a more accurate system. In fact, there is much room for the present CR system to integrate more features if the speed is in the normal range. But the fact is that with addition of present features, the testing takes more than one week. Thus, in the future work on FHMM-based

coreference resolution system, speed improvement is the first work to do. Besides, expansion of the pronoun resolution system to a system which can resolve both pronouns and nominal mentions is another step of future work.

For relation detection work, the final measure, as we can see, is higher than the state-of-art kernel based ERD system. However, the absolute F-measure is still lower than other NLP systems. Thus, much work needs to do so that relation detection can be more effectively used for other tasks. In analysis, we have understood partial reason why there are many errors in ERD. One reason is that the order may be critical for some relations, but the MEDLDA doesn't care about order since MEDLDA, like other LDA models, assume that features are independent of each other. In order to make up this defect, we do use compound features which are formed by combining two sequential features as one feature. However, this still cannot fully help distinguish relations with different orders. In addition, heterogeneous features used enhance the detection results. But like RHMM-based CR system, the processing speed has been quite slow now. That indicates that the feature state space is so large that the learning and inference become complex. Thus, future feature additions may be hard though still possible.

An ideal alternative to this is to transform MEDLDA to conditional topic random fields (CTRF) **??**. As is known, generative models specifies the joint likelihood of all variables so that addition of new features is not a trivial thing. In contrast, CTRF models conditional likelihood which can incorporate arbitrary non-local features. At the same time, CTRF, like conditional random fields, can model the dependence between features since CTRF directly incorporates the Markov dependency between the topic assignments of neighboring words based on the general linear model principle. Lastly, CTRF, in essence is an extension of MEDLDA with addition of Markov dependencies and the adaptation of joint likelihood to conditional one. It doesn't change the nature that integrate MLE and MME. Therefore, update from MEDLDA to CTRF should be a right track we will make effort to improve relation detection model.

References

[1] JK Gundel, N Hedberg, and R Zacharski. Cognitive status and the form of referring expressions in discourse. *Language*, pages 274–307, 1993.

[2] SC Levinson. *Presumptive meanings: The theory of generalized conversational implicature*. MIT Press, 2000.

[3] H.S. Horn and R.H. Mac Arthur. Competition among fugitive species in a harlequin environment. *Ecology*, pages 749–752, 1972.

[4] D Jurafsky, JH Martin, A Kehler, K Vander Linden, and N Ward. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. MIT Press, 2000.

[5] K McCoy and M Strube. Generating anaphoric expressions: Pronoun or definite description. page 63ï£¡71, 1999.

[6] ACE. Automatic Content Extraction. *http://www.ldc.upenn.edu/Projects/ACE/*, 2000-2005.

[7] X. Yang, J. Su, G. Zhou, and C.L. Tan. Improving pronoun resolution by incorporating coreferential information of candidates. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 127. Association for Computational Linguistics, 2004.

[8] L. Qiu, M.Y. Kan, and T.S. Chua. A public reference implementation of the rap anaphora resolution algorithm. *Arxiv preprint cs/0406031*, 2004.

[9] A. McCallum and B. Wellner. Toward conditional models of identity uncertainty with application to proper noun coreference. In *IJCAI Workshop on Information Integration on the Web*. Citeseer, 2003.

[10] P. Denis and J. Baldridge. A ranking approach to pronoun resolution. In *Proc. IJCAI*, 2007.

[11] A. Haghighi and D. Klein. Unsupervised coreference resolution in a nonparametric bayesian model. In *ANNUAL MEETING-ASSOCIATION FOR COMPUTATIONAL LINGUISTICS*, volume 45, page 848, 2007.

[12] V. Ng. Unsupervised models for coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 640–649. Association for Computational Linguistics, 2008.

[13] Eugene Charniak and Micha Elsner. Em works for pronoun anaphora resolution. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL-09)*, Athens, Greece, 2009.

[14] David McClosky, Eugene Charniak, and Mark Johnson. BLLIP North American News Text, Complete. *Linguistic Data Consortium. LDC2008T13*, 2008.

[15] A. Haghighi and D. Klein. Coreference resolution in a modular, entity-centered model. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 385–393. Association for Computational Linguistics, 2010.

[16] X Luo. On coreference resolution performance metrics. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 25–32. Association for Computational Linguistics Morristown, NJ, USA, 2005.

[17] T.S. Morton. Coreference for NLP applications. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 173–180. Association for Computational Linguistics, 2000.

[18] Zoubin Ghahramani and Michael I. Jordan. Factorial hidden markov models. *Machine Learning*, 29:1–31, 1997.

[19] Kevin Duh. Jointly labeling multiple sequences: a factorial HMM approach. In *ACL '05: Proceedings of the ACL Student Research Workshop*, pages 19–24, Ann Arbor, Michigan, 2005.

[20] S Bergsma. Automatic acquisition of gender information for anaphora resolution. In *Proceedings of the 18th Conference of the Canadian Society for Computational Intelligence (Canadian AI 2005)*, page 342ï£¡353. Springer, 2005.

[21] Noam Chomsky. *Lectures on government and binding*. Foris, Dordercht, 1981.

[22] A. Haghighi and D. Klein. Simple coreference resolution with rich syntactic and semantic features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pages 1152–1161. Association for Computational Linguistics, 2009.

[23] L. Hasler, C. Orasan, and K. Naumann. NPs for events: Experiments in coreference annotation. In *Proceedings of the 5th edition of the International Conference on Language Resources and Evaluation (LREC2006)*, pages 1167–1172. Citeseer, 2006.

[24] Dan Klein and Christopher D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430, Sapporo, Japan, 2003.

[25] C Sutton and A McCallum. 1 an introduction to conditional random fields for relational learning. *Introduction to statistical relational learning*, page 93, 2007.

[26] S.J. Russell, P. Norvig, J.F. Canny, J.M. Malik, and D.D. Edwards. *Artificial intelligence: a modern approach*, volume 74. Prentice hall Englewood Cliffs, NJ, 1995.

[27] J.K. Gundel and T. Fretheim. Topic and focus. *The handbook of pragmatics*, pages 175–196, 2004.

[28] D.I. Beaver. The optimization of discourse anaphora. *Linguistics and Philosophy*, 27(1):3–56, 2004.

[29] J. Grimshaw and V. Samek-Lodovici. Optimal subjects and subject universals. *Is the best good enough? Optimality and competition in syntax*, pages 193–219, 1998.

[30] I. Heim. On the projection problem for presuppositions. *Formal Semantics*, pages 249–260, 1983.

[31] M. Kameyama. Intrasentential centering: A case study. *Arxiv preprint cmp-lg/9707005*, 1997.

[32] D. Byron and W. Gegg-Harrison. Evaluating optimality theory for pronoun resolution algorithm specification. *Beaver*, page 1, 2004.

[33] M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52. Association for Computational Linguistics, 1995.

[34] A Bagga and B Baldwin. Algorithms for scoring coreference chains. *Recall*, 5(1):2, 2010.

[35] X. Luo. On coreference resolution performance metrics. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 25–32. Association for Computational Linguistics, 2005.

[36] M. Recasens and E. Hovy. Blanc: Implementing the rand index for coreference evaluation. *Natural Language Engineering*, 1(1):1–26, 2011.

[37] S. Pradhan, L. Ramshaw, M. Marcus, M. Palmer, R. Weischedel, and N. Xue. Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. *CoNLL 2011*, page 1, 2011.

[38] P Denis and J Baldridge. Joint determination of anaphoricity and coreference resolution using integer programming. In *Proceedings of NAACL HLT*, pages 236–243, 2007.

[39] T Hasegawa, S Sekine, and Ralph Grishman. Discovering relations among named entities from large corpora. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 415–422, Barcelona, Spain, July 2004.

[40] D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[41] D. Ramage, D. Hall, R. Nallapati, and C.D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *In 2009 EMNLP*, pages 248–256. Association for Computational Linguistics, 2009.

[42] D.M. Blei and J. McAuliffe. Supervised topic models. *Advances in Neural Information Processing Systems*, 20:121–128, 2008.

[43] J. Zhu, A. Ahmed, and E.P. Xing. MedLDA: maximum margin supervised topic models for regression and classification. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1257–1264. ACM, 2009.

[44] H. Shan, A. Banerjee, and N.C. Oza. Discriminative Mixed-membership Models. In *2009 Ninth IEEE International Conference on Data Mining*, pages 466–475. IEEE, 2009.

[45] T.P. Minka. A comparison of numerical optimizers for logistic regression. *Unpublished draft*, 2003.

[46] J. Jiang and C.X. Zhai. A systematic exploration of the feature space for relation extraction. In *proceedings of NAACL/HLT*, pages 113–120, 2007.

[47] G Zhou, S. Jian, Z. Jie, and Z. Min. Exploring various knowledge in relation extraction. In *In 43rd ACL*, pages 427–434. Association for Computational Linguistics, 2005.

[48] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[49] T. Joachims et al. Making large-scale svm learning practical. advances in kernel methods-support vector learning, b. sch*ö*lkopf and c. burges and a. smola, 1999.

[50] M. Khayyamian, S.A. Mirroshandel, and H. Abolhassani. Syntactic tree-based relation extraction using a generalization of Collins and Duffy convolution tree kernel. In *HLT/NAACL,Companion Volume: Student Research Workshop and Doctoral Consortium*, pages 66–71. Association for Computational Linguistics, 2009.

[51] M. Collins and N. Duffy. Convolution kernels for natural language. *Advances in neural information processing systems*, 1:625–632, 2002.

[52] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins. Text classification using string kernels. *The Journal of Machine Learning Research*, 2:419–444, 2002.

[53] A. Culotta and J. Sorensen. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 423. Association for Computational Linguistics, 2004.

[54] R.C. Bunescu and R.J. Mooney. A shortest path dependency kernel for relation extraction. In *In HLT & EMNLP*, pages 724–731. Association for Computational Linguistics, 2005.

[55] D. Zelenko, C. Aone, and A. Richardella. Kernel methods for relation extraction. *The Journal of Machine Learning Research*, 3:1083–1106, 2003.

[56] S. Zhao and R. Grishman. Extracting relations with integrated information using kernel methods. In *In 43rd ACL*, page 426. Association for Computational Linguistics, 2005.

[57] M. Zhang, J. Zhang, J. Su, and G. Zhou. A composite kernel to extract relations between entities with both flat and structured features. In *In 21st ICCL & 44th ACL*, pages 825–832. Association for Computational Linguistics, 2006.

[58] G Zhou, M. Zhang, D.H. Ji, and Q Zhu. Tree kernel-based relation extraction with context-sensitive structured parse tree information. In *EMNLP/CoNLL-2007*, pages 728–736. Citeseer, 2007.

[59] L. Qian, G. Zhou, F. Kong, Q. Zhu, and P. Qian. Exploiting constituent dependencies for tree kernel-based semantic relation extraction. In *In 22nd ACL*, pages 697–704. Association for Computational Linguistics, 2008.

[60] T.V.T. Nguyen, A. Moschitti, and G. Riccardi. Convolution kernels on constituent, dependency and sequential structures for relation extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pages 1378–1387. Association for Computational Linguistics, 2009.

[61] A. Moschitti. Efficient convolution kernels for dependency and constituent syntactic trees. *Machine Learning: ECML 2006*, pages 318–329, 2006.

[62] N. Cancedda, E. Gaussier, C. Goutte, and J.M. Renders. Word sequence kernels. *The Journal of Machine Learning Research*, 3:1059–1082, 2003.

[63] Y. Chan and D. Roth. Exploiting syntactico-semantic structures for relation extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Portland, Oregon, 2011.

[64] G.D. Zhou, M. Zhang, D.H. Ji, and Q.M. Zhu. Hierarchical learning strategy in semantic relation extraction. *Information Processing & Management*, 44(3):1008–1021, 2008.

[65] S. Miller, H. Fox, L. Ramshaw, and R. Weischedel. A novel use of statistical parsing to extract information from text. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 226–233. Morgan Kaufmann Publishers Inc., 2000.

[66] N. Kambhatla. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 22. Association for Computational Linguistics, 2004.

[67] D. Roth and W. Yih. Probabilistic reasoning for entity & relation recognition. In *In 19th ACL*, page 7. Association for Computational Linguistics, 2002.

[68] M Mintz, S Bills, R Snow, and D Jurafsky. Distant supervision for relation extraction without labeled data. In *In 47th ACL & 4th AFNLP*, pages 1003–1011, Suntec, Singapore, August 2009. Association for Computational Linguistics.

[69] J. Jiang. Multi-task transfer learning for weakly-supervised relation extraction. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1012–1020. Association for Computational Linguistics, 2009.

[70] D.M. Blei and M.I. Jordan. Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–144, 2006.

[71] S. Lacoste-Julien, F. Sha, and M.I. Jordan. Disclda: Discriminative learning for dimensionality reduction and classification. *Advances in Neural Information Processing Systems*, 21, 2008.

[72] I. Titov and R. McDonald. Modeling online reviews with multi-grain topic models. In *Proceeding of the 17th international conference on World Wide Web*, pages 111–120. ACM, 2008.

[73] W.X. Zhao, J. Jiang, H. Yan, and X. Li. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 56–65. Association for Computational Linguistics, 2010.

[74] W.H. Lin, E. Xing, and A. Hauptmann. A joint topic and perspective model for ideological discourse. *Machine Learning and Knowledge Discovery in Databases*, pages 17–32, 2008.

[75] C. Wang, D. Blei, and F.F. Li. Simultaneous image classification and annotation. *Conference on computer vision and pattern recognition*, 2009.

[76] X. Zhao, J. Jiang, J. He, Y. Song, P. Achananuparp, E.P. LIM, and X. Li. Topical keyphrase extraction from twitter. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 2011.

[77] P. Flaherty, G. Giaever, J. Kumm, M.I. Jordan, and A.P. Arkin. A latent variable model for chemogenomic profiling. *Bioinformatics*, 21(15):3286, 2005.

[78] B. Hachey. Comparison of similarity models for the relation discovery task. In *COLING & ACL 2006*, page 25, 2006.

[79] S.T. Dumais. Latent semantic analysis. *Annual Review of Information Science and Technology*, 38(1):188–230, 2004.

[80] T. Hofmann. Probabilistic latent semantic analysis. In *Proc. of Uncertainty in Artificial Intelligence, UAI'99*, page 21. Citeseer, 1999.

[81] D.M. Blei, A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[82] J. Zhu and E.P. Xing. Conditional Topic Random Fields. In *International Conference on Machine Learning (To appear)*. Citeseer, 2010.

[83] W. Schuler, S. AbdelRahman, T. Miller, and L. Schwartz. Broad-coverage parsing using human-like memory constraints. *Computational Linguistics*, 36(1):1–30, 2010.