Discriminative Coherence Evaluation with Discourse Role Matrix

Guifu Liu

4th Year Project Report
Artificial Intelligence
School of Informatics
University of Edinburgh

2024

Abstract

Coherence tells us if different parts of a text are connected logically and meaningfully. Thus, developing system that automatically evaluates coherence is crucial to improve the performances of many NLP tasks, such as text summarization and generation.

In this study, we investigate **Discourse Role Matrix**, the first discriminative coherence evaluation method that combines entity-based modeling with discourse relations. We apply this linguistically-rich framework on **Shuffle Test** to distinguish between a *coherent* original text and an *incoherent*, random permutation of that text.

Motivated by empirical evidence in designed experiments and other coherence frameworks, we extend upon Lin et al. [26] and propose additional sources of linguistic knowledge to improve model performance. These sources of knowledge include: (1) granularity of discourse relation label (2) intra-sentential and inter-sentential distinction for discourse relation (3) types of discourse entities and (4) text genre. In addition, we also investigate if Convolutional Neural Network may be used to extract features with longer discourse role transitions.

We explore and justify their effectiveness in coherence analyses and make suggestions for future design choices to improve entity and discourse relation based coherence model.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Guifu Liu*)

Acknowledgements

I would like to thank my project supervisor, Dr. Adam Lopez, for guiding me with this project with his valuable advice.

I would also like to thank Prof. Bonnie Webber, for her time and efforts on helping me understand various aspects of PDTB.

I would like to thank my mother for her lifelong support of my education.

Thank you to everyone who has supported and encouraged me in my efforts on this project.

Table of Contents

Chapter 1

Introduction

1.1 Motivations

A good writing requires a logical structure to organize and present an author's thoughts. Thereby communicate it clearly to the reader. A well-structured paragraph distinguishes itself from a random sequence of sentences and clauses, because there exists relations between them. These relations make two sentences or clauses **coherent**. Consider this example from Hobbs [15],

(1) Jane took a train from Paris to Istanbul. She likes spinach.

You may be wondering how Jane's love for spinach relates to her travel. These two sentences appear incoherent together because we cannot see a relation between them. By contrast, in this example:

(2) Jane took a train from Paris to Istanbul. She had to attend a conference.

The second sentence gives a REASON for Jane's travel in the first sentence. Relations like this hold text spans together with rhetorical purpose and make them coherent.

Discourse, or a group of coherent sentences, is studied to uncover linguistics phenomena that are beyond a single sentence. **Discourse analysis** has emerged as an important NLP task for many document-level application that involves understanding or generating text. Some downstream tasks include text summarization, essay scoring, question answering, readability assessment and machine translation [19]. Therefore, modeling coherence is critical for NLP systems to pay attention to logical and semantic organization of multi-sentential inputs, and as a result, to improve performance in these downstream tasks [39].

The main focus of this project is a **coherence model** that evaluates text coherence automatically. The model takes a document as an input and returns a score to assess how coherent the given document is. One fundamental decision which forms the foundation of a coherence model is whether it is discriminative or generative. Discriminative models use contrastive learning to distinguish coherent instances from incoherent ones. By contrast, generative models maximizes the likelihood of coherent training text.

The coherence model we consider is **discriminative**. In particular, the model should be able to distinguish an original text from its incoherent renderings by permutating its sentences. Discriminative model sees incoherent texts and incorporate coherence into the the model objective. This is less plausible for generative models that do not see incoherent instances.

1.2 Goal and Research Question

The goal of this study is to investigate components of **Discourse Role Matrix** [26], how other sources of linguistic knowledge can be applied to this model, and their effectiveness on the discriminative coherence evaluation.

We chose this model because to our knowledge, it is the first model that uses information of discourse relations to evaluate coherence. In addition, the model also exploits entity patterns and is compatible with existing entity-based coherence model [2]. Its linguistic richness warrants in-depth analysis on the model's performance in coherence analyses, under various linguistics context. Therefore, the primary research question is:

> What aspects of Discourse Role Matrix help a model to evaluate text coherence?

In the following section, we will break this down further into minor objectives and their corresponding achievements. We summarize the report structure in Figure 1.1.

1.3 Objectives and Achievements

We first reproduce and analyze Lin et al.'s Discourse Role Matrix to understand the strength and weakness of this model on discriminative coherence evaluation. We have successfully implemented (3.1) and replicated the model with comparable test accuracy (5.1). We have found that **discourse units, entities, roles and feature extraction** are model components that we can modify and improve. We do so in the following objectives:

(1) Interpret what type of relation transition that the model prefers for a text to be seen as coherent. This provides analysis on what aspects of input text are best utilized in discriminative evaluation task. We find that the model favors transition of same relation (5.3). In addition, higher relation density of input text improves model accuracy with less training data (5.2).

(2) Investigate whether distinguishing inter and intra-sentential relation is beneficial to the evaluation results. We find that this distinction is indeed helpful and improves the test accuracy by 6.47%. It also suggests that Discourse Role Matrix benefits from inter-sentential relation more than intra-sentential ones. This is because Discourse Role Matrix only models transitions between sentences (5.5).

(3) Investigate the impact of entity extraction on the model performance. We find that by using gold-standard, better quality named entities, the model retain most of its performance with far fewer entities (5.4).

(4) Investigate the impact of more granular label on discriminative coherence evaluation. We find that Level-2 PDTB label has no impact on our task, and we conjecture that this is due to the curse of dimension (5.5).

(5) Investigate whether knowledge in one genre can be transferred to other genres with Discourse Role Matrix. We find that despite distributional differences across genres, cross-domain transfer performs well using our model. It implies that the model trained on discourse role transitions is domain-agnostic. (5.6).

(6) Investigate whether Convolutional Neural Network can be applied to extract longer transitions from Discourse Role Matrix. To our knowledge, this is the **first study** where Discourse Role Matrix is used in a neural setting. We present our methodology to adapt discourse relations in Nguyen and Joty's neural coherence model (3.7). We have found that longer transition length made possible by neural feature learning is indeed helpful to improve our coherence evaluation task (5.7).

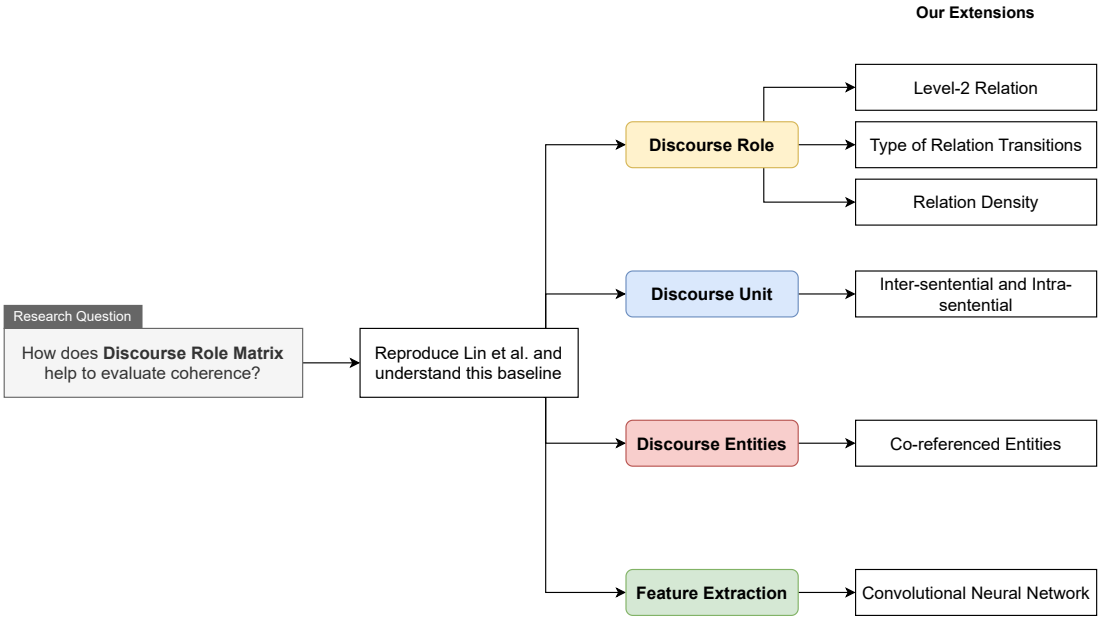


Figure 1.1: Overall Structure with Research Questions and Objectives

1.4 Report Structure

In this report, **Chapter 2** introduces the essential background of coherence modeling, Discourse Role Matrix and the context of related works. Main methodologies are explained in **Chapter 3**. **Chapter 4** explains our dataset and discriminative evaluation task, Shuffle Test. In **Chapter 5**, experimental setup and results are presented and discussed to answer our main research question. Last but not least, we conclude our findings in **Chapter 6**. We discuss limitations of the study and suggestions for our next steps.

Chapter 2

Background

Good writing requires a logical structure to organize and present a writer's thoughts. Thereby communicate it clearly to the reader. **Coherence**, in particularly, has a pivotal role in readable and meaningful writing. In this chapter, I introduce studies that formalize, model and evaluate various aspects of discourse coherence, to put my work in context.

2.1 Coherence: Theories and Frameworks

How can a reader find one text *choppy* and *disorganized* while another *clear* and *connected*? The reader can evaluate texts by how coherent they are, which are determined by how words and sentences are arranged. Coherent text binds sentences together as a whole, and interpretation of one sentence sometimes depends on the meaning of its neighbors [32]. Therefore, **Discourse analysis** considers the position, context, order and adjacency of a text [39]. These are intrinsic features that help us better understand text coherence beyond a single sentence.

In a piece of writing (or **discourse**), **coherence** refers to how each part of a text has a consistent meaning. Sometimes, coherence is also subjective and depends on how a reader interprets the text [41]. While **local coherence** refers to consistency across sentences or short passages, **global coherence** is across a whole document. Global coherence can sometimes be decomposed into many local coherence decisions, which we will illustrate in the next section.

Automatic evaluation of text coherence is one of the key components in many downstream NLP applications that include essay scoring [5], machine translation [46], question answering, readability assessment [2][34] and text generation [30]. Therefore, various discourse theories and frameworks have been proposed to computationally analyze coherence. These theories and frameworks deal with language phenomena across multiple sentences. We will introduce two main linguistic approaches in the following sections: **entities** and **discourse relations**.

2.1.1 Entities and Centering Theory

Coherent sentences often share a few important *topics*. Topics are short and concrete and should be consistent throughout the passage [41]. Since these topics connect sentences or even paragraphs, the reader can follow through the text easily. In discourse analysis, such topics can be approximated to **entities**, which are objects, groups of objects and events mentioned in a text [1]. They differ from *named entities* that discourse entities can be events (in form of verbs).

In **Centering Theory**, *salient* entities capture the focus in a point of discourse. Adjacent sentences that keep the same salient entity are more coherent than ones that repeatedly shift between different entities.

Entities are *salient* when the reader becomes more aware of their existence when reading a text. To model salience with linguistics features, Centering Theory ranks the degree of salience by grammatical roles. From most to least salient, these roles are *subject, direct object, indirect object, any other*, which are ordered by how prominent a syntactic position is. Other studies, such as **Entity Grid** model (to introduce in 2.2.1), also consider frequency of an entity as salience. In this case, coherence is created by repeated entity mentions.

To illustrate Centering Theory, consider this example from [10]:

Discourse A	**Discourse B**
a. John went to his favorite music store to buy a piano.	a. John went to his favorite music store to buy a piano.
b. He had frequented the store for many years.	b. It was a store John had frequented for many years.
c. He was excited that he could finally buy a piano.	c. He was excited that he could finally buy a piano.
d. He arrived just as the store was closing for the day	d. It was closing just as John arrived.

John is the main character of the story and the salient entity. Two discourses have the same meaning but Discourse A is more coherent than B. If we consider subjects in both discourse (underlined), Discourse A focuses on *John* throughout the text and keeps him as the subject. However, Discourse B first focuses on John, then the store, then back to John, then to the store again. Thus, a reader will focus on an entity that the discourse concerns the most locally [11]. In the example, *John* is the focus.

2.1.2 Discourse Relation

Discourse relation, or coherence relation, ties text spans with underlying logics or structures. It is a common device to signify text coherence. Let's revisit this example from the introduction:

(1) Jane took a train from Paris to Istanbul. She likes spinach.

(2) Jane took a train from Paris to Istanbul. She had to attend a conference.

(2) is more coherent because the second sentence provides a REASON to the first sentence. However, in (1) a reader will be less convinced that there is a causal relation.

As discourse relation is an important aspect of coherence, it has led to a proliferation of studies that formalize how they are structured in text. In the discourse community, one such popular model and corpora is **Penn Discourse TreeBank (PDTB)** [40].

PDTB defines discourse relations in local contexts. Each entry in PDTB mainly contains a discourse connective, relation sense and arguments. **Discourse connectives**, such as *because, although, when, since,* or *as a result*, are words that signal the relation type, or **relation sense**. **Arguments** (ARG1 and ARG2) are text spans to be connected by this discourse relation. When two arguments in a relation are connected by a discourse connective found in the text, the relation is **explicit**. The following is a good illustration of explicit discourse relation in PDTB:

> (3) *Jane took a train from Paris to Istanbul* because **she had to attend a conference**.

Here the subordinating conjunction *because* is a discourse connective that signals a causal relation between *Jane took a train from Paris to Istanbul* (Argument 1, italicized) and *she had to attend a conference* (Argument 2, bold). The relation sense in PDTB is CONTINGENCY.CAUSE.REASON, which means that ARG2 gives the reason, explanation or justification, while ARG1 gives its effect [40]. We can label each clause in **predicate-argument** style. That is, *Jane took a train from Paris to Istanbul* is CONTINGENCY.ARG1[1]. PDTB uses a three-level hierarchical classification so that the sense label has varying granularity of semantics: it encompasses **class** (Level-1, CONTINGENCY), **type** (Level-2, CAUSE) and **subtype** (Level-3, REASON) label.

When a discourse connective is not in the text such as (2), the annotator infers the relation sense (in Sentence 3, REASON) if they can and inserts an **implicit connective** (in Sentence 3, *because*) that best conveys the inferred relation.

Including CONTINGENCY, there are four Level-1 relation senses in PDTB. Below we provide examples for each Level-1 sense. Level-2 and 3 senses further refine the semantics in Level-1. The full sense hierarchy, with a more extended example set, is in Table A.1.

TEMPORAL	Situations described in the arguments are **synchronous** (events have overlap) or **asynchronous** (event in one argument precedes the other). In the example below, *knowing* and *eat* are synchronous: *Knowing a tasty – and free – meal* when **they eat one**, the executives gave the chefs a standing ovation. [wsj 0010]
CONTINGENCY	The situation described by one argument provides the **reason**, **explanation** or **justification** for the situation described by the other. In the example below, *no railroad damage* is the reason why the service resumes: *But service on the line is expected to resume by noon today.* (Implicit=since) **"We had no serious damage on the railroad,"** said a Southern Pacific spokesman. [wsj 1803]

[1]Throughout this report, I will format discourse relation and argument labels in small caps like this CONTINGENCY and ARG1, to be distinguishable.

COMPARISON	The discourse relation between two arguments highlights their **differences** or **similarities**. In the example below, the difference between *gold thriving on inflation* and *stocks thriving on disinflation* is contrasted: *After all, gold prices usually soar when inflation is high*. **Utility stocks**, on the other hand, **thrive on disinflation** . . . [wsj 0359]
EXPANSION	Relations that expand the discourse and move its narrative or exposition forward. In the example below, *speaking Filipino* is the detail how *turtle has succeeded*: *An enormous turtle has succeeded where the government has failed*: (Implicit = specifically) **He has made speaking Filipino respectable**.

Table 2.1: Examples of PDTB3 Level-1 relation sense, adapted from Webber et al. [40]. *Arg1* is italicized, **Arg2** is bolded and the connective is underlined.

2.2 Probabilistic Coherence Modeling

2.2.1 Entity Grid

Based on Centering Theory (2.1.1), Entity Grid represents the transitions of entities in adjacent sentences. This in turn provides a framework to evaluate text coherence.

In Figure 2.1, the text is split into sentences, which constitutes the rows of Entity Grid. The entities are extracted from the head of co-referent noun phrases, which include *Department*, *Microsoft* and *Netscape*. Entity Grid is a matrix where row i represents the i^{th} sentence and column j refers to j^{th} entity. A cell (i, j) contains the grammatical role of j^{th} entity in i^{th} sentence, which can be any of *subject* S, *object* O, *other* X or – if the entity is not in that sentence.

	Department	Trial	Microsoft	Evidence	Competitors	Markets	Products	Brands	Case	Netscape	Software	Tactics	Government	Suit	Earnings	
1	S	O	S	X	O	–	–	–	–	–	–	–	–	–	–	1
2	–	–	O	–	–	X	S	O	–	–	–	–	–	–	–	2
3	–	–	S	O	–	–	–	–	S	O	O	–	–	–	–	3
4	–	–	S	–	–	–	–	–	–	–	–	S	–	–	–	4
5	–	–	–	–	–	–	–	–	–	–	–	–	S	O	–	5
6	–	X	S	–	–	–	–	–	–	–	–	–	–	–	O	6

1 [The Justice Department]$_{\text{S}}$ is conducting an [anti-trust trial]$_{\text{O}}$ against [Microsoft Corp.]$_{\text{X}}$ with [evidence]$_{\text{X}}$ that [the company]$_{\text{S}}$ is increasingly attempting to crush [competitors]$_{\text{O}}$.
2 [Microsoft]$_{\text{O}}$ is accused of trying to forcefully buy into [markets]$_{\text{X}}$ where [its own products]$_{\text{S}}$ are not competitive enough to unseat [established brands]$_{\text{O}}$.
3 [The case]$_{\text{S}}$ revolves around [evidence]$_{\text{O}}$ of [Microsoft]$_{\text{S}}$ aggressively pressuring [Netscape]$_{\text{O}}$ into merging [browser software]$_{\text{O}}$.
4 [Microsoft]$_{\text{S}}$ claims [its tactics]$_{\text{S}}$ are commonplace and good economically.
5 [The government]$_{\text{S}}$ may file [a civil suit]$_{\text{O}}$ ruling that [conspiracy]$_{\text{S}}$ to curb [competition]$_{\text{O}}$ through [collusion]$_{\text{X}}$ is [a violation of the Sherman Act]$_{\text{O}}$.
6 [Microsoft]$_{\text{S}}$ continues to show [increased earnings]$_{\text{O}}$ despite [the trial]$_{\text{X}}$.

Figure 2.1: An example of Entity Grid from Barzilay and Lapata [2]. The text on the right is annotated with grammatical role for each entity. This text is represented as Entity Grid on the left.

Entity transitions are extracted from Entity Grid by taking all sequences of cell entries in each column. The sequence length is a parameter of the model, and it is often set to 2 or 3 with best results [2]. A short sequence length thereby captures local entity transitions, where the scope is across a few adjacent sentences. In Fig 2.1, the entity

Microsoft contains five entity transitions of length 2: $\{(\texttt{S},\texttt{O}),(\texttt{O},\texttt{S}),(\texttt{S},\texttt{S}),(\texttt{S},\texttt{-}),(\texttt{-},\texttt{S})\}$. One can then calculate transition probabilities in a grid. For example, $(\texttt{S},\texttt{-})$ appears 6 times, and there are 75 total number of entity transitions. Therefore, the transition probability $P(\texttt{S},\texttt{-}) = \frac{6}{75} = 0.08$. Transition probabilities of all transition types then become features for a predictor on coherence tasks.

2.2.2 Discourse Role Matrix

In 2.1.2, it is shown that the presence of discourse relation often indicates coherence in text. What's more, coherent text often favors arranging text spans in a discourse relation with one of two possible orderings. Consider this example from Lin et al.:

> [Everyone agrees that most of the nation's old bridges need to be repaired or replaced.]$_{S_1}$
> [But there's disagreement over how to do it.]$_{S_2}$

There is a CONTRAST relation in this sentence pair. The given ordering is coherent. If we were to swap this pair, the text will become incoherent. This motivates **Discourse Role Matrix**, a model that leverages the preference of relation ordering to assess text coherence.

While Entity Grid indicates the presence and grammatical role of an entity in a sentence, The Discourse Role matrix makes use of information on an entity's discourse relation, which we refer to as its **discourse role**. While Entity Grid refers discourse entity as a class of coreferent noun phrase [2], Discourse Role Matrix relaxes its choice of entities and uses stemmed form of open class words: nouns, verbs, adjectives, and adverbs [26].

Similar to Entity Grid, in Discourse Role Matrix, row i represents the i^{th} sentence and column j refers to j^{th} term. A cell (i, j) however contains the discourse relation type and argument label of term i in sentence j. In Figure 2.2, the term *cananea* appears in sentence 1 and takes part in the first argument of relation COMPARISON. Therefore, we mark $(cananea, S_1)$ with COMPARISON.ARG1. If term i does not appear in sentence j, or there is no relation that contains term i, we mark the cell *nil*. For example, $(cananea, S_2) = nil$ because the second sentence does not contain term *cananea*. As in Entity Grid, the probability of the discourse role transitions is calculated and used as features. A predictor is trained using these features to discriminate between transitions in coherent documents and those in incoherent documents.

The intuition behind Entity Grid and Discourse Role Matrix is that the distribution of entities in coherent text exhibits certain regularities that can be reflected in grid columns [24]. In particular, a discourse may be centered by a few entities that are salient throughout the passage. The grid column representing that entity will be dense with meaningful discourse roles (*cananea* or *operat* in Figure 2.2). If you combine the set of salient entities, sometimes you can even get the gist of a whole text (*cananea is operating something*). However, most other entities will be sparse or almost empty (like *depend*). They are not the main focus and only provide supporting information to salient entities.

[Japan normally depends heavily on the Highland Valley and **Cananea** mines as well as the Bougainville mine in Papua New Guinea. $]_{S_1}$ [Recently, Japan has been buying copper elsewhere. $]_{S_2}$ [[But as Highland Valley and **Cananea** begin operating, $]_{C_{3.1}}$ [they are expected to resume their roles as Japan's suppliers. $]_{C_{3.2}}$ $]_{S_3}$ [[According to Fred Demler, metals economist for Drexel Burnham Lambert, New York, $]_{C_{4.1}}$ ["Highland Valley has already started operating $]_{C_{4.2}}$ [and **Cananea** is expected to do so soon." $]_{C_{4.3}}$ $]_{S_4}$

5 discourse relations are present in the above text:

1. Implicit Comparison between S_1 as Arg1, and S_2 as Arg2
2. Explicit Comparison using "but" between S_2 as Arg1, and S_3 as Arg2
3. Explicit Temporal using "as" within S_3 (Clause $C_{3.1}$ as Arg1, and $C_{3.2}$ as Arg2)
4. Implicit Expansion between S_3 as Arg1, and S_4 as Arg2
5. Explicit Expansion using "and" within S_4 (Clause $C_{4.2}$ as Arg1, and $C_{4.3}$ as Arg2)

S#	Terms				
	copper	cananea	operat	depend	…
S_1	nil	Comp.Arg1	nil	Comp.Arg1	
S_2	Comp.Arg2 Comp.Arg1	nil	nil	nil	
S_3	nil	Comp.Arg2 Temp.Arg1 Exp.Arg1	Comp.Arg2 Temp.Arg1 Exp.Arg1	nil	
S_4	nil	Exp.Arg2	Exp.Arg1 Exp.Arg2	nil	

Figure 2.2: An example of Discourse Role Matrix from Lin et al. [26]. The text on the left is annotated with discourse relations.

2.3 Neural Coherence Modeling

With the rapid developments in deep learning, neural models have become prevalent in coherence modeling, surpassing previous approaches.

The efficiency of Entity Grid representation (2.2.1) in capturing entity distributions has inspired many extensions: Nguyen and Joty applies Convolutional Neural Network to the entity grid of an input text. The network looks up local regions of an Entity Grid and learn high-level entity-transition features with **convolution filters**. Because the filter size can be large (they use $5-8$), long-range transitions can be captured more efficiently than traditional Entity Grid. The training uses a supervised pairwise approach, where the model takes a pair of documents as input (a coherent and incoherent text) and outputs respective coherence scores. It minimizes the marginal loss, or maximize the distance between two scores.

Neural feature extraction improves probabilistic approach in two ways. First, probabilistic approach defines a length n for entity transitions of G different grammatical roles. This results in G^n transition probabilities to be calculated, which grows exponentially as n increases. Neural entity grid learns k filters of size n and can be applied to an entity grid globally. This results in only $k \cdot n$ feature sets to train. Secondly, probabilistic

approach calculates local transitions for a single document, without referencing other documents. The neural approach uses embedding vectors and convolutional filters to learn distributed representation of grammatical roles and local coherence patterns. It trains on many coherent and incoherent instances, which helps the network to generalize coherence modeling better to a wide range of documents and grammatical roles.

2.4 Current Approach and Rationale

We mainly focus on the **Discourse Role Matrix** (2.2.2), one of the first coherence models that utilizes discourse relations (particularly PDTB) to evaluate discourse coherence. At the time of publication, Discourse Role Matrix has achieved the state-of-the-art result of 89.25 % accuracy in differentiating between original and permutated text on WSJ dataset.

As mentioned in previous sections, Discourse Role Matrix combines two key aspects of coherence modeling: **entities based on Centering Theory** and **discourse relation**. This allows the model to be synergistic with other entity-based models. In fact, Lin et al. combine features from both their Discourse Role Matrix and Entity Grid of Barzilay and Lapata and achieve 1.2% improvement in WSJ dataset. Therefore, a deeper understanding of this architecture allows us to generalize coherence modeling knowledge to existing entity-based models.

In addition, despite it being a traditional approach that uses feature engineering, entity grid like Discourse Role Matrix is still utilized in recent work, with various downstream tasks such as essay scoring [9, 12, 16]. Its linguistic richness warrants closer examination on how existing approaches and new linguistics information (Section 3) affect its behavior in coherence modeling.

With recent interest in discourse-aware text generation systems, more work models the interdependence between sentences when generating coherent text in the document-level [42, 45]. Therefore, modeling and evaluating discourse coherence have become more crucial in such systems. In this study, we would also like to extend coherence relation based Discourse Role Matrix with Nguyen and Joty's neural model. Discourse relations are typically modeled as a discrete class in existing literature. The idea that a discourse role may be modeled as a distributed representation motivates multi-purpose, latent representations to capture local coherence patterns.

Our work builds on previous studies in coherence modeling with two key distinctions: First, we provide an extensive and in-depth analysis on how various aspects of Discourse Role Matrix contribute to discriminative coherence evaluation. To our knowledge, no research has studied the effect of discourse relation types to a coherence model, though some work has indicated their desire to do so [9]. Second, recent work to improve discourse modeling has focused on discourse relation classification (identifying discourse relation from raw text) [25, 33, 44] rather than coherence modeling. We adapt neural model to entity and discourse relation based Discourse Role Matrix to see its performance with long coherence context.

Chapter 3

Methodology

To evaluate text coherence automatically, I use discriminative approach to build my computational model: Supervised discriminative coherence models are trained to discriminate between labelled coherent and incoherent instances. These instances have same lengths and topics, therefore these aspects do not alter their coherence. To generate incoherent examples, **Shuffle Test** (4.2) permutes sentences in the original text that we consider to be coherent, so that the incoherent instance consists of the same sentences but in different order.

We select one of such models, **Discourse Role Matrix** (2.2.2) because it exploits not only entity but also discourse relation patterns to assess coherence. Discriminative model using discourse relations is one of the most recent traditional NLP method and a common approach to assess coherence and readability [9, 12, 16].

Then, I made modifications on the baseline in how discourse units, discourse entities, discourse roles and feature extraction are defined. These changes in discourse features allow me to interpret or improve my model across many phenomena in coherence. This chapter discusses their methodology and justification.

3.1 Baseline

We reproduce Discourse Role Matrix following parameters used in Lin et al. as the baseline model. There are four components to construct a Discourse Role Matrix: discourse units, discourse entities, discourse roles and feature extraction.

3.1.1 Discourse units

Discourse units are text segments that convey semantic meaning. In the Discourse Role Matrix, discourse units are argument spans in each relation. In my baseline, I divide a text into sentences, and each sentence defines a **matrix row**.

In my new model, I consider distinguishing discourse units that span within a sentence from those that span across sentences. We discuss this in detail in Section 3.2.

3.1.2 Discourse entities

To construct a Discourse Role Matrix, we must decide which words have sufficient meaning to be considered entities, and how to link different mentions of the same entity. We follow Lin et al. to use words with following part-of-speech (POS): nouns, proper nouns, verbs, adjectives and adverbs [1]. We use `spacy` library [2] to tokenize and filter with these word types. We have selected this tool because it supports in-context POS tagging. Notice that in the example below, the term *orange* is tagged as both adjective and noun depending on other terms in the context:

The	orange	monster	ate	the	orange	.
DET	**ADJ**	NOUN	VERB	DET	**NOUN**	PUNCT

This allows us to be flexible in identifying the relevant term as entities. The color *orange* is arguably less relevant than the *orange* being eaten. Determiner (e.g. *the*), punctuation, auxiliary (e.g. *is*) and conjunction (e.g. *and*) are **stopwords**. We remove these common terms that are insignificant to entities extraction.

We also use `PorterStemmer` to stem these terms. **Stemming** removes the prefixes and suffixes of words, so that terms of same root are grouped together. This allows us to resolve co-occuring entities naively by removing inflections: for nouns, stemming removes count and possessive case (e.g. *bank's* and *bank*). For verbs, stemming removes tense (e.g. *borrowed* and *borrow*).

In the baseline model, to save computation time, I only consider **salient entities**. This means each entity has a term frequency ≥ 2. This reduces noisy rare terms. Each entity extracted defines a **matrix column**.

We also consider alternatives of discourse entities, such as co-referred named entities, for our new model in Section 3.3.

3.1.3 Discourse roles

In Lin et al., discourse roles concatenate Level-1 PDTB discourse relation label (TEMPORAL, CONTINGENCY, COMPARISON, EXPANSION, ENTREL, NOREL) with argument label (ARG1, ARG2). Each discourse role defines a **matrix cell**.

In my new model, I also consider Level-2 relation labels. We illustrate the motivation and method in 3.4.

To construct a Discourse Role Matrix, we represent row i as the sentence S_i and column j as the entity e_j. If entity e_j appears in S_i with a discourse relation r, then we mark the cell entry (i, j) with this relation r plus the argument label in which the term is located. We call (i, j) the discourse role of e_j at sentence i.

Each cell (i, j) can be either empty, contain a single or multiple discourse roles. When an entity does not appear in a sentence, or it does not participate in a discourse relation,

[1] However, their POS tagging method is not mentioned, so exact replication is not possible.

[2] with default trained pipeline `en_core_web_sm`

the cell is empty, and we mark it *nil*. When an entity appears more than once in a sentence and participates in more than one relations, the cell contains many roles.

I follow the above algorithm and settings to re-implement Discourse Role Matrix. As there is no open-source implementations, I have written all modules myself and verify my implementation with manual checking of a few examples, and by comparing experimental result with Lin et al. (see Figure 5.1). Figure B.1 illustrates the pipeline for my implementation.

3.1.4 Feature Extraction

The three components above create a Discourse Role Matrix representation of input text. To build a coherence model to use such representation, we need to extract features for the model to discriminate between coherent and incoherent text. Our baseline uses the key assumption that **"coherent text exhibits measurable preferences for specific discourse relation ordering"** [26]. To illustrate this, consider this example from Lin et al.:

> (1) *Everyone agrees that most of the nation's old bridges need to be repaired or replaced.* (2) <u>But</u> **there's disagreement over how to do it.**.

Here Sentence 2 illustrates constrasting information about the Sentence 1 (signaled by connective *but*). If we swap them, it will produce an incoherent text [29]. Thus, the discourse role transition COMPARISON.ARG1 $\rightarrow$ COMPARISON.ARG2 is preferred in this context. We would like to capture transitions like this in our feature extraction step.

For every column that represent an entity, we look at local transitions between sentences. We count the sub-sequence of discourse roles in consecutive sentences of length 2 and 3.[3] In a given entity, they resemble **bigrams** (window of 2) and **trigrams** (window of 3) of discourse roles across adjacent sentences. Given the column for entity **cananea** below, all bigram transitions from S_3 to S_4 are (COMPARISON.ARG2 $\rightarrow$ EXPANSION.ARG2), (TEMPORAL.ARG1 $\rightarrow$ EXPANSION.ARG2) and (EXPANSION.ARG1 $\rightarrow$ EXPANSION.ARG2). One of the trigram transitions from S_2 to S_4 is (*nil* $\rightarrow$ EXPANSION.ARG1 $\rightarrow$ EXPANSION.ARG2). Notice we do not consider sub-sequence that only consists of *nil* (like *nil* $\rightarrow$ *nil* from S_1 to S_2). We count all bigrams and trigrams in a Discourse Role Matrix and calculate their probabilities. For example, if a matrix contains only **cananea**, since there are 6 length-2 bigrams, and EXPANSION.ARG1 $\rightarrow$ EXPANSION.ARG2 has a count of 1, its probability is $1/6$.

	S_1	S_2	S_3	S_4
cananea	*nil*	*nil*	COMP.ARG2 TEMP.ARG1 EXP.ARG1	EXP.ARG2

[3] This is the standard in existing work on entity grid [2][26]. Often subsequence longer than 3 scales up the number of features exponentially and leads to the curse of dimensionality [3] problem. We aim to resolve this in our neural Discourse Role Matrix in 3.2.

We also consider alternatives of feature extraction to replace the above probabilistic approach, using Convolution Neural Network. We illustrate this new model in Section 3.2.

Having defined the base model, we notice that Discourse Role Matrix presents numerous design opportunity to define discourse units, entities, roles and feature extractions. We will move on to explore some of these configurations in the subsequent sections.

3.2 Inter-Sentence vs Intra-Sentence Distinction

Discourse relations can be categorized based on the scope of their argument spans. An **intra-sentential relation** (*Intra-S*) contains arguments that both lie within the same sentence (Sentence 1 below), while an **inter-sentential relation** (*Inter-S*) contains arguments that jointly span across sentences (Sentence 2).

> (1) However **risky the business**, *it's brisk these days*. [wsj_0569]
>
> (2) *Small businesses say a recent trend is like a dream come true: more-affordable rates for employee-health insurance, initially at least.* **But** then **they wake up to a nightmare** [wsj _0518]

Intra-sentential discourse relations are explored with some degree in Lin et al.: In Figure 2.2, the fifth discourse relation EXPANSION is intra-sentential. Its arguments are two clauses in the last sentence. As a result, entities in the same sentence have different argument labels: *operat*, in first clause, is EXPANSION.ARG1. It is EXPANSION.ARG2 in second clause.

When modeling discourse roles, Lin et al. and its extended model [9] do not distinguish between inter-sentential and intra-sentential relations. However, distributions of relation senses are quite different between them [9]. We suspect that this can affect the decision boundary when discriminating coherent and incoherent instances. Therefore, we differentiate them when constructing Discourse Role Matrix. We add another dimension in discourse role: INTRA-S and INTER-S. For instance, the first argument of intra-sentential relation TEMPORAL will now have the role INTRAS.TEMPORAL.ARG1.

3.3 Entity Extraction

In Lin et al., entities are nouns, verbs, adjectives and adverbs that occur in text. Co-references are resolved naively by the stemmed form (e.g. *operat* is the stemmed form of *operating* and *operate*). However, when an entity appears as a pronoun, co-references of that entity cannot be resolved by stemming (e.g. *BankAmerica* and *it*). In addition, term-based resolution cannot map compound noun collectively to an entity. For example, *Bank of America* will be mapped to two entities: *bank* and *America*. Each co-reference becomes its own entity or column, which makes the matrix sparse.

Therefore, a robust coreference tool is needed to accurately identify co-occuring entities. To ensure co-references are resolved accurately, I use the gold-standard *BBN Pronoun Coreference and Entity Type Corpus*. The corpus contains a set of antecedents and their

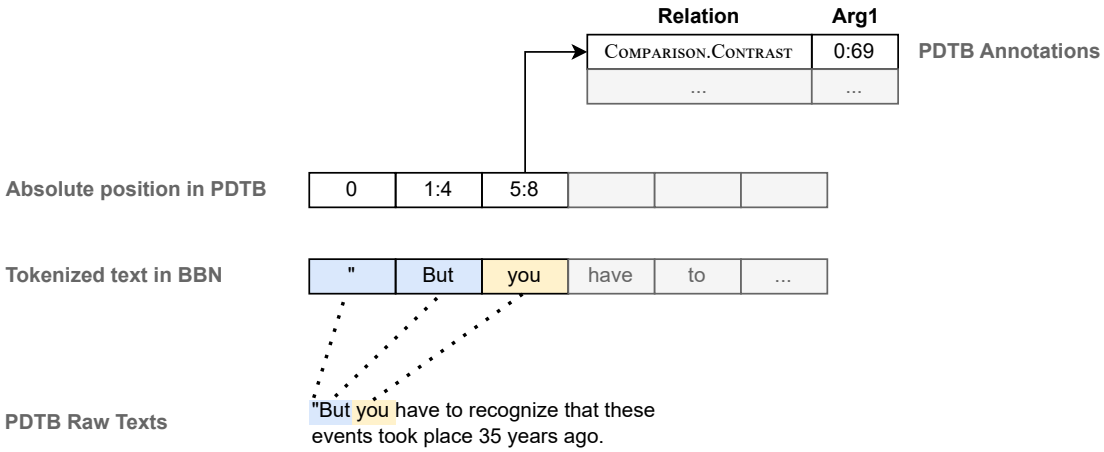


Figure 3.1: An illustration of Corpora Alignment between BBN and PDTB

corresponding pronouns (see below). **Antecedent** is the first occurence of an entity, and its **pronouns** are co-references of that entity, after the antecedent appeared. The corpus contains pronoun co-reference annotations in Wall Street Journal (WSJ) texts (4.1), which are the same documents we use to construct Discourse Role Matrix.

Listing 3.1: An example of antecedent and pronoun in *BBN Pronoun Coreference and Entity Type Corpus*

```
(   Antecedent -> S2:5-5 -> BankAmerica
    Pronoun -> S2: 7-7 -> it )
(   Antecedent -> S13:5-5 -> BankAmerica
    Pronoun -> S13: 34-34 -> it )
```

Corpora Alignment Because BBN corpus does not have information on its tokenization scheme, much effort is put in to align the term and sentence tokenization between BBN and PDTB, which we use for baseline. Figure 3.1 illustrates the alignment process: we first find every term that is tokenized by BBN and locate them iteratively in raw text and record its absolute position. This allows us to convert sentence and term indices to absolute position in PDTB raw text. In this way, pronouns and antecedents are aligned between two corpora. The antecedent and its pronouns are then stored as a single entity. Both their term and positions are recorded for matrix construction later. The absolute position of entities is then used to find the discourse relation it encompasses. We manually checked a few documents to ensure that the implementation is correct.

Resolved vs Unresolved Antecedents In the corpus, antecedents are separated based on each occurrence of that term. For instance, in Listing 3.1, the antecedent `BankAmerica` occured twice in the same text and are stored separately. We combine antecedents of same term to resolve such co-references. We call them **resolved** antecedents.

To see the effect of resolution, we also leave the antecedents of the same term the way it is, without combining them. Specifically, we treat `BankAmerica` in `S2` as a different

entity from BankAmerica in S13. We call them **unresolved** antecedents.

We suspect that our model will perform better when antecedents are resolved because an unresolved antecedent can only capture discourse transitions locally. Its scope is defined within the sentence indices of that antecedent and all its pronouns (in case of BankAmerica in S2, its scope is only S2). Therefore, unresolved antecedent cannot form a long chain of discourse role transitions.

3.4 Level-2 Relation Labels

In Lin et al., Level-1 PDTB relation labels are used as discourse role, which are TEMPORAL, CONTINGENCY, COMPARISON, EXPANSION. However, Level-1 label cannot accurately capture exact relation in adjacent sentences, especially when two different relations of same class co-occur in a context (e.g. EXPANSION.EXCEPTION and EXPANSION.CONJUNCTION).

Secondly, Level-1 label EXPANSION is a "mixed bucket" that contains Level-2 labels that are loosely related (e.g. EXCEPTION, CONJUNCTION). We provide full examples in Table A.1. EXPANSION means that "one of the arguments expands the discourse and move its narrative or exposition forward" [40]. This is less cognitively salient than its counterpart such as TEMPORAL, which means that arguments are time-ordered. Therefore, a coherence model may benefit from a more defined label as Level-2.

3.5 Support Vector Machine and Preference Ranking

Support Vector Machine (SVM) is a pattern recognition technique to create optimal decision boundary for patterns that are linearly separable. **Support vectors** are points that lie closest to the decision boundary. The objective of a SVM model is to find the linear boundary that is as far as possible from the points in different classes, while classifying them correctly. SVM takes a set of input pairs $(\mathbf{x}, y)$ and returns a set of weights $\mathbf{w}$ on each feature. Its linear combination predicts the value of y [4].

We use SVM to discriminate between a coherent text and a shuffled incoherent text. For a document, SVM uses input features $x = x_1, x_2, ..., x_n$, a vector of probabilities for all possible transitions of discourse roles, where n is the total number of features. We assume that the distribution of discourse role transitions in coherent texts is distinguishable from those in incoherent texts [26]. Therefore, SVM separates coherent from incoherent instances with an optimal hyperplane.

Since coherence is a relative measure of text quality without an absolute class label, we follow Barzilay and Lapata and Lin et al. to define the discriminative task as a **ranking problem**: Given a pair of texts, the system ranks them based on how coherent they are. We assign a coherent text d with rank 2 and a less coherent one d' with rank 1.

A preference-ranking SVM model [17] $\mathcal{M}$ uses the feature set for each document, $\mathcal{F}_d$ or $\mathcal{F}_{d'}$, as input and outputs a ranking score. Sorting the ranking score then gives the rank. Two documents of different ranks must meet **pairwise preferential constraint**. That is,

the output score given to coherent instance must be higher than that given to incoherent instance, $\mathcal{M}(\mathcal{F}_d) > \mathcal{M}(\mathcal{F}_{d'})$.

We use SVM-light package with rank setting because its runtime is much faster than `scikit-learn`. It is also used in Lin et al. and Feng et al., which helps to validate the correctness of my re-implementation by achieving equivalent results.

3.5.1 Model Interpretation

SVM in its linear setting, takes the dot product of weight with a given data point (Equation 3.1). If its result is positive, it belongs to the positive class. If it is negative, it belongs to the negative class. However, if it is in between, the model is less certain about its class.

$$y = \text{sign}(\mathbf{w}\mathbf{x} - b) \tag{3.1}$$

Therefore, weights indicate some importance of each feature for separating the data (i.e. the hyperplane would be orthogonal to the support vector). A large weight is more likely to assign data point with a positive label, given feature values are positive because they are probabilities. In fact, SVM weights have been used for feature selection in bioinformatics: Guyon et al. uses $\mathbf{w}^2$ as a feature ranking criterion to select genes for cancer classification.

In my coherence model, positive weights indicate relation transitions that are favored in coherent text, while negative weights indicate those common in less coherent text. Thus, SVM weights provide some evidence for model interpretability, which we will discuss in Section 5.3. To reduce the impact of noisy training data on model weights, we compare SVM weights in model instances trained from 5-fold cross-validation.

3.6 Genre Distinction and Domain Adaptation

The genre of a text often affects the distribution of discourse relations [44]. We would like to explore its impact to our model in evaluating coherence. Although PDTB contains only Wall Street Journal articles, which are mainly expository text, these articles can in fact be further categorized into essays, highlights, letters to editors, news articles and erratas (i.e. corrections and amplifications). We cross-reference two existing genre distinction sets from Webber [38] and Plank [35]. They use patterns in title, content structure, and other metadata to infer the genre of each document. As a preprocessing step, we exclude articles if two sources disagree about their genre. Five genres with the most articles are chosen. Their statistics are listed below. The remaining three genres (*Wit and short verse, Quarterly progress reports, Notable and Quotable*) are left out as they contain fewer than 15 articles.

Essays	Highlights	Letters	News	Errata
103	55	50	1902	23

Table 3.1: The number of WSJ articles in PDTB per genre

We would like to answer: **How well does our discriminative model adapt to different text genre?** We use domain adaptation, a particular case of transfer learning. In our case, **domain adaptation** applies the same task, coherence evaluation, between different domains, or genres. We define $\mathcal{D}_1$ and $\mathcal{D}_2$ as two genres we aim to compare, where their underlying probability distributions P_1 and P_2 differ. We present results of our domain adaptation in Section 5.6.

3.6.1 KL Divergence

To compare the distribution of discourse role transitions among genre, we calculate the KL divergence between two distributions. **Kullback–Leibler (KL) divergence** is a measure for the difference between two probability distributions over the same variable x. Provided that $p(x)$ and $q(x)$ are two distributions that share a discrete random variable x, the KL divergence of $q(x)$ from $p(x)$ is denoted as $D_{KL}(p(x)||q(x))$. This measures the information lost when $q(x)$ is used to approximate $p(x)$:

$$D_{KL}(p(x)||q(x)) = \sum_{x \in X} p(x) \ln \frac{p(x)}{q(x)} \tag{3.2}$$

$p(x)$ and $q(x)$ must fulfill the definition of probability distribution. That is, both $p(x)$ and $q(x)$ must sum to 1 over all x. $p(x) \geq 0$ and $q(x) \geq 0$ for any x in X. KL divergence of two probability distributions is 0 if and only these two distributions are the same. KL divergence is asymmetrical. That is, $D_{KL}(p(x)||q(x)) \neq D_{KL}(q(x)||p(x))$

We use KL divergence to compare the difference between two distribution of discourse role transition. In particular, we derive such distribution by counting the frequency of bigram transition (like COMPARISON.ARG1 $\rightarrow$ COMPARISON.ARG2) and trigram transition in Discourse Role Matrix (like COMPARISON.ARG1 $\rightarrow$ COMPARISON.ARG2 $\rightarrow$ *nil*). We then normalize these transitions to ensure that it is a probability distribution (as discussed in 3.3). We apply **smoothing** to account for unobserved transitions, where their probability is 0. This will ensure that KL divergence behaves reasonably in Equation 3.2 when $p(x)$ or $q(x)$ is close to 0.

3.7 CNN Discourse Role Matrix

While traditional feature extraction has some success in coherence modeling, it is still limited: By calculating discourse role transitions of length k (3.1.4), with $\mathcal{R}$ discourse roles, the number of such transitions $\mathcal{R}^k$ increases exponentially with larger k. This prevents the model from considering longer transitions [3]. In addition, traditional feature extraction is task-agnoistic, which means the same feature representations from entity grids are generated regardless of the downstream task. To solve these two problems, we decide to adopt Nguyen and Joty's convolutional entity grid to our Discourse Role Matrix. We discuss the mechanism of the Convolutional Neural Network (CNN) in the context of our coherence task.

Figure 3.2 summarizes our adapted neural architecture. The model takes a pair of documents as input, and outputs their respective coherence score. Before the text is

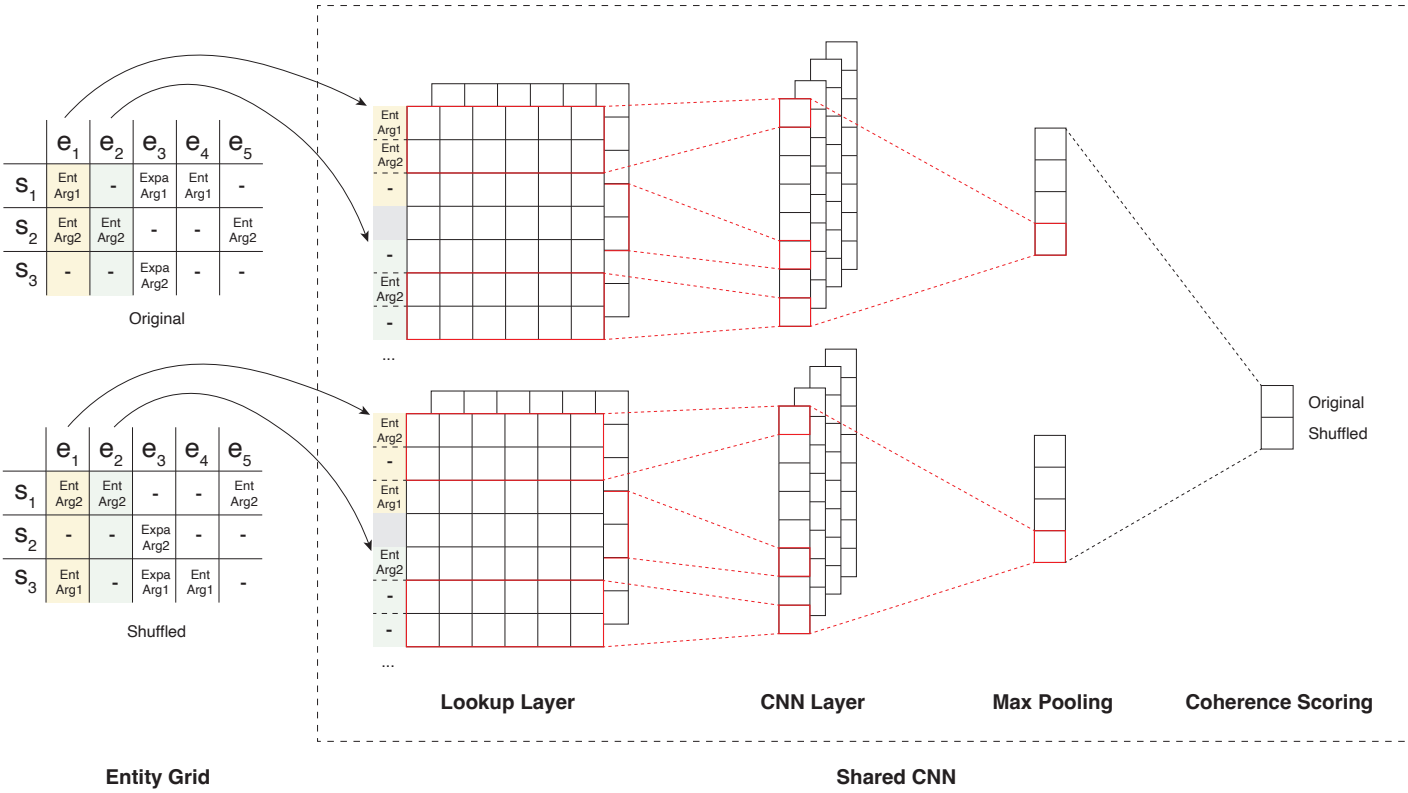


Figure 3.2: Architecture for convolutional Discourse Role Matrix for modeling coherence with pairwise training

fed to CNN, we construct its Discourse Role Matrix (Section 3.1). The first layer of the neural network transforms each discourse role into a distributed representation, or **embedding** vector that captures the meaning or relation of a discourse role. The second layer uses convolution to extract high-level features in each column of Discourse Role Matrix that represents an entity. The third layer chooses the most important high-level features, which are then used to compute coherence score for the text. We now elaborate on each layer of the model.

Lookup Layer We take each column of Discourse Role Matrix (yellow and green), which represents discourse role transitions of an entity between adjacent sentences.

Discourse roles are fed to our model as indices taken from a finite vocabulary $\mathcal{V}$. We obtain this vocabulary by finding all discourse role types in TRAIN set. For instance, our vocabulary will contain ENT.ARG1 and ENT.ARG2, which are first and second argument of ENTREL. We incorporate entity-specific feature to these discourse roles, by attaching the frequency of that discourse role found in that entity. For example, if the column for entity e is (ENT.ARG1, ENT.ARG2, - , ENT.ARG1). Then each ENT.ARG1 in that column will be added to the vocabulary as `Ent.Arg1_F2`, where `F2` means that the role occurs twice. Similarly, `F3` for 3 and `F4` for 4 and more. Our vocabulary size $|\mathcal{V}| = 43$, It includes empty discourse role `nil`, padding token `0`, and all Level-1 discourse roles with frequency `F2`, `F3`, `F4`.

For a Discourse Role Matrix G, The first layer of our CNN maps each of these roles $G_{i,j}$ occuring in the matrix to a distributed representation $\mathbb{R}^d$ by looking up a shared embedding matrix $E \in \mathbb{R}^{|\mathcal{V}| \times d}$. We do so for all m sentences and n entities. The discourse role found in a matrix is defined in the vocabulary, $G_{i,j} \in \mathcal{V}$. Formally, the output of **Lookup Layer** is

$$\mathcal{L}(G) = \langle E(G_{1,1})...E(G_{i,j})...E(G_{m,n}) \rangle \tag{3.3}$$

and is fed to the **Convolution Layer**. The discourse role embedding E is a model parameter that can be learned from back-propagation on a downstream task. We use the setting in Nguyen and Joty and initialize this embedding matrix by sampling from uniform distribution $\mathcal{U}(-0.01, 0.01)$. However, it can also be fine-tuned from pretrained embedding from a general coherence task.

Convolution Layer To extract high-level features from discourse role vectors produced by **Lookup Layer**, we use the **convolution operation**. We multiply weight parameters, or **filter** $\mathbf{w} \in \mathbb{R}^{kd}$ with entity transition of length k, $\mathcal{L}_{t:t+k-1,j}$. This denotes the concatenation of k vectors that represent discourse roles from sentence t to $t+k-1$, for entity e_j. Then we add the product with a bias term b_t. Lastly, we apply nonlinear activation function f to the sum. We follow Nguyen and Joty and use ReLU [31] in our model. The convolution operation results in a new abstract feature h_t.

$$h_t = f(\mathbf{w}^\intercal \mathcal{L}_{t:t+k-1,j} + b_t) \tag{3.4}$$

We apply the same filter $\mathbf{w}$ to all possible k-length transitions and entities. This will output a list of abstract features called **feature map**, $\mathbf{h}^i = [h_1, ..., h_{mn+k-1}]$. We can repeat this process N times with N different filters. This will obtain N feature maps. Each filter learns a set of transition patterns in our Discourse Role Matrix. For example, a filter may learn to detect COMP.ARG1 $\rightarrow$ COMP.ARG2 (2.1), a complete COMPARISON relation between two adjacent sentences. If an entity has the first argument of COMPARISON in sentence t, and the second argument in sent $t+1$, the resulting feature map in that region will have higher values than other regions without this transition. This may in turns affect the final layer to score this input higher.

As in Nguyen and Joty, we use the wide convolution [20]. This means that the convolution operation in Equation 3.4 reach all perimeters of $\mathcal{L}$, regardless of the window size. To do so, we apply zero-padding to out-of-range vectors $t < 0$ and $t > m, n$ (gray area in 3.2), which also separate different columns.

Convolution allows us to model discourse role transitions of arbitrary lengths k in a **location-invariant** way. This means that regardless that a transition of discourse relations occurs toward the beginning or end of text, the convolution will treat it equally for a given filter.

Pooling Layer After convolution, we apply max-pooling operation to each feature map $\mathbf{h}^i$. This reduces the output dimensionality, distillates the most salient features, and prepares for **Coherence Scoring** as the last step. We use $\mu_p(\mathbf{h}^i)$ to denote the max

operation to every non-overlapping window of p features in a feature map. We set the stride size same as the pooling size p so that windows are non-overlapping.

$$\mathbf{m} = [\mu_p(\mathbf{h}^1)...\mu_p(\mathbf{h}^N)] \tag{3.5}$$

Coherence Scoring Finally, the max-pooled output $\mathbf{m}$ is passed to a linear layer with weight $\mathbf{v}$ and bias b, to produce a coherence score y.

$$y = \mathbf{v}^\intercal \mathbf{m} + b \tag{3.6}$$

Training Objective As shown in Figure 3.2, we use **pairwise ranking** [7] to learn model parameter θ, similar to our SVM baseline. This means that the given an ordered pair of documents (d_i, d_j), where d_i is more coherent, we construct their Discourse Role Matrix G_i and G_j as the model input. Then the model minimizes the following ranking objective and outputs coherence score $y = \phi(G|\theta)$. The model shares its parameters θ when training both coherent and incoherent cases, allowing the network to learn coherence patterns from both. This is an advantage of a discriminative model.

$$\mathcal{J}(\theta) = \max(0, 1 - \phi(G_i|\theta) + \phi(G_j|\theta)) \tag{3.7}$$

Incorporating Discourse Roles Nguyen and Joty uses grammatical roles in their entity grid. This means that an entity can be either absent, subject, object or other in a given sentence. Unlike grammatical roles, in our model, there may be more than one discourse roles $\mathbf{G}_{i,j}$ for a given entity e_i and sentence j. This makes modeling input challenging because in the **Lookup Layer**, the model finds the distributed representation of a single discourse role in that position. To resolve this, in a given entity e_i, we enumerate all possible combination of that entity column where there is only one discourse role at at every sentence $1, 2, ..., m$. This is equivalent to **Cartesian product** of discourse roles in an entity column. We randomly downsample the resulting columns to 1000 for every column, if the number of combinations is too large for the model to load. Through experimentation, we obtained this upper limit so that the time required to load the matrices is reasonable.

An entity column

	S_1	S_2	S_3	S_4
cananea	*nil*	*nil*	COMP.ARG2 TEMP.ARG1 EXP.ARG1	EXP.ARG2

Cartesian Product

	S_1	S_2	S_3	S_4
cananea	*nil*	*nil*	COMP.ARG2	EXP.ARG2
	S_1	S_2	S_3	S_4
cananea	*nil*	*nil*	TEMP.ARG1	EXP.ARG2
	S_1	S_2	S_3	S_4
cananea	*nil*	*nil*	EXP.ARG1	EXP.ARG2

We adapt discourse roles by modifying the open-source implementation published in Nguyen and Joty [4]. The training module uses outdated *Tensorflow* and *Keras* framework, so much effort involves resolving the environment and deprecated functions. Due to large computation required for hyperparameter tuning (5.7), I have put much time in setting up multiple Google Cloud virtual machine instances to train in parallel, each of which require configurations for running our neural model.

[4] https://github.com/datienguyen/cnn_coherence/

Chapter 4

Data and Evaluation Task

4.1 Data

The Penn Discourse Treebank (PDTB) [36] [40] is the largest discourse-annotated corpus, with 2162 *Wall Street Journal* (WSJ) articles. In the discourse community, PDTB is a gold-standard corpus for training and evaluating coherence model. It has a higher annotator agreement in relation sense identification and a larger annotated corpora [1] than its counterpart [2], Rhetorical Structure Theory Discourse Treebank (RST-DT) [6] [18]. This enables our method to learn from an underlying distribution that is more representative, with less noise and subjective bias from human annotators.

As discussed in 2.1.2, PDTB contains discourse relations in predicate-argument style: a discourse connective (e.g. *because*, *and*) is a predicate. Two text spans that the connective joins are arguments, which are stored as absolute positions in text. We use these annotations to construct discourse role for our model. Below is a snippet of the dataset. Other discourse information (such as attribution) are left out since they are irrelevant to our model.

...	**Connective Semantic Class**	**Arg1**	**Arg2**
	COMPARISON.CONTRAST	600..722	543..598
	EXPANSION.LEVEL-OF-DETAIL.ARG2-AS-DETAIL	756..776	778..874
	COMPARISON.CONTRAST	778..874	876..916
	CONTINGENCY.CAUSE.RESULT	921..1043	1046..1104

Table 4.1: A snippet of PDTB (Document wsj_0003)

In Table 4.1, we call each row an **annotation** (or annotation instance). The column **Connective semantic class** refers to the discourse relation between two argument spans. Notice that it often has three-level (EXPANSION.LEVEL-OF-DETAIL.ARG2-AS-

[1] RST-DT contains 100K words, while PDTB-3 contains 1M

[2] RST-DT has 65.8% annotator agreement for all relation identification, while PDTB has 94% for class (Level-1), 84% for type (Level-2) and 80% for subtype (Level-3) label identification.

DETAIL), sometimes two. This granularity of relation label is utilized in one of our variant models (Section 3.4).

PDTB 3.0 vs 2.0 In particular, we have chosen PDTB 3.0 as our dataset. It has been modified from PDTB 2.0 to include more consistency testing, giving the current version better quality. It cover more instances of discourse relations ($\sim$ 13K more), particularly in intra-sentential context. When replicating Lin et al., we have removed these additional annotations to ensure an *apple-to-apple* comparison between our baseline and theirs.

Dataset Set-up We first randomly splits articles following the experiment setting in Lin et al. [26]: 1040 articles for training, 42 articles for development and 1079 articles for testing.

When an article contains more relations, the abundance of relation transitions may help distinguish the original article from its permutation better [26]. We balance the train split by the **density** of relations in an article. The density is the ratio between the number of relations in the article and the article length. This ensures that the train split contains a balanced representation of articles of various densities. We discuss its result in Section 5.2.

4.2 Shuffle Test

Shuffle Test [2] is the most common evaluation for coherence modeling. In this task, we supervise the model to distinguish between an original document and the same document in which the sentence order has been permutated. We assume that the original text is more coherent and is ranked higher than the shuffled one. In fact, this assumption has been validated in Lin et al. with human evaluation, with 90% inter-subject agreement in WSJ dataset. Therefore, a successful coherence model should prefer the original ordering.

The ability to choose the correct sentence order has been essential in text generation and multi-document summarization [2]. These are common NLP tasks to which an automatic coherence evaluation model can apply to.

We use documents in *Penn Discourse Treebank (PDTB)*, which contains 2162 *Wall Street Journal (WSJ)* articles, the standard dataset for this test. For each document, we create 20 random permutations by shuffling the original order of the sentences. This results in 20 pairwise rankings between the original and shuffled text. Documents that contain less than four sentences can produce less than 20 permutations. In this case, we include all permutations. We remove any permutation that is the same as the original text.

Evaluation Using the test split in Section 4.1, we conduct 5-fold cross-validation and use default regularization parameter C in `svm-light` package, as in Lin et al.. We evaluate our coherence model with the **ranking accuracy** of Shuffle Test:

$$\text{ranking accuracy} = \frac{\text{\# of pairwise rankings correctly predicted by the ranker}}{\text{total \# of pairwise rankings}} \quad (4.1)$$

The pairwise ranking is correctly predicted by the ranker if it returns a higher score to the coherent original document than the incoherent shuffled one.

When both coherent and incoherent instance are given the same coherence score, as sometimes found in our CNN model (5.7), we call it **ties**. When a coherent instance is ranked higher than incoherent, we call it **win**, and **loss** if incoherent instance is ranked higher. To account for instances of ties, we calculate precision, F1 and recall as in Nguyen and Joty:

$$\text{precision} = \frac{\text{win}}{\text{win} + \text{loss}} \tag{4.2}$$

$$\text{recall} = \frac{\text{win}}{(\text{win} + \text{loss} + \text{tie})} \tag{4.3}$$

$$F_1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{4.4}$$

4.3 Handle Sentence Shuffling

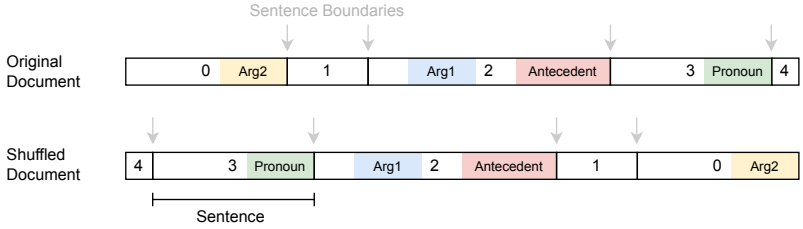


Figure 4.1: An illustration for how argument spans and entities are handled in sentence shuffling. The numbered block denotes a sentence. Each colored block denotes either argument spans (Arg1, Arg2) or co-reference resolved entities (antecedent, pronoun) in sentences.

Here we explain how to process text annotations, argument and entity, when we permute the sentence order. We visualize this in Figure 4.1.

Argument Recall that discourse relations we use to construct our matrices contain two argument spans: ARG1 and ARG2. In PDTB, the absolute positions of argument spans in original document are stored (See 4.1). To ensure that argument spans in the original document match with those in shuffled document, we split the document into sentences and record the sentence boundaries (i.e. absolute positions at the start of each sentence) per document before and after permutation. We then find which sentences these arguments belong to and map argument positions accordingly from original to shuffled document.

We retain all discourse relations in the original text during Shuffle Test, even if two arguments are far from each other after shuffling. For example, the original text consists of sentences $[A, B, C]$ in that order. A contains CONTRAST.ARG1, B contains CONTRAST.ARG2 and C does not contain any role. Given that the permuted instance is

$[B, C, A]$, there remains a CONTRAST between B and A. However, the relation transition is only captured if the transition length is larger or equal to the distance between B and A. For instance, transition length of 3 can capture the permuted instance as (CONTRAST.ARG2, *nil*, CONTRAST.ARG1) but a transition length of 2 can not. This allows the model to capture long-range relation where arguments are not in adjacent sentences.

Entities In my baseline, entities are stemmed open class words that occur at least twice. We store these eligible words in an entity vocabulary, so that they can be used for shuffled instances of that document. To find which entities are included in an argument span, we sweep all terms in the span to see if any belongs to our entity vocabulary. Because co-referred entities are resolved naively through term repetitions, it is not necessary for us to store the absolute positions of antecedents and pronouns, like we do in Section 3.3.

However, in our model with co-reference resolutions, absolute positions are necessary to ensure that the the antecedent and its pronouns are matched. These positions are recorded in BBN corpora as sentence and token indices. For instance, `S1:1-2` means that the entity is in the first sentence and is the first and second term of that sentence (see Listing 3.1 for a complete example). To match sentence ordering before and after permutations, we record the sentence indices respectively. For example, the original document is $[0, 1, 2, 3, 4]$, where each number denotes a sentence. One of its shuffled instance can be $[4, 3, 2, 1, 0]$. This allows us to trace back the absolute positions of entities in shuffled text.

Chapter 5

Experiments and Discussion

To answer our research question, **what aspects of Discourse Role Matrix help a model to evaluate text coherence**, we first present our reproduced baseline and analyze its preference on relation density and relation transition. We observe that the baseline prefers continuation of same relation sense and text with high relation density. This supports coherence theory and the validity of model behavior. Motivated by our observations in the baseline model architecture (3.1), we present our linguistically-enriched models using (1) types of discourse entities (2) intra-sentential and inter-sentential distinction for discourse relation (3) granularity of discourse relation label and (4) domain transfer across genres. We investigate their respective implications in discriminative coherence evaluation. Finally, we present the findings of our new CNN coherence model and conclude why neural feature learning enhances the statistical approach.

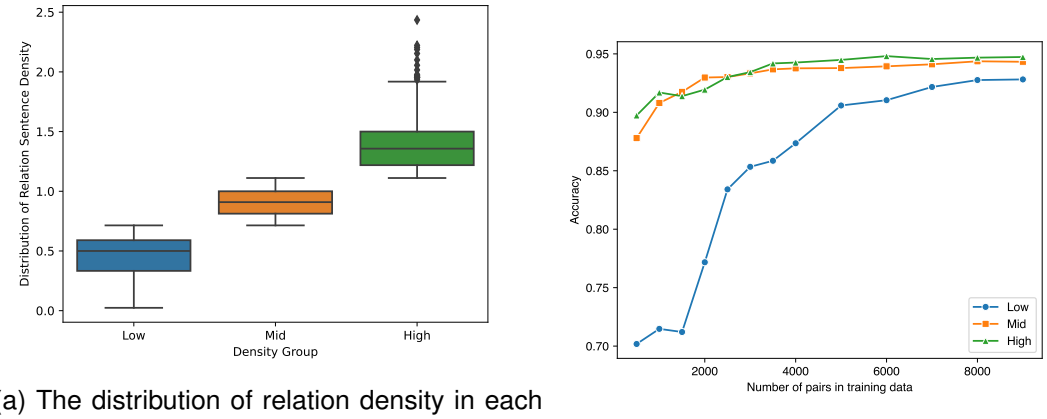
5.1 Baseline

We refer to our implementation of the Discourse Role Matrix model in Lin et al. as the baseline, which we will extend in subsequent experiments. We reproduce the Shuffle Test (4.2) result on *Wall Street Journal (WSJ)* articles, where each article has 20 distinct permutations. We create our own permutations because the random seed in Lin et al. is not provided. However, upon setting several random seeds, we see consistent performance regardless of how these permutations are randomly generated. This is also the case in the literature, where similar performance is achieved when replicating Shuffle Test without knowing the exact permutation in each article [26][32].

	Accuracy
Lin et al	88.06
Baseline	88.09
Baseline + Balanced	**88.65**

Figure 5.1: Test set ranking accuracy for Wall Street Journal (WSJ) data. Test accuracy is averaged across 5-fold cross-validations. **Baseline + Balanced** indicates that the train set contains balanced representation of articles with different densities (5.2).

The similar result of our replication (Row 2 in Figure 5.1) with Lin et al. provides some evidence that our implementation is correct. There is 0.03% difference in



(a) The distribution of relation density in each density group

(b) The learning curve for each density group

Figure 5.2: The impact of relation density on baseline Discourse Role Matrix model

the TEST set ranking accuracy. Some variability is expected because WSJ filenames for dataset split is not shared in the original study, so the exact duplication is not possible.

5.2 Relation Density

In Lin et al. [26], the accuracy of the model differs *across* data sets because the density of discourse relations varies. It is easier to distinguish high density articles from its permutation. We investigate the impact of relation density *within* the same dataset, but with a focus on learning behavior. In each article, we first calculate the density metric, which is the ratio between the number of relations and the article length. Then we use it to rank and split all WSJ articles into three equal-sized groups: low, mid, and high relation density, which have an average density of 0.452, 0.911, 1.38 respectively. To gauge a *typical* density, the average density is 1.2 for all WSJ articles [26]. Figure 5.2a plots the distribution of relation density in each group. 9282 document pairs are held out to evaluate accuracy.

For each density group, we train a model on different number of training pairs. The resulting learning curve is presented in Figure 5.2b. It confirms our hypothesis that the model performance benefits from high relation density. The accuracies for all three groups increase rapidly until 3000 pairs, where accuracies of mid and high group slow down improvement. High density group outperforms other density groups most of the time, although performance of mid density group is comparable to that of high density group. This shows that the performance advantage of using a density greater than 1 is diminishing. On the other hand, low density group requires much more training pairs to reach comparable result like other groups, and its learning behavior is much more different.

To answer **why the learning behavior of low density group is different than others**, we scrutinize its data file. Upon inspection, we realize that in low relation density group, there are more pairs of coherent and incoherent text containing the same transition probabilities as feature vector. This suggests that when a coherent text is less dense

in discourse relations (in an extreme case, without any discourse relation), the chain of relation transitions may not distinguishable from incoherent text. In this case, the model needs to resort to other linguistic devices for coherence, such as grammatical roles (2.2.1) or lexical cohesion [14].

Specific to Shuffle Test, it is also observed that in low-density group, there are often less sentences. Sentence length can affect the difference between feature sets. A longer sentence length can often create more displacements when sentences are shuffled (e.g. Compare $[1,2] \rightarrow [2,1]$ with $[1,2,3,4,5,6,7] \rightarrow [5,2,6,1,3,7,4]$). Therefore, the space of incoherent instances accounted is so much smaller for low-density group. In this case, an evaluation task would work better if it disrupts natural ordering of phrases and clauses.

From these observations, we control the effect of density on our model by balancing the density in the training set. We see a small improvement in test accuracy across all folds (Figure 5.1, Row 3), which suggests that the model gains advantage from a balanced representation of articles with different densities.

5.3 Relation Transition

To understand **how the model sees coherent and incoherent texts**, we interpret it using weight vectors from linear SVM (Section 3.5.1) and transition probabilities in our dataset. In our model, SVM uses relation transition probabilities as input, so weights indicate the importance of a relation transition in coherence scoring. Though weights are only indicative, we observe general patterns where some group of relation transitions are recommended or repressed when evaluating coherence.

Same-Relation Transition Weights for all bigrams are visualized in Figure 5.3. We have marked regions of heatmap with dashed square for same-relation transitions. In each square, ARG1 → ARG2 is the upper right cell. We observe that the five bigram transitions with highest weights are of the same relation type, with ARG1 followed by ARG2. This does not come as a surprise. Coherence theory indicates that adjacent discourse units are often connected with the same relation. Additionally, in our dataset, the argument order for adverbials and coordinating conjunctions is normally ARG1 followed by ARG2 [36].

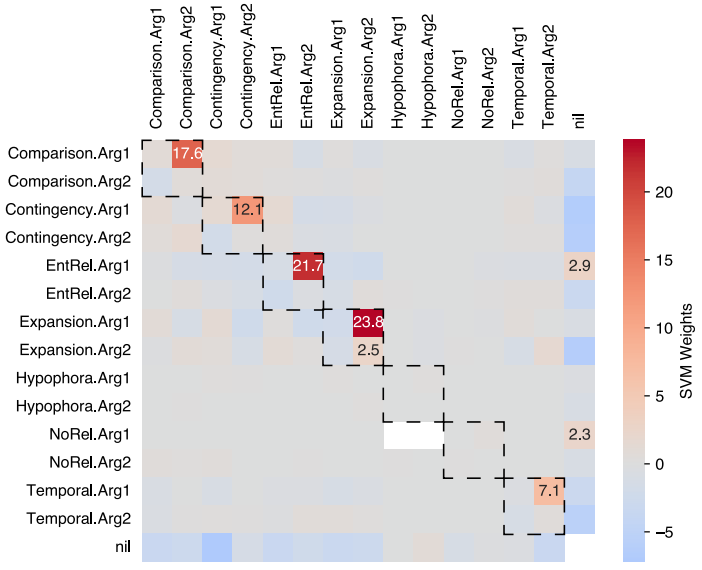


Figure 5.3: Feature weights for all bigram transitions

We have noticed, however, that HYPOHORA.ARG1 → HYPOPHORA.ARG2 is the least important same-relation transition in

our baseline SVM. We provide an example below, where ARG1 is italicized and ARG2 is bolded. We suspect that this is because HYPOPHORA is a new addition to PDTB3 and is less annotated than any other relation type (constitutes 0.47% of inter-sentential tokens annotated) due to its specificity in question-answering. This renders the model with less examples to explore HYPOHORA in the wild. Nevertheless, We do not deny its contribution in coherence evaluation task. This HYPOHORA transition has a feature weight higher than 72.39% of all bigram transitions.

(1) *If not now, when*? "**When the fruit is ripe, it falls from the tree by itself," he says**." [wsj_0300]

However, it is much less prevalent for ARG2 followed by ARG1 among important bigram transitions of same relation type. Some relation types actually discourage placing ARG2 before ARG1. For instance, ENTREL has the most negative SVM weight in this scenario (Table 5.4). This is because ENTREL often introduces the new entity in ARG1 before more information about the entity. Therefore, it is reasonable that its inverse is less coherent. For instance, in the example below, it would be odd to mention *Mr. Milgrim* as a successor without introducing him first:

(2) *Hale Milgrim, 41 years old, senior vice president, marketing at Elecktra Entertainment Inc., was named president of Capitol Records Inc., a unit of this entertainment concern*. EntRel **Mr. Milgrim succeeds David Berman, who resigned last month**.

An exception for this pattern is NOREL, where the model does not penalize this relation if ARG2 precedes ARG1. This does not come as a surprise as NOREL means no relation holds between them. Therefore, reordering its argument does not improve or worsen coherence:

(3) *Jacobs is an international engineering and construction concern*. NoRel **Total capital investment at the site could be as much as $400 million, according to Intel**.

We observe similar patterns when we average transition probabilities across all Discourse Role Matrices constructed from original PDTB documents. This represents the distribution of training data that is used for SVM in Shuffle Test. In Figure 5.5, we annotate each heatmap cell with its probability if it is larger than 1%. This means that relation transitions in those cell are more typical in coherent texts. Then, we mark the same-relation transition with dashed square. The upper right cell in each dashed square represents same-relation transition from ARG1 $\rightarrow$ ARG2. We notice that these cells are often occupied with large probabilities, in fact often the largest for that relation. This again confirms our belief that same-relation transition with ARG1 followed by ARG2 is most favored in evaluating coherence.

What is interesting is that, besides same-type relation, cells of high transition probabilities contain one argument that is ENTREL or NOREL. These cells are from rows and columns whose axis labels we have tinted *gray*. This observation suggests that ENTREL and NOREL co-occur frequently. Despite their frequent occurrence in the dataset,

Relation sense	Model weight
ENTREL	-2.24
CONTINGENCY	-1.87
COMPARISON	-1.70
EXPANSION	-1.34
TEMPORAL	-1.22
HYPOPHORA	-0.04
NOREL	0.12

Figure 5.4: Same-Relation Transition where ARG2 precedes ARG1

our model does not pick up on this pattern, where its feature weights are close to 0 (see 5.3). This showcases the benefit of discriminative model, where it does not overly rely on the distribution of coherence instances but also learn from incoherent ones.

Relation Transition with NIL *nil* indicates a break in relation transition, where an entity is absent, or does not have a discourse role in neighboring sentences. This provides evidence for incoherence. Our result in Figure 5.3 confirms this belief with low SVM weights when *nil* is present (blue patches in *nil* row and column). However, *nil* occurs in a few bigram transitions with high feature weights, particular with ENTREL (2.9) and NOREL (2.3). Nevertheless, it has also been discussed by Lin et al. that a text with relations of these two types are harder to contrast coherence with its permutated text [26]. This is because ENTREL and NOREL are less conclusive relation sense: While NOREL simply means no relation between spans (Sentence 3), ENTREL indicates ARG2 contains some information about entity in ARG1, but we are uncertain with the exact discourse connective to put between arguments (Sentence 2).

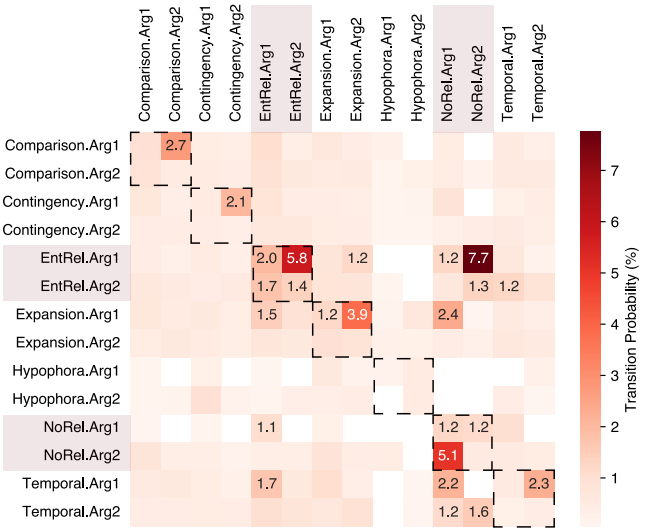


Figure 5.5: Saliency heatmap for **Bigram Transition Probabilities** macro-averaged across all Discourse Role Matrix in original PDTB texts

In addition, the matrix may not pick up the same entity in both spans. It is also not guaranteed that the same entity mentioned in ARG1 reappears in ARG2. For instance, the entity may be either inferred (without explicitly mentioned again, as below) or replaced with pronouns. We can see that simply stemming the entity word is not sufficient in capturing entity's re-occurence. This warrants further investigation using co-reference resolution.

(4) *"I am happy to see the spirit of the people," said Mr. Sisulu*, (EntRel) **looking dapper in a new gray suit**. (wsj 2454) [wsj_0300]

5.4 Entities Extraction

Motivated by our observations above, we use expert-annotated data as gold-standard for co-reference resolution (as described in 3.3). We use 1950 WSJ articles that are overlapped between PDTB, the dataset we use to construct matrices, and BBN, our gold-standard entity dataset. This generates 33509 document pairs for Shuffle Test.

The results are shown in Figure 5.6. Our first impression is that TEST accuracy using

gold-standard is lower than our baseline. However, the BBN dataset uses far fewer entities per document. The average number of entities in BBN is 8.28 (when antecedents are resolved), compared to 63 in our term-based baseline. This shows that Discourse Role Matrix is distillated with true salient entities to represent discourse role transitions. The reduction is also reflected in computation time: On a 2.3 GHz Quad-Core Intel i7 CPU, it takes on average 0.024 second per data pair in our new model, compared with 1.21 second in baseline. To conclude, our model that uses gold-standard entities is able to achieve comparable performance while retaining only a few entities. It provides a small trade-off between model accuracy and cost in computation and space.

	Accuracy
Baseline	88.09
Resolved Antecedents	84.77
Unresolved Antecedents	84.61

Figure 5.6: Test set ranking accuracy for Wall Street Journal (WSJ) data for model with Entity Extraction. Test accuracy is averaged across 5-fold cross-validations.

To answer **how much information do entities in each model convey**, we calculate the percentage of sentences in which each entity appears with a discourse role. We name this **entity information**. If an entity appears in every sentence with at least one discourse role, then the entity information is 1. A high entity information indicates that the entity is salient in the passage. To illustrate, the entity **cananea** below has an entity information of $2/4 = 0.5$. We calculate this statistics across all entities in the baseline, our antecedent resolved and unresolved model (3.3). The distribution of entity information is illustrated in Figure 5.7.

	S_1	S_2	S_3	S_4
cananea	*nil*	*nil*	COMP.ARG2 TEMP.ARG1 EXP.ARG1	EXP.ARG2

We observe that the distribution of entity information in baseline is more right skewed than both our extended models. This shows that fewer entities in the baseline contain sparse entity column, and we believe this is the reason that the baseline outperform our extended model. Although test performance between resolved and unresolved model are similar, we can still see the benefit of a resolved model, where its distribution is more right skewed in the lower tail. This indicates that more entities carry higher discourse information.

Our model using unresolved antecedents does not degrade the performance significantly. We believe that it is uncommon that there are many antecedents of same string that are unresolved in BNN. In particular, when we resolve same-string antecedents, the average number of entities is 8.28. This is similar to 9.6 when antecedents are not resolved.

If we observe the upper tail of the distribution in Figure 5.7, we can see that more proportion of entities in our extended model contain filled or almost filled entity column. This shows the benefit of co-reference resolution. By resolving antecedents with pronouns, our extended models carry more entities with high information of discourse role transitions.

Nevertheless, we believe that BNN dataset has its limitation because it only contains **named entities** with pronouns. This limits our scope of high information entities

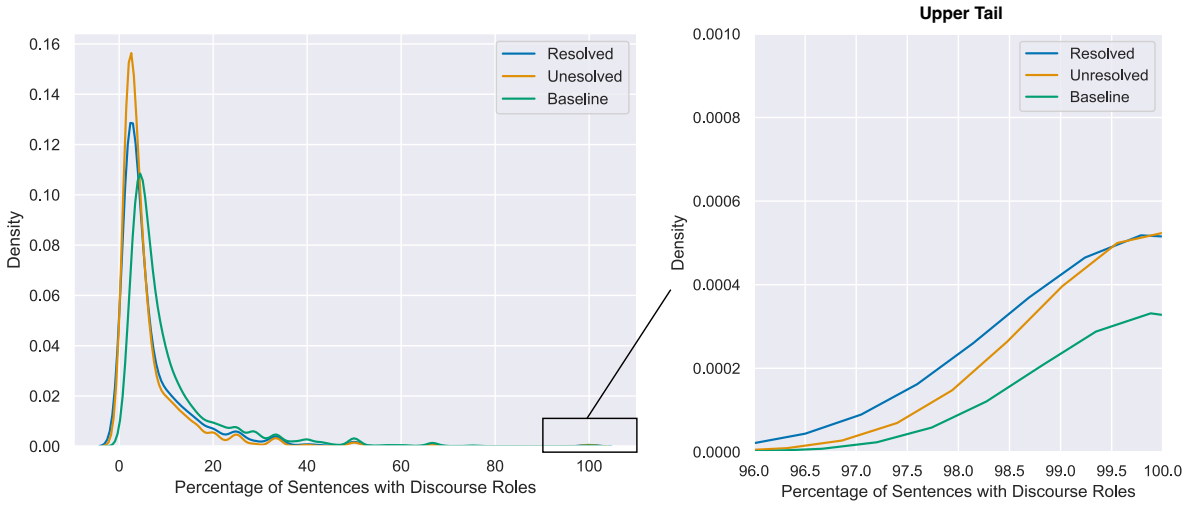


Figure 5.7: Full density plot (with upper tail) for entity information distribution in **Baseline** (green), **Antecedent Resolved** (blue), and **Unresolved** (yellow) models.

because their antecedents are often proper noun. Events (in form of verb), which can be salient in text, are not annotated in this dataset. We have made an effort to scout out additional datasets. Nevertheless, this is also true in entity-annotated corpus *OntoNotes*.

5.5 Ablation Study

We perform ablation study to answer **how Level-2 labels and distinction between inter and intra-sentential relation affect model performance**. We eliminate each of them from the full model (Figure 5.8). In Row 3 and 4, We delete transition probabilities for intra-sentential and inter-sentential relation respectively. Therefore, Row 3 only uses relation that contains INTERS in the label, and Row 4 only uses those with INTRAS (as discussed in 3.2). In Row 5, we delete Level-2 label (underlined below), so only Level-1 relation is used in the discourse role. An example would be:

$$\text{INTERS.EXPANSION.}\underline{\text{CONJUNCTION}}\text{.ARG1} \rightarrow \text{INTERS.EXPANSION.ARG1}$$

Row 5 is equivalent to the baseline but with distinction between inter-sentential and intra-sentential relation (i.e. INTERS and INTRAS in the relation label).

Inter and Intra-sentential Distinction Comparing Row 2 with Row 4, we see drastic performance reductions after eliminating inter-sentential relation. However, comparing Row 2 with Row 3, the performance improves slightly though not significant, after eliminating intra-sentential relation. Similar trend can also be observed in the case where only Level-1 label is used (by comparing Row 5-7). These results suggest that inter-sentential relation plays an important role in mod-

		Accuracy
1	Baseline	88.09
2	Level-2 + InterS + IntraS	93.25
3	Level-2 + InterS	94.14
4	Level-2 + IntraS	57.57
5	InterS + IntraS	94.18
6	InterS	**94.47**
7	IntraS	60.40

Figure 5.8: Test set ranking accuracy for Wall Street Journal (WSJ) data for model with Entity Extraction. Test accuracy is averaged across 5-fold cross-validations.

eling relation transitions, while the contribution for intra-sentential relations is not significant. We suspect this is because Discourse Role Matrix only models transitions between sentences. Because the model uses sentences as discourse unit, both arguments of a relation will be placed in the same cell, making it impossible to model transitions within a sentence. Surprisingly, to the best of our knowledge, this observation has not been reported in the literature [9, 26]. Our results shows that for future work, it is important to design new discourse model that utilizes intra-sentential relations.

Level-2 Label Unexpectedly, Level-2 label does not contribute to better performance. Comparing Row 3 and 5, there was no significant differences after eliminating Level-2 label. We speculate three probable causes: First, the finer granularity results in less instances for each Level-2 label. A solution is to use end-to-end discourse parser [27] to generate more relation-annotated data and augment instances for each Level-2 relation in the wild. We conjecture that longer document may contain more discourse relations and benefit from Level-2 distinction.

Secondly, perhaps it is rare for a document to contain multiple instances of Level-2 relation (such as CONTINGENCY.CAUSE), even with more relation-annotated data. Therefore, transition for Level-2 relations will be sparse, and Level-2 distinction is not necessary. In PDTB, we have conducted ablation study to eliminate each of Level-2 relations under EXPANSION, and the effect is not significant in the discrimination task. This corroborates our observation.

Lastly, to represent discourse role transitions, Level-2 label expands the number of features. While there are 4 Level-1 labels, there are as much as 22 Level-2 labels. The number of features is further amplified by taking argument label (ARG1 andARG2) and transitions of length 2 and 3. In particular, given n discourse roles, the Discourse Role Matrix needs to compute $n^2 + n^3$ transitions. Therefore, the **curse of dimensionality** problem [3] exacerbates when using Level-2 label. Similar to our second point, the distribution obtained from training data becomes very sparse and prevents the model from using more refined discourse role (such as Level-3 label), or longer transitions. Therefore, we aim to alleviate this problem with neural feature learning in Section 5.7.

5.6 Genre

Prior study has shown that label distribution of relation senses are sensitive to the genre of given text [44]. In Figure 5.9, we show that even within a homogeneous corpora of WSJ articles, the label distribution varies across article genres. Similarly to our analysis in 5.5, we average transition probabilities across all Discourse Role Matrices constructed from articles in four genres: NEWS, ESSAYS, HIGHLIGHTS and LETTER. For clarity, we annotate heatmap cells with their probabilities if higher than 25% of all transition probabilities.

In Figure 5.9, we have found that for each genre, there is a cluster of cells in the heatmap with high transition probabilities. We mark them in dashed square. These cells signify common relation senses found specific to that genre. For instance, transitions between COMPARISON is most common for LETTERS. In fact, COMPARISON was

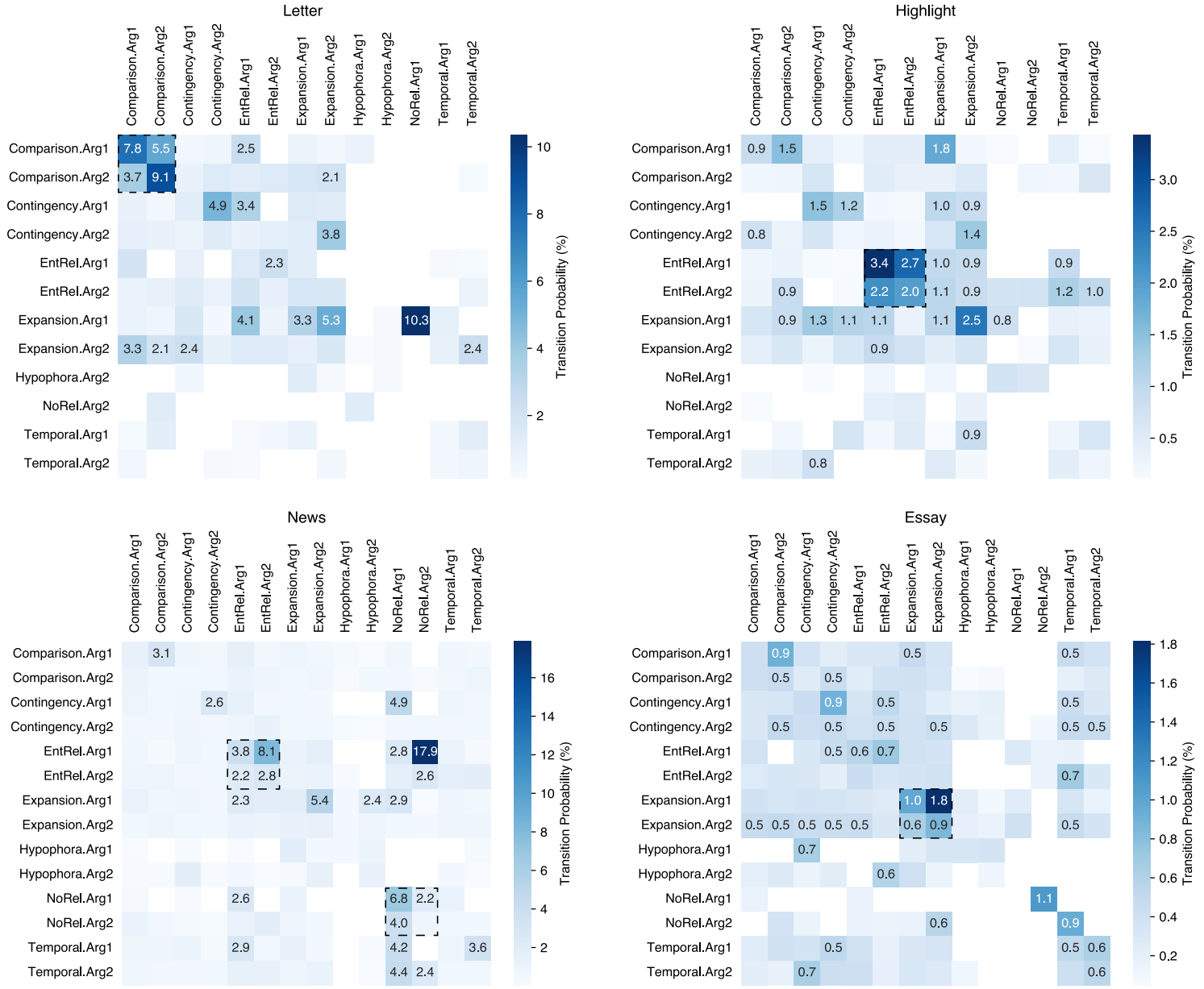


Figure 5.9: Bigram Transition Probabilities macro-averaged across Discourse Role Matrix in original PDTB texts of four genres: **Letter** (top left), **Highlight** (top right), **News** (bottom left) and **Essay** (bottom right). Clusters of high transition probabilities are marked with dashed square

used approximately twice as often in labelling explicit inter-sentential connectives in LETTERS than in NEWS [38]. Upon inspection, these LETTERS are addressed from readers to editors to explain their opinions why an article is erroneous (see example below), so LETTERS are more argumentative. COMPARISON is thus a common device to compare what is written in the article (italicized) and what the reader believes (in bold):

> *This statement surely buttresses your editorial viewpoint that environmental protection is generally silly or excessive*, but **it is simply wrong**. [wsj 2108]

Similarly, in Figure 5.9, ENTREL is most common for HIGHLIGHTS. EXPANSION is most common for ESSAY. ENTREL and NOREL is most common for NEWS. We provide more examples for these genres in Appendix C. It is apparent that the distribution for NEWS has significantly impacted the overall distribution found in Figure 5.5, where transitions containing ENTREL and NOREL are most emphasized. We can conclude from this analysis that there are not only differences in distributions of *relation label* between genres, but also *relation transitions*. These distribution differences are likely to impact our coherence model when we apply domain transfer.

The findings above motivate our experiment to apply domain adaptation to articles in these four genres. This allow us to study the potential of transferring knowledge across text genres in evaluating coherence. In the **cross-domain setting**, the linear SVM model is trained on the subset of PDTB articles in the source domain, and then evaluated on the articles in the target domain. In the **in-domain setting**, we train the model only on articles in the target domain and evaluate on a held-out TEST set of that same domain. Similar to previous experiments, we perform 5-fold cross-validation and report the ranking accuracy in Table 5.1.

Source → Target	**News**	**Essay**	**Highlights**	**Letter**
News	93.57	**98.90**	95.51	98.33
Essay	87.63	**94.57**	88.97	88.19
Highlights	86.02	92.01	**96.41**	87.36
Letter	83.86	**89.45**	84.74	88.75

Table 5.1: Test set ranking accuracy for Wall Street Journal (WSJ) data on different target (columns) and source (rows) domain/genre pairs. Test accuracy is averaged across 5-fold cross-validations. For every source domain, the best performing target domain is in bold.

We have found that the distribution of relation transitions in HIGHLIGHT is more similar to that of ESSAY. As evident in Figure 5.9, high transition probabilities are dense in upper left region of heatmap, where transitions with COMPARISON, CONTINGENCY, and especially ENTREL and EXPANSION are most frequent. Different than NEWS, both HIGHLIGHT and ESSAY are low in bigrams with NOREL. This distribution similarity is evident in cross-domain transfer performance: model trained on HIGHLIGHT perform the best in ESSAY (excluding in-domain setting), and vice versa.

Source → Target	**News**	**Essay**	**Highlights**	**Letter**
News	-	3.27	7.27	8.60
Essay	1.12	-	4.90	6.44
Highlights	0.76	1.07	-	4.10
Letter	1.13	1.06	7.22	-

Table 5.2: KL divergence of transition distribution in target (columns) domain from source (rows) domain, or $D_{KL}(\text{source}(x)||\text{target}(x))$, where x is the probability of discourse role transition of length 2 and 3. KL divergence is asymmetrical, $D_{KL}(p(x)||q(x)) \neq D_{KL}(q(x)||p(x))$

To quantify the distributional difference between two domains, we also calculate the KL divergence between source and target domain. We observe that best performing target domain in Table 5.1 often has a low KL divergence from the source domain. This provides evidence that sometimes distributional similarity helps domain transferring using our model.

Letter	**Highlight**	**Essay**	**News**
79	68	32	4

Table 5.3: The number of transition types that are not found for each genre, from most to least sparse. Their probabilities are smoothed when calculating KL divergence.

Strong evidence of cross-domain transfer is observed across all source and target genre pairs, and the model is less susceptible to cross-domain transfer than expected. Despite the large distributional differences between NEWS and LETTER, the performance of cross-domain transfer is much more significant than random guess.

We also observe that when a source domain has a complex distribution and a wide range of non-zero transition probabilities, like NEWS, the performance for domain transfer is the most successful. This is evident in first row of Table 5.1. NEWS has a low KL divergence from any other source domain (first column in Table 5.2). This is because KL divergence is calculated across all transitions, and the probability distribution of KL in NEWS is less sparse than other domains (Table 5.3).

The success of domain adaptation tells us that in Shuffle Test the model learns from similar characteristics across genres despite distributional differences. Our result confirms Xu et al. that discriminative models work well in domain adaptation. A model that is trained on a plethora of relation transitions, such as NEWS, is helpful when applying it to other target domains. Distributional similarity is also helpful when a domain with complex distribution is not available. We can approximate it with a genre that has similar but less complex transition distributions.

5.7 CNN Discourse Role Matrix

Experiment Setting We first initialize the CNN model as described in Section 3.7. Then, we feed the model with Discourse Role Matrix with only **inter-sentential** relations, as they are best performing in our ablation study (Section 5.5). To make our result

comparable with previous sections, we use the same data split and random seed for the Shuffle Test (4.2). We train the model by optimizing the pairwise ranking loss, which maximizes the difference of coherence score between coherent and incoherent text [1]. We use up to 25 epochs. To combat over-fitting, we have set dropout after max pooling and final dense layer. We also early-stop if the DEV accuracy does not improve for 10 executive epochs.

Hyperparameter Tuning In our neural model, we have the following hyperparameters: window size for discourse role transition, embedding size for discourse role, pooling size, number of filters in convolution layer and dropout rate for training. We fix **embedding** size to 100 and **dropout** rate to 0.5. We conduct limited search for optimal **mini-batch** size in $\{32, 64\}$, **window** size in $\{3, 4, 5, 6, 7\}$, **pooling** length in $\{5, 6, 7, 8\}$, and **filter** number in $\{150, 250\}$. We emphasize search in window size and pooling length, to study the effect of long range discourse role transition on coherence modeling. We have chosen these settings by gauging model complexity with that in Nguyen and Joty. We have found that the number of our discourse roles (or model vocabulary) are more similar to their extended grid model [2], so we use their optimal hyper-parameters as a starting point for our tuning.

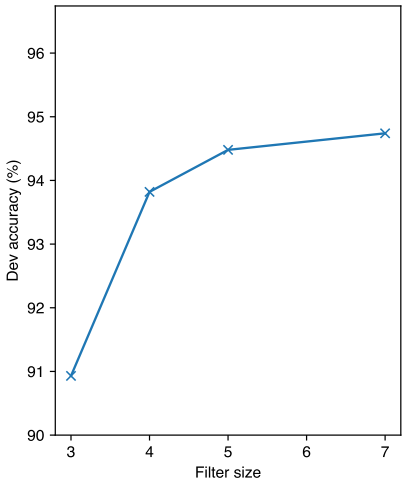


Figure 5.10: DEV set accuracy on varying filter sizes, for batch 32, filter size 150 and pool size 6

Figure 5.11: Parallel Plot for Hyperparameter Tuning Results. Red, green and blues line denote low, mid and high DEV set accuracy respectively. Full result with TRAIN and DEV loss is in Table B.1

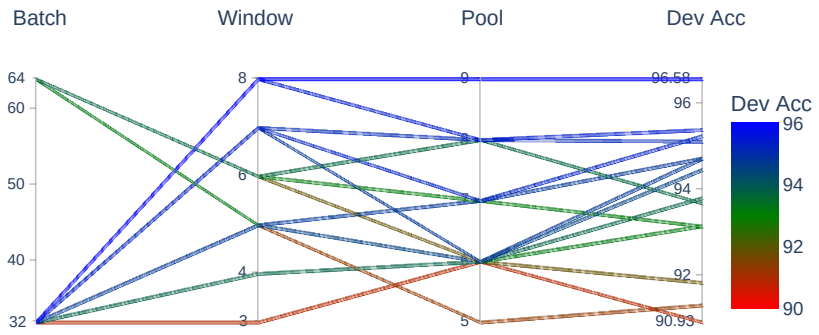


Figure 5.11 visualizes the hyperparameter search performance on the discrimination task. The full result with TRAIN and DEV loss is in Table B.1. Each line denotes one of the hyperparameter combination, and is colored *red*, *green* and *blue* to denote LOW, MID and HIGH DEV set accuracy that setting achieves. The result confirms our belief

[1] Using gradient-based online learning algorithm RMSprop, as implemented in Nguyen and Joty.

[2] Their optimal hyper-parameter setting is the following: batch size of 32, embedding size of 100, dropout 0.5, filter size of 150, window size of 5 and pool size of 6

that a larger window size of discourse role transitions benefits model performance. In Figure 5.10, we observe a leap in DEV set accuracy, from 90.93% to 93.82% when the window size increases from 3 to 4, in a particular setting. But this trend is seen in other settings as well. The DEV loss and accuracy continue to improve when we increase the filter size, which is less feasible in SVM due to the curse of dimensionality [3, 32], as illustrated in Section 3.7.

In addition, we observe that in general, it is best to set the pool size slightly larger or equal to the filter size. This is also the case in Nguyen and Joty. We believe that larger pooling helps the model to generalize local transition patterns and capture prominent patterns more broadly, thereby reducing the workload on the final layer when calculating the coherence score. Given comparable window and pool size, our model tends to lose its generalization ability when mini-batch size is larger (64). This is common in other deep learning scenarios when a larger batch can attract sharp minima, and thereby less likely to escape the basins of the gradient descent landscape [21].

Batch	**Filter**	**Window**	**Pool**	**Train Loss**	**Dev Loss**	**Dev Acc**
32	150	8	9	0.0438	0.7953	96.58

Table 5.4: Optimal hyper-parameter for our CNN Discourse Role Matrix

We then use the best performing hyperparameter setting in DEV set (Table 5.4) for the final evaluation on the TEST set. Neuralizing Discourse Role Matrix has made some promising improvement from our extended model that uses only Level-1 inter-sentential relations. This demonstrates that convolutional feature learning and distributed representation of discourse roles are effective in discriminative coherence evaluation. In addition, information of discourse relations greatly improve Nguyen and Joty's neural model that uses only grammatical role, as these discourse relations formally define logical and semantic relation, and provide linguistically rich information beyond sentence structure. Note that their best result uses entity-specific features in their vocabulary, such as named entity type and whether an entity has a proper mention. We believe that if we adopt this, there may be further improvements, as evident in their work.

Model	**Test Accuracy**	**Test F1**
Baseline	88.09	-
Nguyen and Joty	88.69	88.69
Level-1 InterS	94.47	-
CNN	**95.37**	**95.38**

Table 5.5: Test set performance for coherence evaluation on Shuffle Test, using Wall Street Journal (WSJ) data

In addition to adaptable window sizes for modeling transitions, our CNN model is also trained on many coherent and incoherent instances. This enables the neural network to learn more general and robust feature representations than probabilistic feature extraction, which is calculated within a document.

Chapter 6

Conclusions

The goal of this project is to examine closely the first coherence model that uses information from discourse relations and entities, Disourse Role Matrix. To answer the research question: **What aspects of Discourse Role Matrix help the model to evaluate coherence?** We dissect the model into 1) discourse units 2) discourse entities 3) discourse roles and 4) feature extraction, provide in-depth analysis on each aspect, and investigate their relations with discriminative coherence evaluation.

We first verify the effectiveness of Lin et al.'s model by interpreting what types of **discourse role** transition are favored and repressed in coherence evaluation. We use weights of SVM and statistics of discourse role transitions, and have found that same-relation transition is favored, which agrees with coherence theory. Transitions with *nil* is repressed as they often signify that a discourse relation discontinues (5.3).

Motivated by various design choices in these four aspects, we experiment on using addition sources of linguistic knowledge to interpret the model or improve its performance: We separate intra and inter-sentential relations, as they represent different scope of **discourse units**. We find that this distinction improves the model by 6.47% in test accuracy from the baseline. In addition, since Discourse Role Matrix uses sentences as discourse units, inter-sentential relations benefit the discrimination much more than intra-sentential ones (5.5).

For **discourse entities**, the baseline does not consider entities co-reference and extract entity naively through part-of-speech. We demonstrate gold-standard co-reference corpus as an alternative for entities extraction. We find that the model retain most of its performance with far fewer entities of better quality (5.4).

For **discourse roles**, we found that even within the same corpus, the relation density of input text affects model accuracy. In particular, high relation density requires less training data in learning our discrimination task (5.2). Upon inspecting the hierarchy of PDTB, we observe that Level-1 senses can contain loosely related Level-2 senses. Thus, we use granular Level-2 relation labels as an alternative for discourse roles. However, this does not help the model to further improve its discrimination ability (5.8). We argue that this is due to the curse of dimensionality and less training instances for each Level-2 sense. In addition, there often does not contain multiple instances of a Level-2

relation in an input text.

We supplement the extensions above by investigating how well does our model adapt to different text genre. This is because the distributions of relation transitions are very different across genres. However, we have found that the model can generalize well in a cross-domain setting, which reflects that local coherence cues learnt by the model has similar characteristics across genres. We also discover that a genre with a plethora of discourse role transitions, such as NEWS in WSJ, can effectively transfer to another genre of less complex distribution.

To study the impacts of **feature extraction** on evaluating coherence, we compare probabilistic n-gram transitions with features learnt from our new CNN model. The neural model provides promising improvement and captures longer range of discourse role transitions, which is less feasible in the statistical approach due to the curse of dimension. We learn that logical and semantic relation in discourse role is helpful in modeling coherence, which improves from Nguyen and Joty's neural entity grid.

The significance of the project lies in its implications for future automatic coherence evaluation systems. Recent work in discourse analysis has made leap in automatic discourse relation classification [28]. There is also growing effort in coherence evaluation for text generation systems that rely on Centering Theory and entity grid used in this work [45]. These attempts further illustrate that it is vital to understand this linguistically rich model to introduce meaningful improvements. We have shown a wide range of model architecture that largely affects its coherence evaluation ability.

From our experiments in 5.8, we illustrate that existing model does not fully utilize intra-sentential relations, even though they are majority of annotated instances. Coherence models should consider beyond adjacent sentences and attend to relation transitions between clauses or even phrases.

In addition, our CNN model is not fully **lexicalized**. This means that the model does not consider other lexical information of entities, such as named entity type (e.g. PERSON, ORGANIZATION). A more generalized coherence model will consider these entity-specific information and semantic similarities across text. We aim to explore this in future by combining in-context entity embedding from language model with linguistic features suggested in Elsner and Charniak.

Despite that Shuffle Test is a long-standing benchmark for coherence evaluation, these artificially created instances do not reflect realistic incoherence in the wild. Though some work has been proposed to design more difficult task, such as *k*-block Shuffle Test [22], sentence insertion [37], or predicting human judgment score on real-world text [23], the community should build more elaborate coherence measures to capture more complete set of linguistic phenomena in text coherence.

Lastly, this work uses discourse relations in accordance with PDTB. This relation style uses shallow structures by connecting two clauses and sentences. Recent work has shown that coherence model using Rhetorical Structure Theory (RST), which annotates deep hierarchical discourse structure is better at differentiating text coherence [9]. In future work, we wish to adapt Graph Neural Networks in RST tree (similar to our CNN Disocurse Role Matrix) and explore its effect on discriminative coherence evaluation.

Bibliography

[1] Damaris Ayuso. Discourse entities in janus. In *27th Annual Meeting of the Association for Computational Linguistics*, pages 243–250, 1989.

[2] Regina Barzilay and Mirella Lapata. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34, 2008.

[3] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

[4] Robert Berwick. An idiot's guide to support vector machines (svms). *Retrieved on October*, 21:2011, 2003.

[5] Jill Burstein, Joel Tetreault, and Slava Andreyev. Using entity-based features to model coherence in student essays. In *Human language technologies: The 2010 annual conference of the North American chapter of the Association for Computational Linguistics*, pages 681–684, 2010.

[6] Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. Building a discourse-tagged corpus in the framework of rhetorical structure theory. *Current and new directions in discourse and dialogue*, pages 85–112, 2003.

[7] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12:2493–2537, 2011.

[8] Micha Elsner and Eugene Charniak. Extending the entity grid with entity-specific features. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 125–129, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL `https://aclanthology.org/P11-2022`.

[9] Vanessa Wei Feng, Ziheng Lin, and Graeme Hirst. The impact of deep hierarchical discourse structures in the evaluation of text coherence. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 940–949, 2014.

[10] Barbara J Grosz, Aravind K Joshi, and Scott Weinstein. Centering: A framework for modelling the local coherence of discourse. 1995.

[11] Barbara J Grosz et al. The representation and use of focus in a system for understanding dialogs. In *IJCAI*, volume 67, page 76, 1977.

[12] Camille Guinaudeau and Michael Strube. Graph-based local coherence modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 93–103, 2013.

[13] Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46:389–422, 2002.

[14] Michael Alexander Kirkwood Halliday and Ruqaiya Hasan. *Cohesion in english*. Routledge, 2014.

[15] Jerry R Hobbs. Coherence and coreference. *Cognitive science*, 3(1):67–90, 1979.

[16] Guimin Huang, Min Tan, Zhenglin Sun, and Ya Zhou. Rst-based discourse coherence quality analysis model for students' english essays. In *MATEC Web of Conferences*, volume 232, page 02020. EDP Sciences, 2018.

[17] Thorsten Joachims. Making large-scale svm learning practical. Technical report, Technical report, 1998.

[18] Shafiq Joty, Giuseppe Carenini, Raymond Ng, and Gabriel Murray. Discourse analysis and its applications. In Preslav Nakov and Alexis Palmer, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 12–17, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-4003. URL https://aclanthology.org/P19-4003.

[19] Dan Jurafsky and James H. Martin. *Speech and language processing*. Prentice Hall, Pearson Education International, 2014.

[20] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014.

[21] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

[22] Philippe Laban, Luke Dai, Lucas Bandarkar, and Marti A Hearst. Can transformer models measure coherence in text? re-thinking the shuffle test. *arXiv preprint arXiv:2107.03448*, 2021.

[23] Alice Lai and Joel Tetreault. Discourse coherence in the wild: A dataset, evaluation and methods. *arXiv preprint arXiv:1805.04993*, 2018.

[24] Mirella Lapata, Regina Barzilay, et al. Automatic evaluation of text coherence: Models and representations. In *Ijcai*, volume 5, pages 1085–1090, 2005.

[25] Li Liang, Zheng Zhao, and Bonnie Webber. Extending implicit discourse relation recognition to the pdtb-3. In *Proceedings of the First Workshop on Computational Approaches to Discourse*, pages 135–147, 2020.

[26] Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. Automatically evaluating text coherence using discourse relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 997–1006, 2011.

[27] Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. A pdtb-styled end-to-end discourse parser. *Natural Language Engineering*, 20(2):151–184, 2014.

[28] Xin Liu, Jiefu Ou, Yangqiu Song, and Xin Jiang. On the importance of word and sentence representation learning in implicit discourse relation classification. *arXiv preprint arXiv:2004.12617*, 2020.

[29] Daniel Marcu. Distinguishing between coherent and incoherent texts. In *The Proceedings of the Student Conference on Computational Linguistics in Montreal*, pages 136–143, 1996.

[30] Neil McIntyre and Mirella Lapata. Plot induction and evolutionary search for story generation. In Jan Hajič, Sandra Carberry, Stephen Clark, and Joakim Nivre, editors, *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1562–1572, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL `https://aclanthology.org/P10-1158`.

[31] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[32] Dat Tien Nguyen and Shafiq Joty. A neural local coherence model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1320–1330, 2017.

[33] Joonsuk Park and Claire Cardie. Improving implicit discourse relation recognition through feature set optimization. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 108–112, 2012.

[34] Emily Pitler, Annie Louis, and Ani Nenkova. Automatic evaluation of linguistic quality in multi-document summarization. In *Proceedings of the 48th annual meeting of the Association for Computational Linguistics*, pages 544–554, 2010.

[35] Barbara Plank. PTB/PDTB files belonging to different genres, 1999. URL `https://www.let.rug.nl/~bplank/metadata/genre_files_updated.html`.

[36] Rashmi Prasad, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, Aravind Joshi, Livio Robaldo, and Bonnie Webber. The penn discourse treebank 2.0 annotation manual. *December*, 17:2007, 2007.

[37] Aili Shen, Meladel Mistica, Bahar Salehi, Hang Li, Timothy Baldwin, and Jianzhong Qi. Evaluating document coherence modeling. *Transactions of the Association for Computational Linguistics*, 9:621–640, 2021.

[38] Bonnie Webber. Genre distinctions for discourse in the penn treebank. 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, pages 674–682, 2009.

[39] Bonnie Webber and Aravind Joshi. Discourse structure and computation: Past, present and future. In Rafael E. Banchs, editor, *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*, pages 42–54, Jeju Island, Korea, July 2012. Association for Computational Linguistics. URL https://aclanthology.org/W12-3205.

[40] Bonnie Webber, Rashmi Prasad, Alan Lee, and Aravind Joshi. The penn discourse treebank 3.0 annotation manual. *Philadelphia, University of Pennsylvania*, 35: 108, 2019.

[41] Joseph Williams and Joseph Bizup. *Style: Lessons in clarity and Grace*. Pearson, 2021.

[42] Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. Discourse-aware neural extractive text summarization. *arXiv preprint arXiv:1910.14142*, 2019.

[43] Peng Xu, Hamidreza Saghir, Jin Sung Kang, Teng Long, Avishek Joey Bose, Yanshuai Cao, and Jackie Chi Kit Cheung. A cross-domain transferable neural coherence model. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 678–687, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1067. URL https://aclanthology.org/P19-1067.

[44] Frances Yung, Kaveri Anuranjana, Merel Scholman, and Vera Demberg. Label distributions help implicit discourse relation classification. In *Proceedings of the 3rd Workshop on Computational Approaches to Discourse*, pages 48–53, 2022.

[45] Wei Zhao, Michael Strube, and Steffen Eger. Discoscore: Evaluating text generation with bert and discourse coherence. *arXiv preprint arXiv:2201.11176*, 2022.

[46] Wei Zhao, Michael Strube, and Steffen Eger. DiscoScore: Evaluating text generation with BERT and discourse coherence. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3865–3883, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.278. URL https://aclanthology.org/2023.eacl-main.278.

Appendix A

PDTB-3 Sense Hierarchy

Table A.1: PDTB-3 Sense Hierarchy collated from Webber et al. [40]. We have left out Level-3 and SpeechAct senses as they are not used in this work. In **Example** column, *arg1* is italicized, **arg2** is bolded and the connective is underlined.

Level-1	Level-2	Definition	Example
TEMPORAL	SYNCHRONOUS	Temporal overlap between the events described by the arguments.	*Knowing a tasty – and free – meal* when **they eat one**, the executives gave the chefs a standing ovation. [wsj 0010]
	ASYNCHRONOUS	One event is described as preceding the other.	*Back downtown, the execs squeezed in a few meetings at the hotel* before **boarding the buses again**. [wsj 0010]
CONTINGENCY	CAUSE	The situations described in the arguments are causally influenced but are not in a conditional relation.	*But service on the line is expected to resume by noon today.* (Implicit=since) **"We had no serious damage on the railroad,"** said a Southern Pacific spokesman. [wsj 1803]
	CAUSE + BELIEF	When evidence is provided to cause the hearer to believe a claim.	With **this sort of sentiment common**, *it's natural for investors to seek out "defensive" investment*. [wsj 0359]
	CONDITION	One argument presents a situation as unrealized (the ANTECEDENT), which (when realized) would lead to the situation described by the other argument (the CONSEQUENT).	*Call Jim Wright's office in downtown Fort Worth, Texas, these days* and **the receptionist still answers the phone, "Speaker Wright's office."** [wsj 0909]

continues on next page →

	NEGATIVE-CONDITION	One argument (the ANTECEDENT) describes a situation presented as unrealized, which if it doesn't occur, would lead to the situation described by the other argument (the CONSEQUENT).	The National Institutes of Health policy would require researchers to *cut financial ties with health-care businesses* or **lose their government money**. [wsj 0975]
	PURPOSE	One argument presents an action that an AGENT undertakes with the purpose of the GOAL conveyed by the other argument being achieved.	There are the strict monetarists, who believe that floating exchange rates free an economy *to stabilize its price level* by **stabilizing the monetary aggregate**. [wsj 0553]
COMPARISON	CONCESSION	An expected causal relation is cancelled or denied by the situation described in one of the arguments.	*It's as if investors, the past few days, are betting that something is going to go wrong* – even if **they don't know what**. [wsj 0359]
	SIMILARITY	One or more similarities between two arguments are highlighted	*Builders get away with using sand* (implicit=similarly) **and financiers junk** . . . [wsj 1849]
	CONTRAST	At least two differences between two arguments are highlighted.	While **the earnings picture confuses**, observers say the *major forces expected to shape the industry in the coming year are clearer*. [wsj 2365]
EXPANSION	CONJUNCTION	Both arguments bear the same relation to some other situation evoked in the discourse	*I can adjust the amount of insurance I want against the amount going into investment*; (Implicit=Conjunction) **I can pay more or less than the so-called target premium in a given year**. [wsj 0041]
	DISJUNCTION	Two arguments are presented as alternatives, with either one or both holding.	If **we want to support students**, *we might adopt the idea used in other countries of offering more scholarships based on something called "scholarship," rather than on the government's idea of "service."*. [wsj 2407]

continues on next page →

EQUIVALENCE	Both arguments are taken to describe the same situation, but from different perspectives	*But the battle is more than Justin bargained for.* (implicit=indeed) "**I had no idea I was getting in so deep**," says Mr. Kaye, who founded Justin in 1982. [wsj 2418]
EXCEPTION	One argument evokes a set of circumstances in which the described situation holds, and the other argument indicates one or more instances where it doesn't.	*Some Japanese operations, such as securities-trading rooms, may be ahead of their American counterparts, he says*, **but** (Implicit=otherwise) "**basically, there's little analysis done on computers in Japan**." [wsj 0445]
INSTANTIATION	One argument describes a situation as holding in a set of circumstances, while the other argument describes one or more of those circumstances.	*Then*, as if **to show that he could play fast as well**, he offered the second movement from Saint-Saens's Sonata for Clarinet, . . . [wsj 0207]
LEVEL-OF-DETAIL	Both arguments describe the same situation, but in less or more detail.	*An enormous turtle has succeeded where the government has failed*: (Implicit = specifically) **He has made speaking Filipino respectable**. [wsj 0804]
MANNER	The situation described by one argument presents the manner in which the situation described by other argument has happened or been done. It answers the *how* question.	Taking a cue from California, *more politicians will launch their campaigns* by **backing initiatives**, says David Magleby of Brigham Young University. [wsj 0120]
SUBSTITUTION	Arguments are presented as exclusive alternatives, with one being ruled out.	Eliminate arbitrage *and liquidity will decline* instead of **rising**, creating more volatility instead of less. [wsj 0118]

Appendix B

CNN Discourse Role Matrix

Table B.1: Coherence evaluation results of CNN Discourse Role Matrix on hyper-parameter settings and discrimination task. Best hyper-parameter setting is in **bold**.

Batch	Filter	Window	Pool	Train Loss	Dev Loss	Dev Acc
32	150	3	6	0.0617	0.9277	90.93
32	150	4	6	0.0508	0.9177	93.82
32	150	5	5	0.0440	1.2807	91.33
32	150	5	6	0.0504	1.0211	94.48
32	200	5	6	0.0586	0.9162	94.09
32	250	5	6	0.0356	1.2543	94.74
64	250	5	6	0.0429	0.9845	92.77
32	150	5	7	0.0575	0.8056	94.74
32	150	7	6	0.0300	1.3591	94.74
32	150	7	7	0.0395	0.7095	95.26
32	150	7	8	0.0572	0.5119	95.13
32	150	8	8	0.0426	0.4144	95.40
32	**150**	**8**	**9**	**0.0438**	**0.7953**	**96.58**
64	150	5	6	0.0442	1.1397	93.17
64	150	6	6	0.0461	0.9982	91.85
64	150	6	7	0.0497	0.9891	93.16
64	150	6	8	0.0530	0.8673	93.69

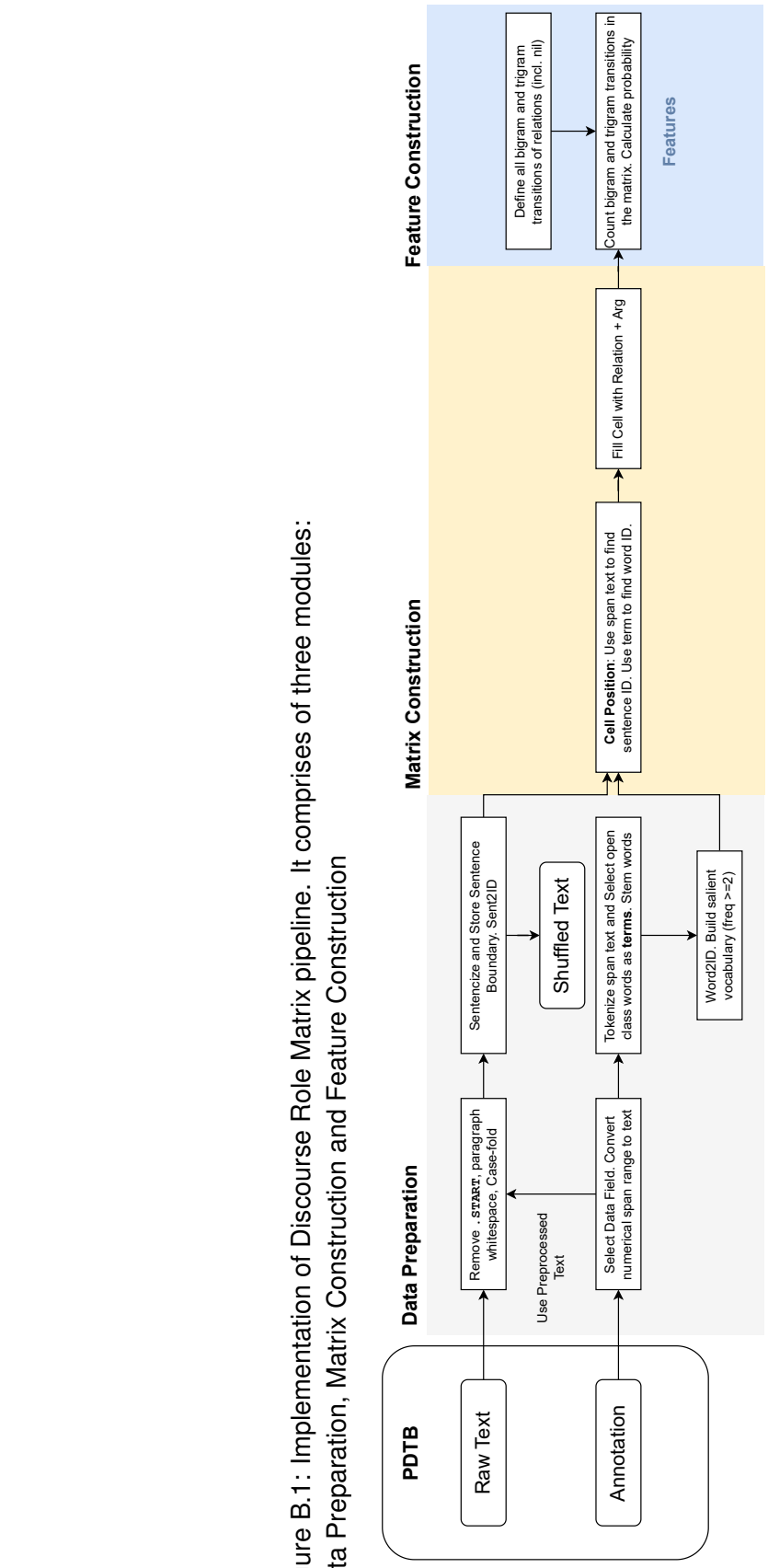


Figure B.1: Implementation of Discourse Role Matrix pipeline. It comprises of three modules: Data Preparation, Matrix Construction and Feature Construction

Appendix C

Relation Patterns in PDTB Genres

As discussed in Section 5.6, ENTREL is most common for HIGHLIGHTS. These documents provide entity heavy information because they summarize financial news in a few sentences. Therefore, we can consider them a more compact version of NEWS. We color these entities in red in the following examples. Articles in this genre involves multiple companies and their actions in the financial market:

> *International Business Machines Corp. – $750 million of 8 3/8% debentures due Nov. 1, 2019, priced at 99 to yield 8.467%.* (ENTREL) **The 30-year non-callable issue was priced at a spread of 57 basis points above the Treasury's 8 1/8% bellwether long bond.** [wsj_0125]

Sometimes, the entity can be even financial instruments like *bonds*.

> *Serial bonds are priced at par* (ENTREL) **to yield from 6.40% in 1991 to 7.15% in 1999**. [wsj_0125]

Unsurprisingly, ENTREL is also most common for NEWS. Similar to HIGHLIGHTS, articles in NEWS focuses on a wide range of entities and narrates their situations. The following example focuses on the owner of *Giant*, a baseball team in San Francisco:

> *He is an avid fan of a proposition on next week's ballot to help build a replacement for Candlestick Park.* (ENTREL) **Small wonder, since he's asking San Francisco taxpayers to sink up to $100 million into the new stadium**. [wsj_0126]

EXPANSION is most common for ESSAY. Articles in this genre often discuss current issues with opinions from journalists. This often requires providing details to a topic and carrying multiple points across. The example below is the beginning of an ESSAY and has already contained two EXPANSION relations. In the first relation, ARG2 provides detail to why the rationale is clear. In the second relation, ARG2 continues the point in ARG1.

The rationale for responding to your customers' needs faster than the competition can is clear: **Your company will benefit in terms of market share, customer satisfaction and profitability**. In fact, managers today are probably more aware of speed as a competitive variable than ever before. [wsj_0562]

The rationale for responding to your customers' needs faster than the competition can is clear: Your company will benefit in terms of market share, customer satisfaction and profitability. In fact, **managers today are probably more aware of speed as a competitive variable than ever before**. [wsj_0562]