TOKYO INSTITUTE OF TECHNOLOGY

DOCTORAL THESIS

Parsing Argumentative Structure in English-as-a-Foreign-Language Learner Essays

Author:
Jan Wira Gotama PUTRA

Supervisors:
Prof. Takenobu TOKUNAGA
Prof. Simone TEUFEL

A thesis submitted in fulfilment of the requirements
for the degree of Doctor of Engineering

in the

School of Computing
Department of Computer Science

August 17, 2021

Abstract

Argument mining (AM) aims to explain how individual argumentative discourse units (e.g., sentences and clause-like segments) relate to each other and what roles they play in the overall argumentation. The automatic recognition of argumentative structure is attractive because it benefits various downstream tasks, such as text assessment, text generation, text improvement and summarisation. Existing studies focused on analysing well-written texts provided by proficient authors. However, the majority of English speakers worldwide are non-native, and their texts are frequently poorly structured, particularly if they are still in the learning phase. Yet there is no prior study that addresses the analysis of argumentative structure specifically in non-native texts. This thesis presents an argument mining study on English-as-a-foreign-language (EFL) essays of intermediate quality. It focuses on three tasks: (i) constructing a new language resource for training an AM system for EFL essays, (ii) argumentative structure parsing, and (iii) improving the quality of essays by reordering sentences.

Concerning the first task, I present the first corpus of annotated EFL essays, together with a specially designed annotation scheme. The resulting corpus, called "ICNALE-AS2R," contains 434 essays written by learners from numerous Asian countries, along with two types of manual annotation: annotation of their argumentative structure and reordering annotation. The second type of annotation indicates one way how the sentences could be reordered, resulting in an essay of overall higher quality. The annotated corpus is particularly useful for the education domain, as the argumentative structure annotation can reveal learners' argumentation-related problems, and the reordering annotation shows one way to improve the essay so that it more closely resembles a native-level production. My argument annotation scheme is demonstrably stable, achieving good inter-annotator agreement and near-perfect intra-annotator agreement. The annotated corpus comes with some additional methodological and technical contributions. First, I propose a set of novel document-level agreement metrics that can quantify structural agreement from various argumentation aspects, thus, providing a more holistic analysis of the quality of the argumentative structure annotation. The metrics are evaluated in a crowdsourced meta-evaluation experiment, achieving moderate to good correlation with human judgements. Second, all corpus annotation is performed by an external expert annotator using my newly developed web-based annotation tool TIARA. It provides versatile visualisation for structural annotation and reduces clutter in the display. The tool is easily customisable via a configuration script. Apart from its use as an annotation tool, it is also designed to support the educational use case of learning-to-write.

I also conduct a secondary evaluation using three third-party professional essay assessors to confirm whether the reordered version of essays is indeed better than the original one, in the light of the likely inherent subjectivity of the quality of sentence arrangement. The assessors exhibited low agreement with each other in their judgement. My conclusion from this experiment is that the reordered version of essays in the ICNALE-AS2R corpus cannot be treated as the single correct one. Nevertheless, the evaluation confirms that the reordering operation improves essays' quality to some degree.

Concerning the second task, I propose deep learning models to parse the argumentative structure in EFL essays in two steps: a sentence linking and a relation labelling step. The experimental results show that a biaffine model combined with sentence-BERT encoder performs best in the sentence linking task, whereas fine-tuned BERT model shows the best results in the relation labelling task. I also evaluate the parser on a cross-domain setting, where training is performed on both in-domain (EFL essays) and out-domain (reordered essays), and evaluation is performed on the in-domain test data. I observe that the best cross-domain system achieves 94% of the in-domain system in terms of end-to-end performance. I conclude that the best training regime for my parser might mix well-written texts with less well-structured texts. I identify the sentence linking task as the main challenge; the model seems to stumble when confronted with the hierarchical nature of arguments. To improve the sentence linking performance, I extend the biaffine model using a multi-task learning setup to provide a richer supervision signal. I also propose multi-corpora training with a selective sampling strategy to increase the available amount of training data. These two strategies consistently improved the sentence linking performance on all evaluation aspects, resulting in a 15.8% increase in the F1-macro score for individual link predictions, amongst other improvements.

Concerning the third task of providing discourse level feedback for language learners, I propose a new computational task of sentence reordering. Given a sequence of sentences, presumably in sub-optimal order, the goal is to rearrange them into a well-structured text. I develop a sentence reordering system based on the results of an earlier step of argumentative structure analysis. The reordering task is formulated as a tree-traversal problem consisting of two steps: a pairwise ordering constraint classification between argumentatively related sentences, followed by a tree traversal step which generates the final output. Experimental results show that the system can perform the reordering operation selectively, that is, it reorders sentences when necessary and retains the original input order when reordering would not result in an improvement. The usefulness of argumentative structure information is confirmed in an ablation study where the system's performance on three types of input is compared: automatically generated structures, gold standard structures and random structures. I found that the factor that would boost reordering performance most would be a further improvement in the argumentative structure parsing.

Overall, this thesis contributes towards providing automated discourse-level analysis and feedback to language learners. Both the argumentative structure visualisation and reordering recommendation facilitate the learning process, particularly in analysing and revising texts.

Acknowledgements

I would like to express my highest gratitude to my supervisor, Prof. Takenobu Tokunaga, for ideas, discussions, directions, continuous support and valuable advice throughout my graduate study. I would also like to thank my co-supervisor, Prof. Simone Teufel, for the collaboration and help throughout my doctoral program. I would not be who I am today without the amazing education from them. My supervisors always took the time for me however busy they might have been, and I am very thankful for this.

I am also grateful to Prof. Atsushi Fujii for comments and feedback during laboratory-wide plenaries. I am thankful for Prof. Jun Miyazaki, Prof. Naoaki Okazaki, Prof. Suguru Saito and Prof. Tsuyoshi Murata for finding the time to evaluate my thesis. Prof. Yasuyo Sawaki, Kana Matsumura, Dolça, Ogawa and Michael have been very kind in providing me with feedback for the annotation guidelines as well as for the annotation tool TIARA. Ms. Matsumura especially played a big role in my annotation project and the development of TIARA into its current stage.

I am particularly grateful for the financial assistance from the Japan Society for the Promotion of Science through "JSPS Doctoral Fellowship for Young Scientists" Program and KAKENHI Grant Number 20J13239, and Japan Ministry of Education, Culture, Sports, Science and Technology through MEXT Scholarship. This study would have not been possible without the support from JSPS and MEXT. I am particularly indebted to Prof. Hitoshi Nishikawa for his guidance when writing my JSPS funding proposal.

I am thankful to my colleagues in the Tokunaga-Fujii laboratory for the constructive feedback I received on numerous occasions. They have also been very helpful in assisting with daily-life matters, especially Yamada, Yuni, Kimura, Nakayama, Arief and Sidik. As an international student, I was not accustomed to life in Japan, and their help meant a lot. Finally, I would also like to thank my girlfriend Natasha, and a long list of friends for company, comfort and always keeping my sanity in check.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Argumentation is an activity to persuade a person to a particular action or view. It has been practised for over two millennia since Aristotle's (Aristotle and Kennedy, 1991). Nowadays, argumentation is ubiquitous in everyday discourse, for example, in the form of debates, court proceedings, essays and news editorials. Argumentation is also at the centre of scientific practice, in building an accurate description of nature and how it operates. It is no wonder that our education system emphasises the importance of argumentation skills (cf. USA Common Core, CEFR).

Argument mining (AM) is an emerging area in computational linguistics (CL) that addresses the automatic analysis of argumentation. It aims to explain how argumentative discourse units (ADU; e.g., sentences and clause-like segments) function in the text and relate to each other, forming an argument as a whole (Lippi and Torroni, 2016). Argumentative structure is particularly useful for computational models of argument and reasoning engines. The ubiquity of argumentation in daily life prompted AM studies in various areas, such as in the legal domain (Ashley, 1990), in news (Al-Khatib et al., 2016a) and in education (Stab and Gurevych, 2017).

It is common in AM to use well-written texts by proficient authors, including in Ashley (1990), Mochales and Moens (2001) and Peldszus and Stede (2016), among others. However, it is well-known that texts written by students suffer from several textual problems because they are still learning how to write effectively. It has been observed that student texts often require improvement at the discourse level, where persuasiveness and content organisation are concerned (Carlile et al., 2018; Zhang and Litman, 2015). Worse still, non-native students' writings are less coherent and less lexically rich, and exhibit less natural lexical choices and collocations (Rabinovich et al., 2016; Silva, 1993). There are more non-native speakers of English than native speakers in the world (Fujiwara, 2018), and yet there is no specific prior study in AM focusing on non-native texts.

Writing an effective argumentation is difficult, and even more so if it has to be expressed in a non-native language (Bacha, 2010). The analysis of argumentative structure can facilitate learning. Particularly, it enables students and teachers to flag argumentation-related problems in texts, and subsequently, formulate ways to improve the texts (cf. Section 2.1.2). In this thesis, I propose an application of AM for non-native speakers of English with intermediate-level proficiency. I analyse the argumentative structure in essays written by English-as-a-foreign-language (EFL) learners from various Asian countries. I also propose to improve the essay quality by sentence rearrangement based on the structural analysis.

The following example shows an argumentative essay written by a Chinese student in response to the prompt "*Smoking should be banned at all the restaurants in the* *country*"[1] (ICNALE (Ishikawa, 2013; 2018) essay "W_CHN_SMK0_275_B2_0_EDIT"; I refer to this essay as "high-quality example"):

> (S1)*It is universally recognised that smoking does much damage to human health and that second-hand smoking causes even more serious effects to the people around smokers.* (S2)*According to the statistics shown in newspapers, about five percent of deaths are related to second-hand smoking.* (S3)*Due to the terrible effects of public smoking, I hold the opinion that smoking should be banned in any public restaurants across the country.* (S4)*By doing so, one of the most important favourable effects is that diseases related to smoking, such as lung cancer, can be cut down significantly.* (S5)*The ban contributes a lot to the creation of a healthy dining environment for people who frequently eat outside, which takes up a large proportion of the whole population.* (S6)*Secondly, prohibiting smoking in some public areas contributes greatly to the development of social culture and ideology.* (S7)*Like drunken driving, which poses threats to citizens' safety, smoking in public does harm to others' health.* (S8)*Such behaviour is against our goal of establishing a harmonious society.* (S9)*In addition, the forceful act of a complete ban raises the awareness of the harm of smoking among the general public.* (S10)*More and more smokers will gradually get rid of this bad habit for the interest of their own health.* (S11)*To conclude, it is high time for us to take strong measures to put an end to this smoking era.* (S12)*A complete change to the legal system regarding the smoking issue is necessary for the final settlement of this social problem.*

Successful argumentative essays such as this example typically introduce the discussion topic (here, S1–S2), state their *major claim* (sometimes called *main stance* or *main claim*) on the topic (S3), support their stance by presenting reasons from various perspectives (S4–S10), and then provide a conclusion (S11) (Bacha, 2010; Silva, 1993). The author of the previous example was at upper-intermediate to advanced proficiency and had a TOEFL iBT Score of 98. However, not all EFL students possess the skill to write at this level.

Consider the following essay, which was written in response to the prompt "*It is important for college students to have a part-time job*", by an Indonesian student with a lower-intermediate to intermediate proficiency (ICNALE essay "W_IDN_PTJ0_050 _A2_0_EDIT"; I refer to this essay as "intermediate-quality example"):

> (S1)*The costs students incur on campus are not small; every month can cost up to a million for meals, transportation, books, and cigarettes for smokers.* (S2)*The income of a parent who is an entrepreneur can sometimes cover this amount, but other parents need more than one income.* (S3)*Every student wants to cover the cost when they live far away from their parents.* (S4)*Some students who have many necessary payments on campus need to look for money by themselves, so they usually work at a cafe, do car washing, work as a newspaper agent, or work at an Internet rental shop.* (S5)*But sometimes, they have problems dividing their time, and they sometimes ignore their assignments from college.* (S6)*But, they feel proud they can complete part of their costs of college without asking their parents.* (S7)*If all students do this, surely all parents would feel proud but they must not complete all of the necessary things.* (S8)*Thus, if sometimes the parents' income is not enough to pay the campus costs, we have to get money by ourselves to cover everything from books to the boarding house without asking our parents.* (S9)*In my opinion, a part-time job helps students support their financial problems and I agree that students should work part time.*

In this thesis, I work on essays of intermediate quality, such as this second example; this essay differs from the previous high-quality example in at least two respects. First, the intermediate-quality example does not adhere to the typical English

[1] A prompt is a question or a sentence used to elicit an argumentative response. Note that students are usually asked to write their essays in a stand-alone fashion, that is, under the assumption that the prompt is not considered as part of the essay and therefore not read together with it.

argumentation development strategy. For instance, the discussion topic is not introduced, and the major claim (underlined) is given at the end of the essay rather than at the beginning. This contrasts with a more straightforward structure in the previous high-quality example, which presented the major claim right at the beginning. Second, the intermediate-quality example presents the argument only from a single viewpoint (arguing in favour of part-time jobs for financial reasons), whereas the high-quality example considers two (arguing in favour of banning smoking for health and cultural reasons). We can observe that essays written by intermediate-level writers are likely to pose more challenges to any computational treatment because of their poorer structure.

Rearrangement of sentences may improve the intermediate-quality example. For instance, we can make the text more straightforward by placing S9 at the beginning of the text, which is shown as follows.

> (S9)*In my opinion, a part-time job helps students support their financial problems and I agree that students should work part time.* (S1)*The costs students incur on campus are not small; every month can cost up to a million for meals, transportation, books, and cigarettes for smokers.* (S2)*The income of a parent who is an entrepreneur can sometimes cover this amount, but other parents need more than one income.* (S3)*Every student wants to cover the cost when they live far away from their parents.* (S4)*Some students who have many necessary payments on campus need to look for money by themselves, so they usually work at a cafe, do car washing, work as a newspaper agent, or work at an Internet rental shop.* (S5)*But sometimes, they have problems dividing their time, and they sometimes ignore their assignments from college.* (S6)*But, they feel proud they can complete part of their costs of college without asking their parents.* (S7)*If all students do this, surely all parents would feel proud but they must not complete all of the necessary things.* (S8)*Thus, if sometimes the parents' income is not enough to pay the campus costs, we have to get money by ourselves to cover everything from books to the boarding house without asking our parents.*

The reordered essay previously mentioned is still not perfect even though it has improved in quality; for instance, it still only argues from a single viewpoint. That is not a problem, however, as we could not hope to reach a perfect essay without much deeper understanding which currently eludes all CL. Also, it is a step-wise improvement that is particularly helpful in educational situations.

Most of the existing approaches in AM use an annotated corpus to train supervised machine learning models. To this end, I see the creation of an annotated EFL corpus as the first step towards an automatic argumentative structure analysis and improvement system. Such a corpus in and of itself can already support the theoretical and practical teaching of how to argue in the EFL context. Kaplan (1966) introduced the teaching strategy based on *contrastive rhetoric*, where the idea is to show EFL students the differences between the structures of their writings and native (and thus presumably "good") writings. My corpus can be used for theoretical studies in contrastive rhetoric, and it can also be used practically in the classroom today. It can be even more effective if combined with argumentative structure visualisation (Cullen et al., 2018; Matsumura and Sakamoto, 2021).

1.2 Contributions

This thesis can largely be divided into three tasks: (i) constructing a new language resource for training an AM system for EFL essays, (ii) argumentative structure parsing, and (iii) improving the quality of essays by reordering sentences. The following list provides an overview of my contributions per task.

Constructing a new language resource for training an AM system for EFL essays

- **Novel annotation scheme** (Section 3.1) – I present a new annotation scheme for argumentative structure (AS) and sentence reordering (SR) in EFL essays. The argumentative structure annotation consists of two steps: (AS-1) argumentative component identification and (AS-2) argumentative structure prediction. The sentence reordering annotation consists of two steps: (SR-1) rearranging sentences in the texts followed by (SR-2) a text repair procedure to adjust connectives and referring expressions.

- **Annotation tool TIARA**[2] (Section 3.2) – A new annotation study often has specific, so far unserved, needs, and my study is no exception. To this end, I developed a new client-side tool, TIARA, to support my annotation needs. While the tool is originally invented to support my annotation scheme, it is also designed to be useful for general discourse structure annotation and educational purposes.

- **Novel agreement metrics** (Section 3.3) – I propose a novel structure-based metric, called "mean agreement in recall" (MAR), for the calculation and better interpretation of inter-annotator agreement for argumentative structure analysis. The metric contains several variants that differ in how the calculation units are defined. This thesis also presents a meta-evaluation study via crowdsourcing, quantifying the reliability of the proposed metric in comparison with existing ones.

- **ICNALE-AS2R**[3] **corpus** (Section 3.4) – I present the first corpus of EFL texts annotated with argumentative structure and sentence reordering. It contains 434 essays written by English learners from various Asian countries. Inter-annotator and intra-annotator agreement studies are performed that show a reasonable level of agreement for argumentative structure annotation, considering the difficulty of the task. I also perform a secondary human evaluation to confirm whether the reordered version of the essays is indeed better than the original ones.

Argumentative structure parsing (Chapter 4) – I parse the argumentative structure in EFL essays in two steps: (1) a *sentence linking* step where I identify related sentences that should be connected, forming a tree structure, and (2) a *relation labelling* step, where I label the relationship between linked sentences. Several deep learning models are evaluated to address each step. The models' performance is not only evaluated based on individual links but also structural properties. This provides more insights into the models' ability to learn different aspects of the argumentative structure. This thesis also experiments on multi-task learning and multi-corpora training strategies to provide a richer supervision signal for such a structural modelling task. My proposed auxiliary tasks help the model to learn the role of each sentence in the argument hierarchy. The multi-corpora training with a selective sampling strategy helps increasing training data size while ensuring that the model indeed learns the desired target distribution.

[2] TIARA stands for "**T**ool for **I**nteractive **AR**gument **A**nnotation." This thesis describes TIARA at version 2.0, which is an extended version of my LREC2020 paper. The tool is publicly available at https://wiragotama.github.io/TIARA-annotationTool/

[3] ICNALE-AS2R stands for "ICNALE annotated with **A**rgumentative **S**tructure and **S**entence **R**eordering." It is publicly available at https://www.gsk.or.jp/en/catalog/

Improving the quality of argumentative essays by reordering sentences

- **Novel task** – I propose a novel computational task of sentence reordering. Given an essay as a sequence of sentences, the goal is to rearrange sentences in the text so that it results in a well-structured text. However, the original order should be retained if the input essay is already well-structured.

- **Automatic sentence reordering model** (Chapter 5) – I propose an approach to reorder sentences based on the analysis of argumentative structure. My sentence reordering module consists of two steps. The first is a pairwise ordering constraint classification (POCC). For each pair of sentences connected by argumentative relations, the goal is to decide which sentence should come first in linear order. The second is a traversal step where I generate output texts based on the argumentative structure that has been augmented with pairwise ordering information.

1.3 Publication Record

Several parts of this thesis have already been published or accepted for publication at peer-reviewed international journals, conferences and workshop proceedings. I list all publications below and indicate the chapters and sections of this thesis which build upon them.

- Jan Wira Gotama Putra, Simone Teufel, and Takenobu Tokunaga (2021). Annotating Argumentative Structure in English-as-a-Foreign-Language Learner Essays. In: *Natural Language Engineering* [*in press*]. DOI: https://doi.org/10.1017/S1351324921000218 (Section 1.1, 2.1, 3.1, 3.3, 3.4)

- Jan Wira Gotama Putra, Simone Teufel, and Takenobu Tokunaga (2021). Parsing Argumentative Structure in English-as-Foreign-Language Essays. In: *Proceedings of Sixteenth Workshop on Innovative Use of NLP for Building Educational Applications*. Online: Association for Computational Linguistics, pp. 97-109. URL: https://aclanthology.org/2021.bea-1.10 (Section 2.3, 4.1)

- Jan Wira Gotama Putra, Simone Teufel, Kana Matsumura, and Takenobu Tokunaga (2020). TIARA: A Tool for Annotating Discourse Relations and Sentence Reordering. In: *Proceedings of 12th International Conference on Language Resources and Evaluation (LREC)*. Marseille, France: European Language Resources Association (ELRA), pp. 6914-6922. URL: https://aclanthology.org/2020.lrec-1.854. (Section 2.1.2, 2.2.3, 3.2)

I describe related works concerning the fields related to this thesis in the following chapter.

Chapter 2

Related Work

This chapter provides an overview of related studies in argument mining, including existing corpora, annotation tools for discourse annotation and computational models for argumentative structure parsing and sentence ordering.

2.1 Argument Mining

Discourse theories aim to explain how discourse units (e.g., sentences and clause-like segments) relate to each other and what roles they play in the overall discourse (Grosz and Sidner, 1986; Mann and Thompson, 1988). The automatic recognition of discourse structure is attractive as it would benefit various downstream tasks, such as text assessment (Feng et al., 2014; Wachsmuth et al., 2016), text generation (Al-Khatib et al., 2017; Hovy, 1991; Yanase et al., 2015) and summarisation (Teufel and Moens, 2002; Yamada et al., 2019).

Different types of discourse structure have been proposed over the years (Webber et al., 2012). Rhetorical Structure Theory (RST) modelled the relations between adjacent discourse units, which form a tree (Mann and Thompson, 1988). The Penn Discourse Treebank project (Prasad et al., 2008) analysed local discourse relations and the discourse markers that signal the relations. Wolf and Gibson (2005) observed that texts often contain various kinds of crossed dependencies between sentences as well as nodes with multiple parents. As a result, they modelled text as a graph. In contrast, Hearst (1997) segmented text into a linear sequence of thematically coherent topics.

While the theories mentioned above are designed to be general across genres, discourse structure analysis is also often tailored to the target text genre and the research goal. Here, I approach the discourse structure analysis from the argumentation perspective as I am trying to analyse argumentative essays written by EFL students.

Argumentative structure annotation consists of two main steps (Lippi and Torroni, 2016). The first is *argumentative component identification* (ACI), which determines the boundaries of ADUs, and then differentiating them into argumentative and non-argumentative components. Argumentative components (ACs) function to persuade readers, whereas non-argumentative components (non-ACs) do not (Habernal et al., 2014). Non-ACs are often excluded from further processing as they do not contribute to the argumentative structure. ACs can be further classified according to their roles in argumentation, for example, *proponent* and *opponent* (Peldszus and Stede, 2016). These roles can be extended or tailored according to the application context. For example, Stab and Gurevych (2014) used *major claim*, *claim* and *premise* for persuasive essays, whereas Al-Khatib et al. (2016a), working on news articles, differentiated between *common ground*, *assumption*, *testimony*, *statistics*, *anecdote* and *other*.

The second step is *argumentative discourse structure prediction*. This step establishes labelled links from *source* to *target* ACs to form the text structure, which can be a tree (Stab and Gurevych, 2014) or a graph (Kirschner et al., 2015; Sonntag and Stede, 2014). Typically, all ACs must be connected to the structure, while all non-ACs remain unconnected. Links (also called edges) can be directed (Stab and Gurevych, 2014) or undirected (Kirschner et al., 2015). The links are then labelled according to the relationship between the source and target ACs, for instance, using the labels SUPPORT and ATTACK (Stab and Gurevych, 2014). Similar to the variations in AC labels, previous studies in argument mining have also tailored relation labels to specific research goals and needs. Kirschner et al. (2015), for example, proposed the DETAIL relation that roughly corresponds to the ELABORATION and BACKGROUND relations in RST (Mann and Thompson, 1988). Skeppstedt et al. (2018) observed another frequent relation, namely RESTATEMENT, which applies in those cases when an important part of the argument, such as a major claim, is repeated and summarised in strategically important places, such as at the end of the essay.

2.1.1 Argumentative Structure and Text Quality

Writing coherent argumentative texts requires reasoning and effective framing of our opinions. A coherent argumentative text has to contain the desired argumentative elements; ideas should be clearly stated, connected to each other and supported by reasons. The ideas should also be logically developed in a particular sequencing, such as by time or importance, and accompanied by appropriate discourse markers. Only then can the writing ultimately communicate the desired ideas as a whole (Bacha, 2010; Bamberg, 1983; Blair, 2012; Garing, 2014; Hofmockel et al., 2017; Reed and Wells, 2007; Silva, 1993).

The idea that there is a close connection between argumentative structure (and discourse structure in general) and text quality has been applied in text assessment studies. Persing et al. (2010) provided an automatic organisation score based on the patterns of rhetorical-category transitions between sentences. Wachsmuth et al. (2016) also used a similar strategy when scoring various aspects of argumentation. Discourse structure also correlates with text coherence, and various coherence models have been developed that rely on this interaction. For example, Lin et al. (2011) and Feng et al. (2014) measured text coherence based on discourse-relation transition bigrams.

It has been argued that discourse structure forms a plan to order sentences (Hovy, 1991). Hence, many natural language generation (NLG) studies tried to produce coherent and persuasive texts by following certain discourse patterns. Yanase et al. (2015), for instance, ordered sentences in debate texts using a "*claim-support*" structure. In the claim-support structure, the first sentence describes an opinion, which is followed by support sentences expressing reasons for the opinion. On the other hand, Al-Khatib et al. (2017), working on news editorial texts, assumed that a persuasive argument can be built based on fixed argumentation strategies; they identified several such argumentation strategies in the form of common patterns of N-grams over component types. In another NLG approach, El Baff et al. (2019) pooled text pieces from many different texts and then generated text as a slot-filling process. Their system proceeded by selecting one discourse unit after the other from the pool if it satisfied the rhetorical function needed in the template. In the final output, only a small proportion of all available sentences were used.

The studies previously mentioned approached sentence order from the rhetorical transition viewpoint, but there are also studies that used a relational viewpoint. Grosz and Sidner's (1986) theory of text coherence stipulates that sentence order mirrors the intentional structure of discourse, which is represented as relationships between sentences. Two important factors decide the order of sentences in texts: *dominance* and *satisfaction-precedence*. The dominance factor concerns the hierarchy of ideas, that is, which sentence presents a more general or specific idea. The satisfaction-precedence factor concerns the pairwise ordering relation between ideas. For example, if a reason R (source) ideally follows the opinion O (target) it supports, it can be said that O has to be *satisfied* before R, that is, O has to appear before R in linear order. This means that we may infer the coherent order of sentences upon structural analysis because the argumentative structure represents the dominance hierarchy of sentences (Hovy, 1991; Webber and Joshi, 2012). Despite a promising approach, the use of argumentative relations for ordering sentences is still under-explored.

2.1.2 The Role of Argumentative Structure Analysis in Teaching

Many existing studies in the CL community have attempted to correct spelling and grammatical errors (e.g., Bryant and Briscoe, 2018; Han et al., 2006; Hirst and Budanitsky, 2005; Yuan and Briscoe, 2016), but studies at the discourse and argumentation levels are still limited (Strobl et al., 2019). Teaching students how to argue effectively can be difficult, particularly if the medium of expression is not their first language (Bacha, 2010; Silva, 1993). Cullen et al. (2018) showed how teaching to argue can be supported by annotating the implicit argumentative structure. They performed a controlled study where one group of students were taught to annotate argumentative structure visually, whereas the control group was taught traditionally, that is, through written or verbal explanation. They found a larger increase in the visually taught group than in the control group when measuring the improvement of both groups in a logical reasoning test before and after the teaching sessions, suggesting that learning to annotate arguments led to improvements in students' analytical-reasoning skills.

The argumentative structure analysis allows writers to check completeness (*are all necessary parts there?*) and coherence (*do relations among parts make sense?*) (Bobek and Tversky, 2016). Such analysis also facilitates discussions between students and instructors about text structure because students can share their interpretations and writing plans through the annotated structure. This allows instructors to quickly identify gaps in students' understanding of the learning material and then provide relevant feedback to the students (Cullen et al., 2018). For example, instructors may check whether an argument is balanced and contains the necessary material (Hsin and Snow, 2020; Matsumura and Sakamoto, 2021) or, if not, encourage a student to find new relevant material and to incorporate it into the essay. In this thesis, I am more interested in a situation where the necessary material has already been provided by the student in this thesis, but it is possibly in a sub-optimal order. Rather than organising a text from scratch, I therefore am interested in reorganisation of sentences in the text, an aspect which EFL students often struggle with.

Studies in *contrastive rhetoric* investigate how students' first language might influence their writings in the second language. Many studies found that non-native speakers tend to structure and organise their texts differently from native speakers (Connor, 2002; Johns, 1986; Kaplan, 1966; Silva, 1993). If EFL students use the customs, reasoning patterns and rhetorical strategies of their first language when writing in the second language, there is a danger that the different organisation of ideas can violate the cultural expectations of native speakers (Kaplan, 1966). For example, it is sometimes observed that reasons for a claim are presented before the claim in writings by Asian students, which is not common in Anglo-Saxon cultures (Johns, 1986; Silva, 1993). This can result in a situation where writings by Asian students may appear poorly organised in the eyes of native readers. The instructional approaches for argumentation strategies also vary among cultures. For example, Liu (2005) found that American instructional approaches encourage the consideration of opposing ideas, whereas the Chinese approaches describe the importance of analogies, and epistemological and dialogical emphases. Therefore, studies argued that EFL students need specific instructions to account for cultural differences in L1 and L2 (Bacha, 2010; Connor, 2002; Kaplan, 1966; Silva, 1993).

Argumentative structure analysis helps EFL students to understand and bridge the cultural gaps between writing strategies in their native languages and English, but no AM study before this thesis has provided support for this specific task. On top of that, reordering recommendations may also serve as feedback for students, enabling them to know how to improve their texts to satisfy the argumentation-development organisation expected by native speakers (Britt and Larson, 2003; Invanic, 2004; Silva, 1993; Stab and Gurevych, 2014). Altogether, these analyses enhance learners' skills in self-monitoring and revising texts.[1]

2.2 Corpora

2.2.1 Argument Annotated Corpora

There exist corpora covering various aspects of argumentation analysis, for instance, argument strength (Persing and Ng, 2015), type of reasoning (Reed et al., 2008) and argumentative relations (Kirschner et al., 2015). Considering our target domain, the most relevant corpora for the current work are the *microtext corpus* (MTC) by Peldszus and Stede (2016) and the *persuasive essay corpus*[2] (PEC) by Stab and Gurevych (2014; 2017).

MTC is a collection of 112 short texts that were written in response to various prompts. The texts contain roughly five ACs per text, with no non-ACs present. Each text is centred around a single major claim, and other ACs act as *proponent* (defending the major claim) or *opponent* (questioning the major claim). All components form a single tree structure, whereby the links can be of three types: SUPPORT, REBUTTAL and UNDERCUT. The texts in the original study were written in German and then translated into English, but in a follow-up study (Skeppstedt et al., 2018), crowd workers were employed to write in English. Efforts were made to create argumentation of the highest possible quality; texts with possible lower quality argumentation were removed.

The 402 texts in PEC are longer than those in MTC with their average length of 18 sentences. The PEC texts contain both ACs and non-ACs, on average, 15 ACs and 3 non-ACs. The texts, which are written in English, were randomly collected from essayforum.com, an online forum where students can receive feedback on their writing. ACs are subdivided into *major claim*, *claim* and *premise*, with link types SUPPORT and ATTACK, forming a tree in which the major claim acts as the root (level 0). Supporting or attacking claims, which are marked as such, then follow in level 1, which

[1] Self-monitoring concerns the skills to think about the effectiveness of writing strategies used, and revising skills concern the ability to evaluate and improve texts (Strobl et al., 2019).

[2] The authors use the term "persuasive" as synonymous with "argumentative".

in turn is followed by premises at even deeper levels (≥ 2). This means that the discourse function is doubly marked in this scheme: by the level of an AC in the hierarchy and by an explicit labelling of ACs.

Neither of these corpora is appropriate for my goal. The authors of the MTC were assumed to be fully competent in the creation of argumentative texts or the texts were filtered so that only high-quality texts remain. PEC is also not suitable for my research purpose because it does not distinguish between native and non-native speakers and gives no information about the (assumed or observed) quality of the essays. These corpora also do not contain discourse-level improvements for the essays, something that is needed in my study. Only one such parallel corpus exists to the best of my knowledge. Zhang et al. (2017) constructed a corpus of essay drafts and their revisions. However, content modification was allowed during the revision process, for example, by deleting or adding a new sentence. This thesis aims to improve EFL essays using the same content; therefore, I also could not use Zhang et al.'s corpus.

To address all of the above concerns, I propose a custom-made corpus in this thesis. I specifically sample intermediate level texts from an English learner corpus.

2.2.2 English Learner Corpora

English learners' writings have been extensively studied in the CL community, particularly in the field of automated essay scoring (Shermis et al., 2006) and native language identification (Koppel et al., 2005).

The International Corpus of Learner English is a collection of 6,805 English essays written by undergraduate students of 16 non-English mother tongues. Most of the essays (91%) are argumentative (Persing et al., 2010). However, these essays were written by upper-intermediate and advanced level learners. Therefore, it is outside the scope of this thesis.

The TOEFL11 corpus is a collection of 12,100 English essays written by non-native speakers of 11 non-English mother tongues, ranging from 2 to 876 words (Blanchard et al., 2013). The essays were written in response to eight prompts, and authored by TOEFL iBT® 2006–2007 test takers. The corpus was originally compiled for the native language identification experiment, but it also includes holistic three-grade (low/medium/high) score levels to support automated essay scoring research.

The International Corpus Network of Asian Learners of English (ICNALE) is a collection of 5,600 argumentative essays written by college students from 10 Asian countries (Ishikawa, 2013).[3] The vast majority of these are written by non-native English speakers of intermediate proficiency, although 7.1% of the essays are written by Singaporeans, for whom English is typically the first language. The corpus was originally designed for contrastive interlingua analysis, and the essays were collected in a controlled setting. ICNALE essays contain 200–300 words and are written in response to two prompts: (1) "*It is important for college students to have a part-time job*" and (2) "*Smoking should be completely banned at all the restaurants in the country.*"

The ICNALE corpus is more suitable for my research compared with other corpora considering the original corpus design, that is, contrastive analysis. It also offers a subset of 640 essays that have been scored with respect to five aspects, namely,

[3] http://language.sakura.ne.jp/icnale/

content, organisation, vocabulary, language use and mechanics,[4] which are combined into a total score in the range of $[0, 100]$.[5] This subset has additionally been corrected in terms of grammatical and mechanical aspects (Ishikawa, 2018). An important aspect of my work is to treat students' argumentation skills as separate from their lexical and grammatical skills, following Skeppstedt et al. (2018). Thus, the 640 subset makes the ICNALE corpus particularly appealing for my project. Hence, I take this subset as the starting point of my study, and strategically sample intermediate-quality essays based on scores provided by professional ICNALE assessors.

2.2.3 Annotation Tool

Many annotation tools have been developed in the CL community. Among them, BRAT (Stenetorp et al., 2012) is relatively popular as it supports a wide range of annotation tasks. It also offers annotation visualisation and collaboration features. BRAT also has been used for AM in the study of Stab and Gurevych (2017). Built in the same spirit as BRAT, WebAnno (Yimam et al., 2013) offers additional management and monitoring features. These tools are easy to customise, offering the flexibility to accommodate a wide range of annotation tasks. However, BRAT and WebAnno were originally designed for morphological, syntactic and semantic annotations, that is, rather local word or phrase-level annotations. While they support link display and could thus theoretically be used for discourse annotation, the visual display of links appears as drawn directly on top of text. This style of display has already been identified by others as a source of confusion for argumentative structure annotation projects (Kirschner et al., 2015). PDTB annotator (Prasad et al., 2008) also falls into the class of annotation tools designed for local relations. When it comes to the display of larger-scale hierarchical or graphical structure of discourse, this falls entirely outside the purview of these tools.

Discourse and argumentative structures are inherently hard to visualise without either cluttering the display or confusing the annotators. To this end, tools have also been developed which specifically aimed at visualising a larger-scale and more global structure, for example, RSTTool (O'Donnell, 2000), TreeAnno (De Kuthy et al., 2018), OVA (Janier et al., 2014), DiGAT (Kirschner et al., 2015) and GraPat (Sonntag and Stede, 2014).

There are four features required for a discourse structure annotation tool: (a) discourse unit segmentation, (b) unit categorisation, (c) establishing links between discourse units and (d) labelling the links. It is commonly assumed that all units are connected to the structure. However, ADUs are used selectively in the argumentative structure annotation. There is a differentiation between ACs and non-ACs. ACs can be further categorised in a more fine-grained manner and they are all connected to the structure, while non-ACs are not connected to the structure. Table 2.1 show how the tools previously mentioned satisfy the features for argumentative structure annotation.

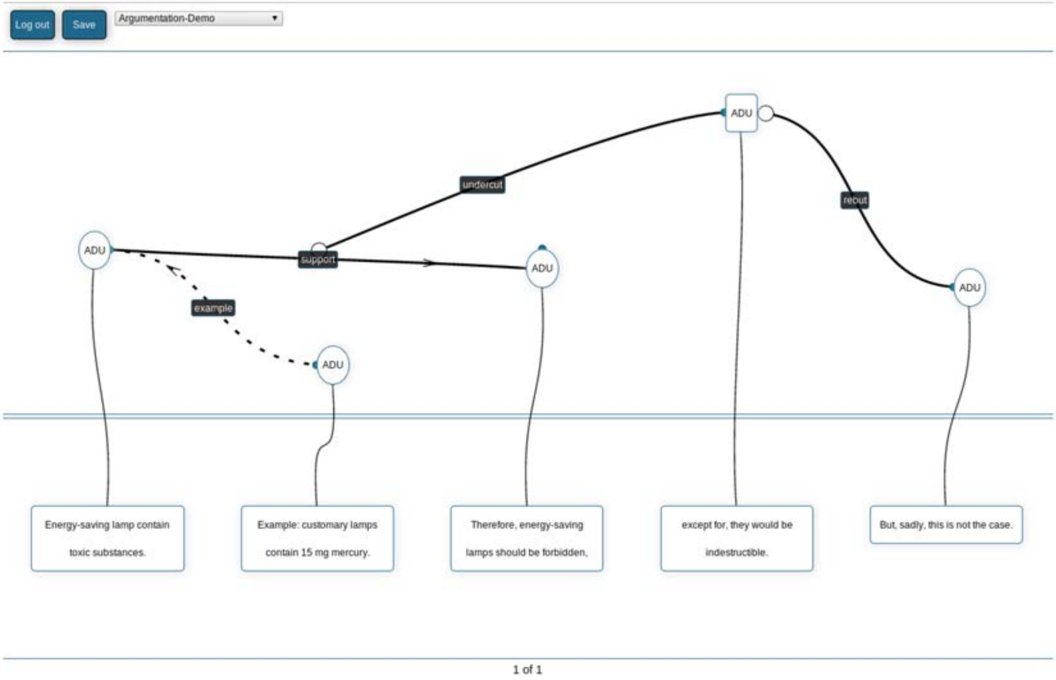
TreeAnno and RSTTool are designed for tree-structured discourse annotation. TreeAnno is easy to use but falls short in the number of features implemented. It visualises the hierarchy of units via node indentation, but it does not show the links between discourse units. RSTTool has a better visualisation, but it only allows RST style annotation, i.e., only two adjacent units can be attached and all units have to

[4] Mechanical aspects are defined as capitalisation, punctuation and spelling.

[5] ICNALE assessors used the scoring rubrics proposed by Jacobs et al. (1981) for English-as-a-second-language composition.

	Feature	TreeAnno	RSTTool	GraPat	DiGAT	OVA
1.	Structure	Tree	Tree	Graph	Graph	Graph
2.	Segmentation		✓			✓
3.	AC and non-AC categorisation			✓	✓	✓
4.	Discourse unit categorisation			✓		
5.	Linking	✓	✓	✓	✓	✓
6.	Link labelling		✓	✓	✓	✓
7.	Structure visualisation	✓	✓	✓	✓	✓
8.	Annotation scheme customisation		✓			✓

TABLE 2.1: Comparison of features in existing discourse annotation tools for argumentative structure annotation



Energy-saving lamp contain toxic substances.

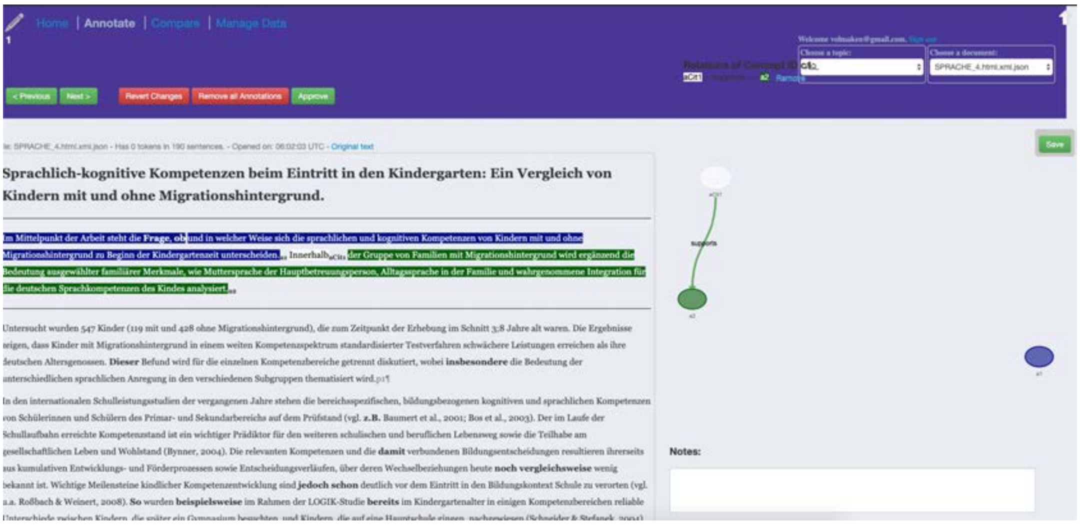
Example: customary lamps contain 15 mg mercury.

Therefore, energy-saving lamps should be forbidden,

except for, they would be indestructible.

But, sadly, this is not the case.

FIGURE 2.1: A screenshot of the GraPAT annotation tool (adapted from Fig. 2 in Sonntag and Stede (2014)).



le: SPRACHE_4.html.xml.json - Has 0 tokens in 190 sentences. - Opened on: 06:02:03 UTC - Original text

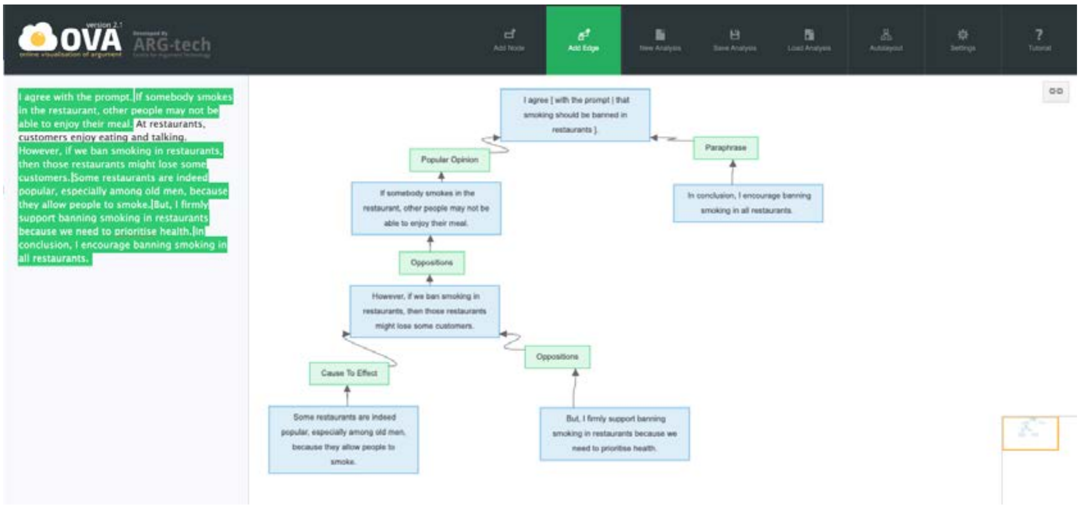
Sprachlich-kognitive Kompetenzen beim Eintritt in den Kindergarten: Ein Vergleich von Kindern mit und ohne Migrationshintergrund.

Im Mittelpunkt der Arbeit steht die **Frage, ob** und in welcher Weise sich die sprachlichen und kognitiven Kompetenzen von Kindern mit und ohne Migrationshintergrund zu Beginn der Kindergartenzeit unterscheiden. Innerhalb der Gruppe von Familien mit Migrationshintergrund wird ergänzend die Bedeutung ausgewählter familiärer Merkmale, wie Muttersprache der Hauptbetreuungsperson, Alltagssprache in der Familie und wahrgenommene Integration für die deutschen Sprachkompetenzen des Kindes analysiert.

Untersucht wurden 547 Kinder (119 mit und 428 ohne Migrationshintergrund), die zum Zeitpunkt der Erhebung im Schnitt 3;8 Jahre alt waren. Die Ergebnisse zeigen, dass Kinder mit Migrationshintergrund in einem weiten Kompetenzspektrum standardisierter Testverfahren schwächere Leistungen erreichen als ihre deutschen Altersgenossen. **Dieser** Befund wird für die einzelnen Kompetenzbereiche getrennt diskutiert, wobei **insbesondere** die Bedeutung der unterschiedlichen sprachlichen Anregung in den verschiedenen Subgruppen thematisiert wird.p1¶

In den internationalen Schulleistungsstudien der vergangenen Jahre stehen die bereichsspezifischen, bildungsbezogenen kognitiven und sprachlichen Kompetenzen von Schülerinnen und Schülern des Primar- und Sekundarbereichs auf dem Prüfstand (vgl. **z.B.** Baumert et al., 2001; Bos et al., 2003). Der im Laufe der Schullaufbahn erreichte Kompetenzstand ist ein wichtiger Prädiktor für den weiteren schulischen und beruflichen Lebensweg sowie die Teilhabe am gesellschaftlichen Leben und Wohlstand (Bynner, 2004). Die relevanten Kompetenzen und die **damit** verbundenen Bildungsentscheidungen resultieren ihrerseits aus kumulativen Entwicklungs- und Förderprozessen sowie Entscheidungsverläufen, über deren Wechselbeziehungen heute **noch vergleichsweise** wenig bekannt ist. Wichtige Meilensteine kindlicher Kompetenzentwicklung sind **jedoch schon** deutlich vor dem Eintritt in den Bildungskontext Schule zu verorten (vgl. u.a. Roßbach & Weinert, 2008). **So** wurden **beispielsweise** im Rahmen der LOGIK-Studie **bereits** im Kindergartenalter in einigen Kompetenzbereichen reliable Unterschiede zwischen Kindern, die später ein Gymnasium besuchten, und Kindern, die auf eine Hauptschule gingen, nachgewiesen (Schneider & Stefanek, 2004)

FIGURE 2.2: A screenshot of the DiGAT annotation tool.



I agree with the prompt. If somebody smokes in the restaurant, other people may not be able to enjoy their meal. At restaurants, customers enjoy eating and talking. However, if we ban smoking in restaurants, then those restaurants might lose some customers. Some restaurants are indeed popular, especially among old men, because they allow people to smoke. But, I firmly support banning smoking in restaurants because we need to prioritise health. In conclusion, I encourage banning smoking in all restaurants.

I agree [with the prompt | that smoking should be banned in restaurants].

If somebody smokes in the restaurant, other people may not be able to enjoy their meal.

In conclusion, I encourage banning smoking in all restaurants.

However, if we ban smoking in restaurants, then those restaurants might lose some customers.

Some restaurants are indeed popular, especially among old men, because they allow people to smoke.

But, I firmly support banning smoking in restaurants because we need to prioritise health.

FIGURE 2.3: A screenshot of the OVA annotation tool.

be connected to the structure. In contrast, argumentative structure annotation may require establishing relations between arbitrary ACs.

GraPat, DiGAT and OVA offer relatively many features that support argumentative structure annotation tasks, and they assume a graph structure of texts. However, GraPat and DiGat require a considerable effort to customise their annotation schemes. While any tree structure is by definition also a graph, these tools cannot ensure structural compliance, should an annotation project assumes a tree structure. GraPat is the only tool among the surveyed tools that support discourse unit categorisation, into *proponent* and *opponent*; the distinction is represented by different shapes of nodes (circular for proponent and rectangular for opponent in Figure. 2.1).

GraPat draws the relations between units on top of the texts. However, Kirschner et al. (2015) argued that the visuals in GraPat might be confusing for texts with multiple long sentences. Their solution to the problem, DiGAT, splits the display into a text and its structural view (Figure. 2.2); a design which is also implemented in OVA (Figure. 2.3). In DiGAT's structural view, the text corresponding to a node is not shown. Texts and nodes are associated by IDs instead. I believe it is essential to see both text and structure at the same time as OVA does, because it is cognitively expensive to synthesise two views in one's mind by switching between the left and right sides of the screen as in DiGAT.

Despite supporting almost all features for argumentative structure annotation, OVA does not fulfil my desiderata because it does not support sentence reordering annotation. I develop my own annotation tool instead considering the time and effort to modify an existing tool. A task-specific tool is also often better for the annotation process and can lead to a better inter-annotator agreement (Sonntag and Stede, 2014). Section 3.1 outlines my annotation needs, and Section 3.2 describes how these requirements translate to design and features implemented in my annotation tool TIARA.

2.3 Argumentative Structure Parsing

Past studies in AM have proposed various models to parse the argumentative structure in texts. Peldszus and Stede (2015) proposed a joint minimum-spanning-tree model for parsing the argumentative structure in the MTC. Stab and Gurevych (2017) proposed separate feature-based models for argumentative component segmentation, component labelling and argumentative structure prediction tasks in PEC. The individual models were then combined using an integer linear programming formulation. Song et al. (2020) proposed to use inter-sentence attentions to capture sentence interactions for component categorisation in the PEC. They also experimented with sentence positional encoding, which improved the model performance. Nguyen and Litman (2016) performed a relation classification between pre-linked source and target ACs in PEC based on contextual features obtained from other surrounding ACs.

Potash et al. (2017) formulated argumentative structure parsing as a sequence prediction task. They jointly performed AC classification and AC linking using a Pointer Network (Vinyals et al., 2015), assuming that the segmentation and AC versus non-AC categorisation tasks have been pre-completed. Working in the same setting, Kuribayashi et al. (2019) proposed to treat discourse markers and argumentative segments separately. They utilised LSTM-minus (Wang and Chang, 2016) to represent textual spans. Both studies experimented on MTC and PEC.

Eger et al. (2017) formulated AM in three ways: as relation extraction, as sequence tagging and as dependency parsing tasks. They performed end-to-end AM at token-level, executing all sub-tasks in AM at once. Eger et al. defined a BIO tagging scheme that contained the distance attribute between the current token and another token it relates to. This BIO scheme also contained the argumentative component and relation attributes. Eger et al. achieved the highest performance in their experiments with the relation extraction model LSTM-ER (Miwa and Bansal, 2016). Ye and Teufel (2021) also performed end-to-end AM at the token-level. They proposed a novel representation for the dependency structure of arguments, which was more efficient than Eger et al.'s. They achieved the state-of-the-art (SotA) performance of .729 and .459 in F1-score for component and relation identifications, respectively, on the PEC dataset by using a biaffine attention model (Dozat and Manning, 2017).

The biaffine attention model was originally designed to parse token-to-token dependency, but Morio et al. (2020) extended it to parse proposition (span) level dependency. Their model also can deal with arguments that form a graph structure. They evaluated the model on the Cornell eRulemaking corpus (avg. 6.7 sentences) (Park and Cardie, 2018), and achieved the performance of .795 and .338 in F1 score for component and relation predictions, respectively.

The argumentative structure parsing task, particularly the identification of links between ADUs, has been identified as a complex and difficult task in AM (Cabrio and Villata, 2018; Lippi and Torroni, 2016). There are many possible combinations of links between textual units, and a parsing model has to find the most proper structure out of many possibilities; this involves understanding the flow of reasoning in the text. The common approach to solve this challenge is the use of multi-task learning (MTL), executing several or all AM sub-tasks at once to provide a rich supervision signal, as carried out in several of the aforementioned studies. Another MTL direction is to train an AM model with other related CL tasks. For example, Lauscher et al. (2018) jointly performed argumentative component identification and rhetorical classification tasks in scientific publications.

In general, AM studies also suffer from the size of annotated corpora (Schulz et al., 2018). Corpus construction is a complex and time-consuming task; it often requires a team of expert annotators. Existing corpora in AM are relatively "small" compared with more established fields, such as machine translation or syntactic analysis. This hinders training AM models when using a supervised machine learning framework.

Several approaches have been applied to alleviate the data sparsity problem in AM. Al-Khatib et al. (2016b) used a distant supervision technique to acquire a huge amount of data without explicit annotation. Accuosto and Saggion (2019) pretrained a discourse parsing model and then fine-tuned it on AM tasks. Schulz et al. (2018) performed a cross-genre argumentative component identification. They employed a sequence tagger model with a shared representation but different prediction layers for each genre. Cabrio and Villata (2012) used a textual entailment model to detect the relations between pairs of arguments. Toledo-Ronen et al. (2020) adopted a multilingual language model to perform AM in low-resourced languages.

Data augmentation can also be employed to mitigate the data sparsity problem. This aims to increase the amount of training data without directly collecting more data (Feng et al., 2021; Liu et al., 2020). The distribution of augmented data should neither be too similar nor too different from the original to avoid both underfitting and overfitting (Feng et al., 2021). Common strategies either add slightly modified copies of existing data or create synthetic data, for instance, the *backtranslation* method in the machine translation field (Sennrich et al., 2016). A relatively straightforward strategy is to use multiple corpora when training models. For example, Chu et al. (2017) proposed a *mixed fine-tuning* approach, training a machine translation model on an out-genre corpus, and then fine-tuning it on a dataset that is a mix of the target-genre and out-genre corpora. However, the use of multiple corpora of different genres is challenging in AM because argumentation is often modelled differently across genres (Lippi and Torroni, 2016). Daxenberger et al. (2017) found that training a claim identification model with mixed-genre corpora only performs as well as training on each specific corpus. The use of data augmentation may cause the *distributional shift* problem as well, where augmented data alter the target distribution that should have been learned by the model (Feng et al., 2021; Gontijo-Lopes et al., 2021),.

In this thesis, I adapt Eger et al.'s sequence tagging formulation, using vanilla Bidirectional Long-short-term memory (BiLSTM) network (Hochreiter and Schmidhuber, 1997; Huang et al., 2015), as this architecture can be straightforwardly applied to my task. I also approach argumentative structure parsing as a dependency parsing task, as both tasks model tree structures. To this end, I adapt the biaffine attention model. However, different to Morio et al. (2020), I operate at the sentence level instead of span level and model argumentative structure as a tree instead of a graph. I also propose an MTL scheme to provide a richer supervision signal by using structural-modelling-related auxiliary tasks instead of the more commonly used rhetorical or discourse-related auxiliary tasks. My auxiliary tasks require no additional annotation. I also explore the possibility of multi-corpora training in AM in depth. First I investigate whether it is possible to use corpora of different text quality but having the same genre to train an AM model, for instance, one corpus concerns intermediate level and another one concerns more proficient writers. Second, I also explore whether it is possible to use multiple corpora of the same genre but are annotated using different schemes.

2.4 Sentence Ordering

My sentence *re*-ordering task aims to find a better sequence for sub-optimally ordered input sentences, otherwise, retain the original input order if it is already well-structured. On the other hand, the existing sentence ordering task aims to find a coherent order for given a set of unordered sentences. Both tasks differ in terms of the presence of prior ordering in the input, but they have a similar goal to generate a well-ordered text. Therefore, strategies for the sentence ordering can be useful for reordering.

The sentence ordering task is useful in many NLG applications, such as in multi-document news summarisation. One way to generate a well-ordered news summary is sorting sentences according to their time stamp, that is, chronological ordering (Barzilay et al., 2002; Okazaki et al., 2004). On top of this, Bollegala et al. (2006) considered three additional criteria: topical closeness, contextual precedence and contextual succession.[6] Christensen et al. (2013) introduced a joint model for sentence selection and ordering. A summary is then generated by searching a sequence of

[6] Given two sentences a and b (both should be included in the summary) from two different documents D_a and D_b, the precedence criterion measures how similar a is to sentences that appear before b in D_b. If the similarity is high, then a should appear before b in the summary. The succession criterion is the opposite of precedence.

sentences that balances ordering, summary salience and redundancy scores. However, strategies in the aforementioned studies are often tailored to the news genre and tightly integrated with the multi-document summarisation task.

Other studies also proposed more generic or open-domain methods, approaching the sentence ordering task as a standalone problem. For example, Lapata (2003) proposed a probabilistic ordering approach. Their idea is to generate a sequence of sentences that maximises the local transition probability between two adjacent sentences $P(S_{i+1} \mid S_i)$, that is, maximising local coherence. Since there are $N!$ possible number of arrangements for a set of N sentences, they used a greedy beam-search procedure to find the best sequence. This idea is analogous to bigram language modelling, and it has been adopted in other NLG studies (e.g., El Baff et al., 2019). The simplest method adopts lexical cohesion as a proxy for local coherence, that is, overlapping words or nouns between sentences (Barzilay et al., 2002; Barzilay and Lapata, 2008). More complex methods consider syntactical property (Louis and Nenkova, 2012), topic-comment structure (Ermakova et al., 2017), semantic similarity (Putra and Tokunaga, 2017) and rhetorical-category transition (cf. Section 2.1.1). Some also proposed a neural method to compute the bigram probability (Chen et al., 2016).

Li and Jurafsky (2017)[7] presented an approach from a more global perspective. They proposed the *paragraph reconstruction task*: reconstructing a paragraph from a bag of constituent sentences. More formally, given a set of permuted (scrambled) sentences, the goal is to return the original order of sentences (presumably the best order). They trained a generative encoder-decoder model to solve the task. Other studies followed suit and proposed various sequence-to-sequence architectures, for example, Gong et al. (2016), Logeswaran et al. (2018), Cui et al. (2018) and Yin et al. (2019). It is assumed that the end-to-end models could capture some form of implicit discourse structure when performing the paragraph reconstruction task. Training such kind of end-to-end model is expensive. Ideally, the model has to be provided with $N! - 1$ number of permutations of input for a given paragraph of N sentences. However, only a small number of subset from these permutations were used in reality, for example, 20 random permutations per paragraph (Logeswaran et al., 2018).

Prabhumoye et al. (2020) formulated the paragraph reconstruction task as a constraint learning problem. They trained a model to learn the relative ordering between all pairs of sentences in text. The sequence of coherent text was then generated by using a topological sorting technique (Tarjan, 1976). Although computationally simpler than the end-to-end model, this approach attained the SotA performance for the paragraph reconstruction task on paper abstracts (approximately 5 to 6 sentences in length): .81, .66 and .83 in Kendall's Tau (Kendall, 1938) on NIPS, NSF and ANN abstracts, respectively. It also attained the SotA performance for SIND captions (.60 in Kendall's Tau).[8]

In this thesis, my approach for rearranging sentences in EFL essays consists of two modules: (1) argumentative structure parsing and (2) sentence reordering. Compared with existing studies, I explicitly analyse the argumentative structure and then utilise it as an input for the subsequent reordering module. Model explainability is very important in the educational domain. To this end, the argumentative structure, as well as acting as an intermediate output in the reordering context, is

[7] The first draft of Li and Jurafsky's paper appeared on ArXiv on 5 June 2016, earlier than Gong et al.'s paper which appeared on 15 November 2016.

[8] NIPS = Neural Information Processing Systems, NSF = National Science Foundation, AAN = ACL Anthology Network, SIND = Sequential Image Narrative Dataset.

an output in itself and useful to provide some form of explanation for students (cf. Section 2.1.1 and 2.1.2).

Chapter 3

Corpus Construction and Annotation Study

I present a novel language resource of EFL essays annotated with argumentative structure and sentence reordering to facilitate implementing argumentative structure parsing and sentence reordering models in this thesis. This chapter describes my annotation scheme, followed by a new annotation tool to support my task. It then discusses the appropriate metrics to evaluate such structural annotation. Finally, this chapter describes a corpus of annotated 434 essays that results from the annotation effort.

3.1 Annotation Scheme

3.1.1 Target Domain

My target texts are sourced from the 640-essay-subset of the ICNALE corpus which has been corrected in terms of grammar and mechanics (cf. Section 2.2.2). From this 640 subset, I exclude low-quality essays, those with extremely poor structure or so little content that they are hard to interpret. I manually investigated the quality of randomly sampled essays to check the total score at which the quality drops to a point where it is hard to understand what the students want to convey. I identified that point as a score of 40 points, affecting 4.1% of the essays. Essays scoring below this point would require a major rewriting before they could be analysed.

At the other end of the spectrum, I also exclude essays that are of very high quality. The annotation of such already well-written essays would be of limited use towards my long-term goal of improving the writing of EFL students who have not yet reached this level. I found that essays scoring 80 points or more (15.2% of the total) are already well-written and coherent. Of course, it might be possible to improve their quality and persuasiveness even further, but they are comparable with essays written by advanced or proficient writers. The remaining 517 essays scoring between 40 and 80 points (80.8% of the total) should therefore be what can consider intermediate-quality essays. I had to manually discard a further 63 essays for the reason that they contained a personal episode related to the prompt instead of a generalised argument or they lacked a clear argumentative backbone for some other reason. While the 454 remaining texts are sometimes still far from perfect, they are quite clear in almost all cases in terms of what the author wanted to say. These essays also contain a plan for an argument that is at least roughly acceptable, as well as the right material for the plan.

The average length of the remaining essays is 13.9 sentences. I used 20 essays for a pilot study not reported here,[1] leaving us with 434 essays;[2] these constitute the pool of essays I use in this thesis (hereafter referred to as "ICNALE essays" or "ICNALE corpus").

3.1.2 Annotation of Argumentative Structure

Following the common practice in AM (cf. Section 2.1), my annotation scheme consists of two steps. The first is *argumentative component identification*, where annotators identify sentences as ACs and non-ACs. The second step is *argumentative structure prediction*, where annotators identify relations between ACs. These relations then form a hierarchical structure. For other genres, such as scientific papers (Kirschner et al., 2015) and user comments (Park and Cardie, 2018), annotation schemes are sometimes based on graphs rather than trees. For argumentative essays, however, I observed that a simple tree structure suffices in the overwhelming number of cases and that it most naturally expresses the predominant relation where a single higher-level statement is recursively attacked or supported by one or more lower-level statements (Carlile et al., 2018; Stab and Gurevych, 2017).

In a departure from existing work, where the textual units of analysis are represented at the clause level, the units (ACs and non-ACs) in my scheme are always full sentences. Textual units smaller than sentences but bigger than words, such as clauses, are hard to define in a logical and linguistically clear manner suitable for annotation. There is still no easily applicable annotation instruction for capturing meaningful argumentation units at the sub-sentential level despite several attempts in the literature (e.g., Fries, 1994; Huddleston and Pullum, 2002; Leffa and Cunha, 1998). In practice, annotation studies often use an idiosyncratic definition of which textual units constitute an argumentative component (Daxenberger et al., 2017; Lippi and Torroni, 2016), resulting in a lack of interoperability between annotation schemes. While I acknowledge that the use of sentence in this thesis is a theoretical simplification, it is well-motivated from the computational perspective. In fact, existing studies in AM also operate at the sentence level, for example, Carstens and Toni (2015), Kirschner et al. (2015) and Wachsmuth et al. (2016). When defining units, we certainly cannot go beyond the sentence level toward larger units. Students may have added paragraph breaks, but these are not recorded in the ICNALE corpus. In any case, paragraphs would certainly be too large as atomic units given that the ICNALE essays only have an average length of 13.9 sentences.

In my scheme, as in that by Stab and Gurevych (2017), the *major claim* is topologically distinguished as the root of the tree structure, which is recognisable as the only node with incoming but without outgoing links. In contrast to their scheme, however, I do not additionally label ACs as *major claim*, *claim* and *premise*. The decision not to do so is to avoid conflicts that might arise in long argumentation chains, particularly between claims and premises. A premise at level X can easily itself become the claim for a lower-level premise at level $X + 1$, making the AC act as both claim and premise at the same time. This means that none of the fixed labels is applicable with a finite number of labels. I note that such ambiguous cases do happen in Stab and Gurevych's PEC; these cases were resolved according to topology, a treatment that is consistent with my decision not to label ACs in the first place. I argue that omitting AC labels makes my annotation scheme not only more economical but also

[1] Putra et al. (2019) shows a partial description of my pilot study.

[2] Approximated CEFR level of A2 (94 essays), B1 (253) and B2 (87).

intrinsically consistent. I use the term *major claim* to refer to a concept, and not as an explicitly annotated category in the rest of this thesis.

Non-argumentative Material

My scheme marks discourse units as ACs and non-ACs. Traditionally, non-ACs refer to units that do not function argumentatively. In another departure from existing work, I use a more fine-grained model of non-ACs as follows.

(a) Disconnected sentences.
My scheme excludes isolated sentences, that is, those that do not function argumentatively and thus are not connected to the logical argument. Such sentences might convey an opinion about the prompt statement, for example, "*this is a good question to discuss*" or a personal episode regarding the prompt.

(b) Meta-information.
My scheme excludes sentences that make statements about other sentences without any new semantic content because such sentences contribute nothing substantial toward the argument. An example is "*I will explain my reasons.*"

(c) Redundant material.
My scheme also excludes repetitions of low-level argumentative material, such as facts. For instance, "*a barista has to interact with lots of people.*" might be repeated as "*baristas have much contact with customers.*" In my scheme, one of these sentences (most often the second one) would be marked as non-AC.

Directed Relation Labels

My scheme employs three directed relation labels: SUPPORT (*sup*), DETAIL (*det*) and ATTACK (*att*). In my scheme, these relations are defined as going from a child node (here also called *source* sentence) to a parent node (*target* sentence).

SUPPORT is a commonly used relation label in AM. Here, the source sentence asserts the reasons why readers of an essay should believe the content of the target sentence. This is done by providing argumentative material in support of the target, such as supporting evidence, and this material should be new to the argument. ATTACK is another commonly used relation label, denoting a source sentence that argues for the opposite opinion of the target sentence.

The DETAIL label is less common, but there is precedent for it in the work of Kirschner et al. (2015). It is applied if the source sentence does not provide any new argumentative material in my scheme. This typically happens in two cases: (1) when the source sentence presents additional detail, that is, further explanation, example, description or elaboration of the target sentence or (2) when the source sentence introduces the topic of the discussion in a neutral way by providing general background. Thus, it is the presence or absence of new argumentative material that differentiates the labels DETAIL and SUPPORT. There is an interesting distinction between DETAIL and SUPPORT when it comes to the ordering of sentences. The canonical ordering in a SUPPORT relation places the target sentence before the source sentence (Bacha, 2010; Kaplan, 1966; Silva, 1993; Yanase et al., 2015). Things are a little more nuanced with detail. We tend to regard it as background information when a source sentence in the DETAIL relation appears before its target, whereas we tend to regard it as a further elaboration if it appears after the target sentence.

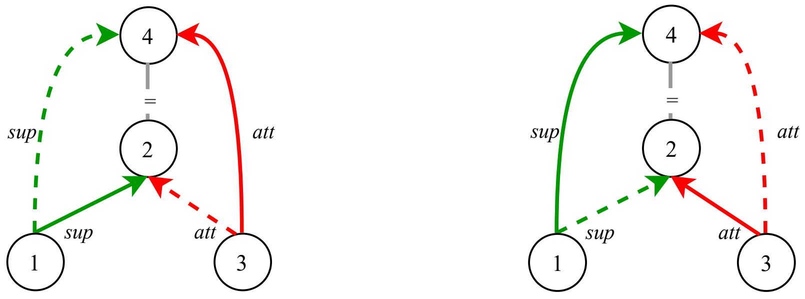
Restatement

I noticed that in many cases, the major claim is restated in the conclusion section of an essay, summing up the entire argument. Skeppstedt et al. (2018) also noticed this and coined the label RESTATEMENT to model this phenomenon. In my scheme, the RESTATEMENT relation holds between two sentences if the second one repeats high-level argument material that has been previously described by the first, without adding a new idea into the discourse. Restatements repeat key argumentative material at a high level in the argument (claims or main claims, not premises or mere facts), and they do so at strategic points in the linear text. This can reinforce the persuasiveness of the overall argument.

I distinguish redundant material from restatements, which are considered ACs although they do contain repeated information—the difference is that in the case of a restatement, we can assume the repetition is intentional and aimed at affecting the flow of argumentation.

Unlike SUPPORT, ATTACK and DETAIL, the RESTATEMENT relation (which is expressed by the symbol "=") is an equivalence relation and therefore semantically non-directional. Source and target sentences convey the same meaning; they are not in a hierarchical relationship. As a result, I treat the two sentences as an equivalence class with respect to all outgoing and incoming relations they participate in.

In argumentative structure annotation, implicit relations can arise which follow logically or semantically from other annotations even though those relations are not explicitly stated. Restatements introduce one particular kind of such implicit relations. Therefore, it can be necessary to also consider the implicit relations to correctly interpret the argument.



a: Annotation *A*. b: Annotation *B*.

FIGURE 3.1: Closure over RESTATEMENT relation. Solid links are explicit, dashed lines implicit.

Figure 3.1 shows such a situation involving implicit links, where different annotations are compared under restatement closure. Annotation *A* recognises a SUPPORT link between nodes 1 and 2 and an ATTACK link between nodes 3 and 4, whereas annotation *B* recognises the SUPPORT link between nodes 1 and 4 and the ATTACK link between nodes 3 and 2. Annotations *A* and *B* do not share a single one of these explicit links, yet they are identical if we consider implicit restatement-based links. If nodes $\{2, 4\}$ are considered a restatement cluster, then both annotations agree that an ATTACK link connects node 3 to restatement cluster $\{2, 4\}$ and a SUPPORT link connects node 1 to the restatement-cluster $\{2, 4\}$, even though they mark this differently.

This new interpretation of the semantics of RESTATEMENT as an equivalence class is a conscious decision on my part, which necessitates the computation of implicit links by some additional machinery. Other implicit links are also theoretically possible in argumentation,[3] but I do not consider them here.

3.1.3 Annotation of Sentence Reordering and Text Repair

After annotating the argumentative structure, the next task in my annotation scheme is to reorder sentences to produce a better structured texts. There is no particular ordering strategy employed, for instance, "all essays should follow the 'claim-support' structure." Instead, annotators are asked to rearrange sentences such that the most logically well-structured text (that they can think of) results.

There are two reasons for this open-ended instruction. First, it allows us to investigate in which circumstances reordering may happen naturally. Second, existing studies have argued that it is not possible to identify a single ordering configuration as the correct one (Barzilay et al., 2002; Todd et al., 2004). In the production annotation, I employ an expert annotator and assume that they can identify one of the possible better orderings. I then evaluate whether the expert's reordering indeed improves the quality of the essay, while conceding that quality improvement might also come from other possible configurations.

Reordering, however, may cause irrelevant or incorrect referring and connective expressions (Iida and Tokunaga, 2014). To correct these expressions, I allow superficial repair of the text where this is necessary to retain the original semantics of the sentence. An example of this is to replace a pronoun with its referent noun phrase or to make an implicit connective explicit by the use of conjunctions, for example, "because".

Sometimes, EFL students make the error of assuming that the prompt is read alongside the text, although an argumentative essay should be understandable without readers knowing the prompt. It is necessary to rewrite the major claim by including some information from the prompt if we aim to make the repaired essay stand-alone. For example, "*I think* ***so***" with *so* referring to the prompt (underlined as follows) needs to be rephrased as "*I think smoking should be banned at all restaurants*."

Sentence reordering annotation makes my corpus unique compared with existing argument-annotated corpora (cf. Section 2.2.1) because it contains aligned student texts and the parallel improved texts. This also makes my corpus useful for conducting empirical analyses of argumentative structure and sentence order.

3.1.4 Annotation Procedure and Example

Annotators start by dividing the text into its introduction, body and conclusion sections in their minds,[4] and then dividing the body section recursively into sub-arguments. They also have to identify the major claim during this process.

The idea of sub-arguments is based on the observation that it is common for groups of sentences about the same sub-topic to operate as a unit in argumentation, forming a recursive structure. I instruct annotators to start the annotation process by marking relations within a sub-argument; later, they analyse how the sub-argument as a whole interacts with the rest of the text. The connection between the sub-argument and the rest of the argument is annotated by choosing a representative sentence standing in for the group.

[3] For instance, the "double-attack" construction, where there is an attack on an attacking claim, can in some cases be interpreted as involving an implicit support link.

[4] Note that this structure is a very common development plan of argumentative essays (Bacha, 2010; Silva, 1993).

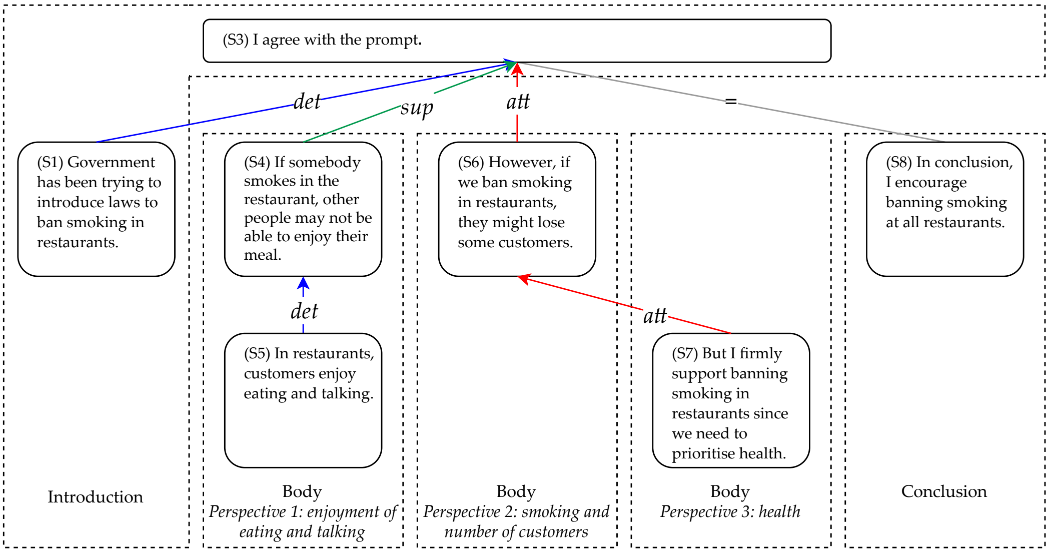

(S3) I agree with the prompt.
det
sup
att
=
(S1) Government has been trying to introduce laws to ban smoking in restaurants.
(S4) If somebody smokes in the restaurant, other people may not be able to enjoy their meal.
det
(S5) In restaurants, customers enjoy eating and talking.
(S6) However, if we ban smoking in restaurants, they might lose some customers.
att
(S7) But I firmly support banning smoking in restaurants since we need to prioritise health.
(S8) In conclusion, I encourage banning smoking at all restaurants.
Introduction
Body
Perspective 1: enjoyment of eating and talking
Body
Perspective 2: smoking and number of customers
Body
Perspective 3: health
Conclusion

FIGURE 3.2: Argumentative discourse structure annotation of example text from page 26.

I now illustrate how my annotation scheme works using a fictional argumentative essay with the prompt "*Smoking should be completely banned at all the restaurants in the country.*"

> (S1)*Government has been trying to introduce laws to ban smoking in restaurants.* (S2)*I have watched the news.* (S3)*I agree with the prompt.* (S4)*If somebody smokes in the restaurant, other people may not be able to enjoy their meal.* (S5)*In restaurants, customers enjoy eating and talking.* (S6)*However, if we ban smoking in restaurants, they might lose some customers.* (S7)*But I firmly support banning smoking in restaurants since we need to prioritise health.* (S8)*In conclusion, I encourage banning smoking at all restaurants.*

This essay can be divided into several parts. S1–S3 together form the introduction section of the essay. S1 provides a background for the discussion topic, and S3 serves as the major claim of the essay. S2 describes a personal episode that does not have an argumentative function; therefore, is identified as a non-AC and excluded from the argumentative structure.

S4–S5 discuss the topic of enjoyment of eating and talking, with S4 providing the introduction of this idea, and S5 giving an opinion on the topic. Sentence S6 then presents an argument about the number of customers; it supports the opposite opinion of S3. S7 repeats some high-level information that has already been stated before and introduces a new health-related argument. Here, we have to make a choice because we cannot assign two relations for S7 as a source sentence. Our rule is to always give preference to the new argument; here, this is the material about health. Hence, S7 is marked as attacking S6 (and not as restatement). Finally, S8 concludes the whole argument, by restating the major claim, which this time we can mark as a restatement (expressed by "="). Figure 3.2 illustrates the argumentative structure of the essay and shows how it relates to the typical essay development plan.

The second layer of annotation is the sentence reordering followed by text repair. We can swap S4 and S5 to improve the text, presenting background information before an opinion. We can also make the implicit discourse marker "*thus*" at the beginning of S4 explicit. Another step is to repair the prompt-type error of the S3 by changing the phrase "*the prompt*" with some information from the prompt. The final essay is therefore as follows.

> (S1)*Government has been trying to introduce laws to ban smoking in restaurants.* ~~(S2) *I have watched the news.*~~ (S3)*I agree* ~~*with the prompt*~~ ***that smoking should be banned at all restaurants****.* (S5)*In restaurants, customers enjoy eating and talking.* (S4)***Thus****, if somebody smokes in the restaurant, other people may not be able to enjoy their meal.* (S6)*However, if we ban smoking in restaurants, they might lose some customers.* (S7)*But I firmly support banning smoking in restaurants since we need to prioritise health.* (S8) *In conclusion, I encourage banning smoking at all restaurants.*

In this case, I argue that the procedure of argumentative structure analysis followed by reordering and surface text repair has resulted in a better-organised essay. However, there are two caveats. First, I do not claim that this is the only way to improve the example text; in fact, there might be other reordering configurations that work well too. Second, it is still not perfect even though the repaired text has improved in quality. For instance, it has not managed to deal with the redundant mentions of banning smoking in the last two sentences. Nonetheless, I believe that step-by-step improvement is very beneficial in educational settings.

The next section explains how the requirements in my annotation scheme translate into annotation tool functionalities. I also show how the annotation is performed using the newly developed tool.

3.2 TIARA Annotation Tool

This section presents TIARA, a new client-side tool for annotating argumentative structure. Even though the tool was originally developed for my project, it is also designed to be useful for four different levels of annotation as follows.

(a) *Discourse structure* annotation, that identifies how discourse units function in the text and connect amongst each other to form a hierarchical structure.

(b) *Argumentative structure* annotation, as opposed to the generic discourse structure, employs discourse units selectively, that is, the differentiation between ACs and non-ACs. Hence, some units are not connected to the structure.

(c) *Sentence reordering* annotation, that aims to improve text coherence and organisational qualities. Different to the previous two levels of annotation which analyse the texts as they are, sentence reordering modifies the textual surface without modifying the content.

(d) *Content alteration* annotation, that modifies textual content. This feature reflects the consideration of TIARA's potential usage in a real classroom environment. In argumentative writing education, instructors may encourage students to add, delete or modify sentences to enhance persuasiveness or make the argumentation balanced (cf. Section 2.1.2). The tool supports these operations to encourage revisions, and students can accommodate instructors' feedback directly in TIARA.

3.2.1 Design Considerations

There are several considerations that influence TIARA's technical and visual design.

(a) Intuitive interface and visualisation
I believe an annotation tool should provide an intuitive interface and visualisation. In the context of this thesis, it means the annotators must be able to read the sentences in linear order while also viewing the argumentative structure. This is to support both logical-sequencing and structural analysis. The novelty of TIARA lies in this dual-view (text view and tree view), which I believe provides an important aspect of global overview to the annotators, who operate by making local changes.

(b) Annotation scheme compliance and completeness checking
An annotation tool ideally prevents annotation scheme violations, such as illogical annotations, for instance, connecting a sentence to itself. Compliance guarantees offered by annotation tools are attractive; annotators can follow their normal workflow without having to worry about doing something wrong or having to perform separate checks. Project owners also benefit from this as they do not have to ask the annotators for a post-hoc repair of the annotations. TIARA checks in real-time whether the annotation violates any constraints of the annotation scheme and warns the annotator when it does. I implement three constraints in TIARA. First, TIARA does not allow self-loop and circular links. Second, users cannot establish relations from and to non-AC nodes. Third, the annotated structure should form a hierarchical structure. On top of compliance to the scheme, TIARA also checks whether the annotation is *complete* upon saving (incomplete annotation cannot be saved, but this feature can be turned off). Particularly, TIARA checks whether all sentences have been categorised as ACs and non-ACs, and whether all ACs have been connected to the whole structure. This is to ensure that the annotators indeed finish their assignments.

(c) Annotation tracking
Tracking changes and actions performed by the annotators is important because it provides information about annotation behaviour. It is also valuable for troubleshooting annotation schemes because project owners can identify the parts that often cause confusion or require post-hoc repair. For example, we know that labels X and Y are potentially confusing when annotators often change the links labelled with X to Y (and vice versa). TIARA records the annotator actions in each annotation file.

(d) Ease of use, installation and deployment
Ease of use and installation for annotators is often prioritised for annotation design, but I believe that deployment is equally important. Not every project owner is tech-savvy; for them, an annotation tool that is hard to deploy is practically unusable. In contrast, tools that are usable without deployment and may run at the client-side, such as EasyTree (Little and Tratz, 2016), are able to reach and help many potential users, including those who have no knowledge in the inner-work of computer systems; TIARA shares the same principle. Users only need a web browser and the TIARA package. TIARA is designed to be a client-side tool and is written using standard web technologies (JavaScript, HTML, CSS). I use JsPlumb[5] and Treant-js[6] as the visualisation libraries.

[5] https://jsplumbtoolkit.com

[6] https://fperucic.github.io/treant-js/

The deployment necessity (server-side) is often coupled with annotation management features (Yimam et al., 2013), and this is important in a large annotation project (Kaplan et al., 2010). Although the current version of TIARA does not actively support such annotation management yet, I plan to do so in future TIARA versions.

(e) Customisability
An annotation tool must be flexible in order to accommodate a wide variety of annotation tasks (Kaplan et al., 2010). This is important in the early stage of an annotation study when the project goal and annotation scheme might frequently change. I adhere to the principle that users should never have to touch the main code at all; they should be able to customise the annotation tool easily in some other way. Similar to BRAT (Stenetorp et al., 2012), the annotation scheme of TIARA can be changed by editing a configuration file. The project owners should define this configuration script at the start of an annotation project, and keep it unchanged throughout the project. I chose this approach over the alternative, a user interface provided by the tool, e.g., as in RSTTool (O'Donnell, 2000), since JavaScript should not modify local files on-the-fly for security reason.

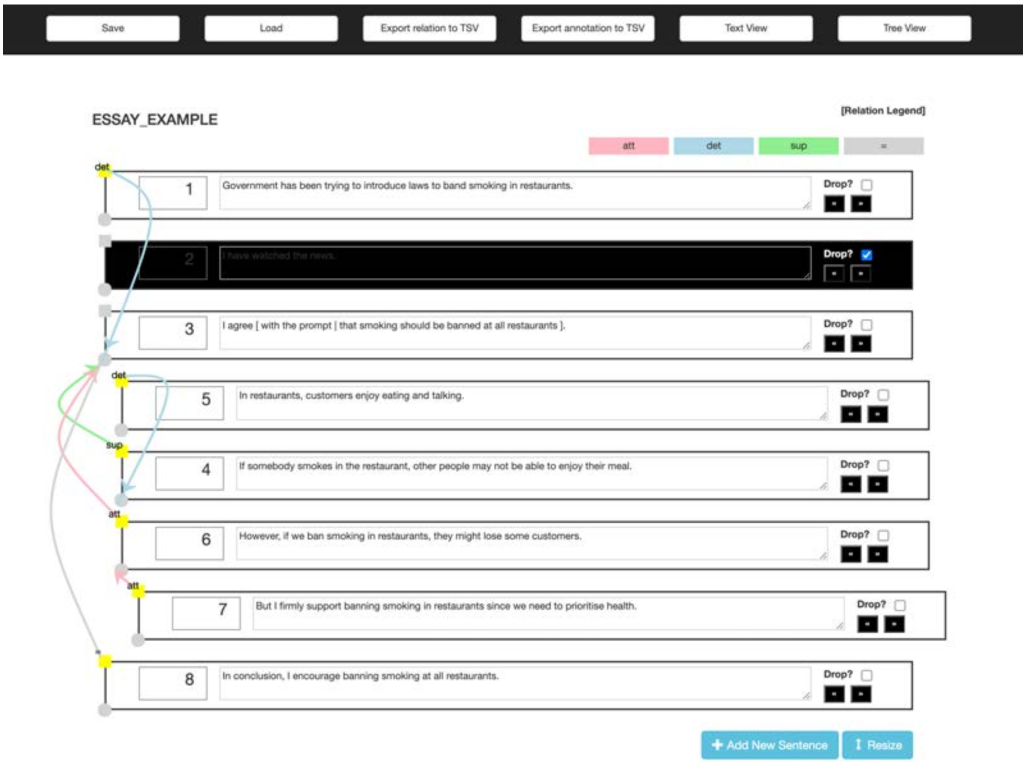
3.2.2 Dual-view and Annotation Example

To illustrate TIARA's dual-view design and how my annotation is operationalised, I show an annotation for the example essay in Section 3.1.4 (full text is presented again here for readability purposes).

> $_{\text{(S1)}}$*Government has been trying to introduce laws to ban smoking in restaurants.* $_{\text{(S2)}}$*I have watched the news.* $_{\text{(S3)}}$*I agree with the prompt.* $_{\text{(S4)}}$*If somebody smokes in the restaurant, other people may not be able to enjoy their meal.* $_{\text{(S5)}}$*In restaurants, customers enjoy eating and talking.* $_{\text{(S6)}}$*However, if we ban smoking in restaurants, they might lose some customers.* $_{\text{(S7)}}$*But I firmly support banning smoking in restaurants since we need to prioritise health.* $_{\text{(S8)}}$*In conclusion, I encourage banning smoking at all restaurants.*

Figure 3.3 illustrates TIARA's **text view** in which the annotation is performed on the example essay. Annotators can read the sentences sequentially while viewing the annotated argumentative structure at the same time. The interface in the text view is split into two parts, the menu navigation part at the top and the work area at the bottom of the interface. After loading a text file, the contents are shown in the work area. Each sentence (i.e., ADU) appears framed in a box (denoting node), numbered ("ID") according to its original order in the input text. Coloured links (defined by the user) depict the annotated relations and their labels. Text repair is present in sentence (3). Note that sentence (2) is *dropped*, that is, deemed non-AC and blacked-out. Users cannot establish a link to or from non-ACs. Sentences (4) and (5) are swapped in position. Sentences (4) to (7) are indented to the right for readability purpose and quickly simulate the hierarchical structure (De Kuthy et al., 2018). Note that indentation does not alter the structural interpretation.

While the text view can be used to illustrate a local hierarchical structure of the argumentation by using indentation, I think that it is not enough for the analysis of the whole argumentative structure. Another view offered by TIARA is the **tree view** that illustrates the shape of the argumentative structure as a whole. Figure 3.4 shows the tree view of the annotation in Figure 3.3. The tree view emphasises the analysis of the overall structure, whereas the text view emphasises the text analysis



ESSAY_EXAMPLE

1 Government has been trying to introduce laws to band smoking in restaurants. Drop? □

2 I have watched the news. Drop? ☑

3 I agree [with the prompt | that smoking should be banned at all restaurants]. Drop? □

5 In restaurants, customers enjoy eating and talking. Drop? □

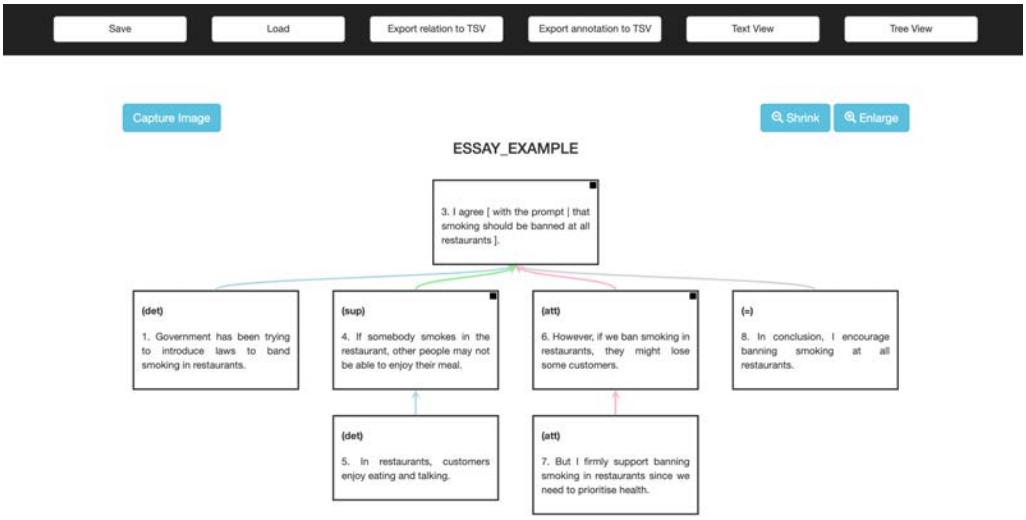
4 If somebody smokes in the restaurant, other people may not be able to enjoy their meal. Drop? □

6 However, if we ban smoking in restaurants, they might lose some customers. Drop? □

7 But I firmly support banning smoking in restaurants since we need to prioritise health. Drop? □

8 In conclusion, I encourage banning smoking at all restaurants. Drop? □

FIGURE 3.3: A screenshot illustrating TIARA's text view.



ESSAY_EXAMPLE

3. I agree [with the prompt | that smoking should be banned at all restaurants].

(det) 1. Government has been trying to introduce laws to band smoking in restaurants.

(sup) 4. If somebody smokes in the restaurant, other people may not be able to enjoy their meal.

(att) 6. However, if we ban smoking in restaurants, they might lose some customers.

(=) 8. In conclusion, I encourage banning smoking at all restaurants.

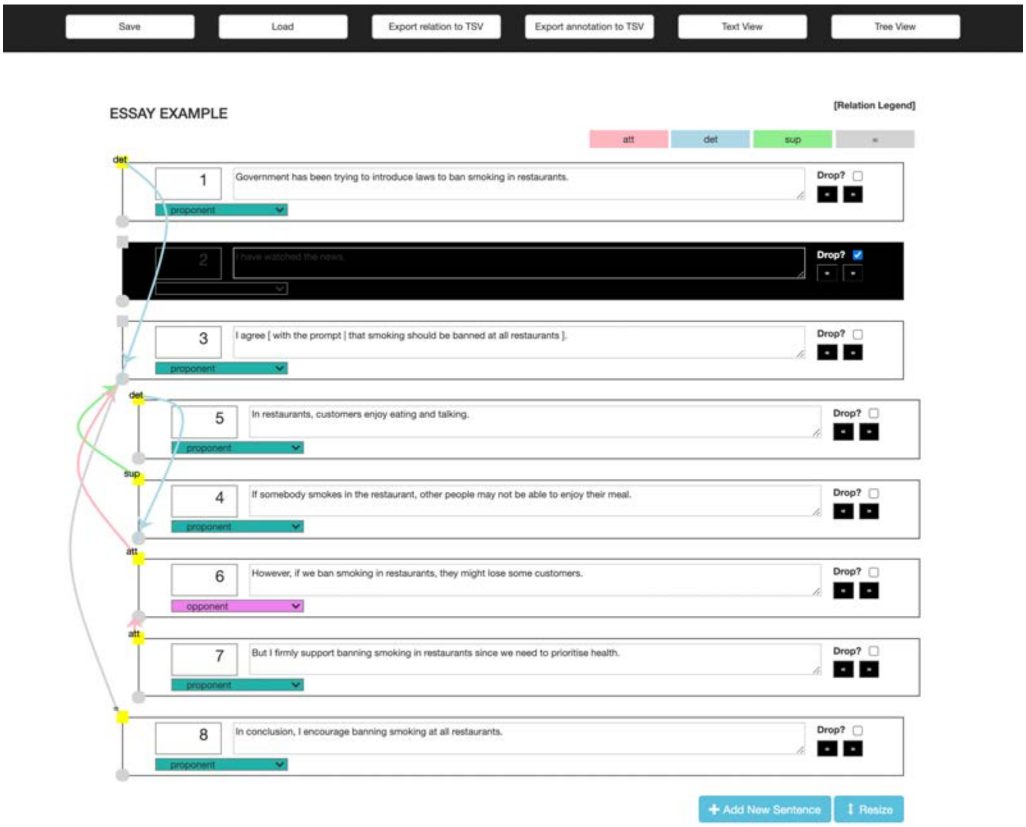
(det) 5. In restaurants, customers enjoy eating and talking.

(att) 7. But I firmly support banning smoking in restaurants since we need to prioritise health.

FIGURE 3.4: A screenshot illustrating TIARA's tree view for the annotation in Figure 3.3. Users may fold and unfold a subtree by clicking the rectangular button on the top-right corner of its root.

on logical sequencing and local connections. Annotators annotate in the text view and then verify their annotation in the tree view; they can freely switch between both views while annotating. I believe that providing the tree view enhances the annotation experience, and therefore, the annotation quality. Annotators may also fold/unfold subtrees in the tree view, which is useful for analysing longer texts as it prevents annotators from being overwhelmed by too much content at once. It is also possible to adjust the text size using the "shrink" and "enlarge" buttons. Users can save the hierarchical visualisation by clicking the "capture image" button.

I have shown an annotated-essay example using my annotation scheme without AC categorisation. However, as has previously mentioned, TIARA also facilitates AC categorisation (discourse unit categorisation in general). This functionality can be turned on and off depending on the project needs (more on this in Section 3.2.3). Figures 3.5 and 3.6 illustrate an annotation with AC categorisation as *proponent* and *opponent*. The difference between Figures 3.4 and 3.6 lies in the additional AC label information inside the boxes and the box colouring.



ESSAY EXAMPLE

1 Government has been trying to introduce laws to ban smoking in restaurants. proponent Drop? □

2 I have watched the news. Drop? ☑

3 I agree [with the prompt | that smoking should be banned at all restaurants]. proponent Drop? □

5 In restaurants, customers enjoy eating and talking. proponent Drop? □

4 If somebody smokes in the restaurant, other people may not be able to enjoy their meal. proponent Drop? □

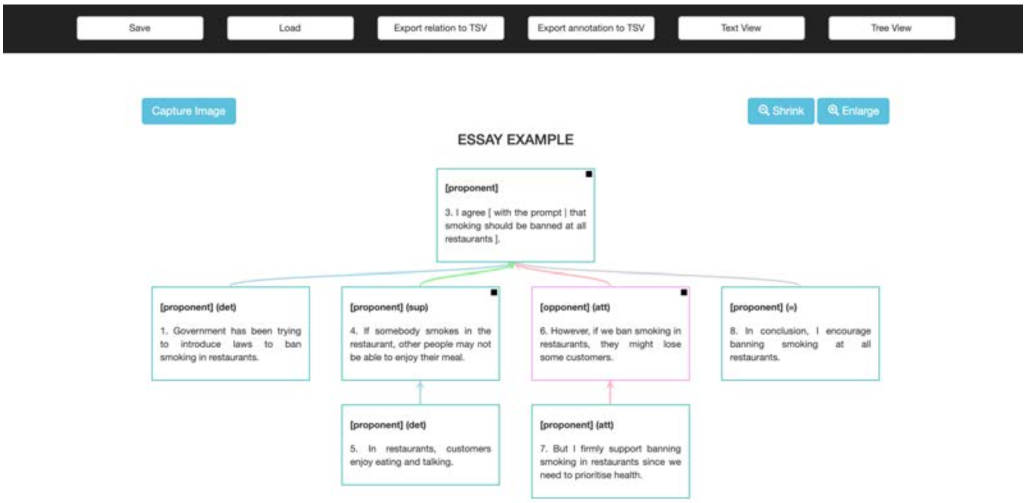
6 However, if we ban smoking in restaurants, they might lose some customers. opponent Drop? □

7 But I firmly support banning smoking in restaurants since we need to prioritise health. proponent Drop? □

8 In conclusion, I encourage banning smoking at all restaurants. proponent Drop? □

FIGURE 3.5: A screenshot illustrating TIARA's text view with discourse unit categorisation functionality.

Note that TIARA is not the first to offer both tree (structural) and text view; DiGAT (Kirschner et al., 2015) also did this (cf. Section 2.2.3). But we can switch between the two views in TIARA instead of looking at both of them simultaneously as in DiGAT. The division between views in TIARA is more advantageous from the cognitive perspective. Human brains are unable to simultaneously process all visual information. Visual attention that is focused on a small area (single task) enables performance benefits, while distributing attention over a large area (multiple tasks in parallel) incurs penalties (Evans et al., 2011; Sun et al., 2015). In my annotation scheme, annotators have to analyse the logical sequencing of sentences and



ESSAY EXAMPLE

[proponent]
3. I agree [with the prompt | that smoking should be banned at all restaurants].

[proponent] (det)
1. Government has been trying to introduce laws to ban smoking in restaurants.

[proponent] (sup)
4. If somebody smokes in the restaurant, other people may not be able to enjoy their meal.

[opponent] (att)
6. However, if we ban smoking in restaurants, they might lose some customers.

[proponent] (=)
8. In conclusion, I encourage banning smoking at all restaurants.

[proponent] (det)
5. In restaurants, customers enjoy eating and talking.

[proponent] (att)
7. But I firmly support banning smoking in restaurants since we need to prioritise health.

FIGURE 3.6: A screenshot illustrating TIARA's tree view for the annotation in Figure 3.5.

the overall discourse structure; both of them are complex and cognitively demanding. Thus, my decision to implement the dual-view allows annotators to focus on one type of analysis at one time. I also take a further step by introducing features that reduce clutter in the display, for instance, the indentation in the text view and the fold/unfold subtrees in the tree view; these functionalities are explained in more detail in the next section.

3.2.3 Functionalities

I split the functionalities provided by TIARA into tree groups: (1) annotation operation, (2) visual operation and (3) miscellaneous.

Annotation operation – TIARA provides the following annotation operation functionalities in its text view. Each function can be enabled or disabled according to a configuration file.

(a) Dropping discourse units
TIARA supports the differentiation between ACs and non-ACs. ACs are connected to form the argumentative structure while non-ACs are not connected to the structure. Users mark non-ACs by checking the "drop" checkbox located at the right-hand side of each sentence box. The box is blacked-out and annotators cannot establish a relation to or from the dropped units when checked. Users may uncheck the checkbox to revert back. This feature can be used to simulate deleting sentences as well in the educational use cases.

(b) Discourse unit categorisation
Users can classify discourse units into their rhetorical categories. This is carried out by selecting the category from the drop-down menu under the sentence in question.

(c) Linking and link labelling
Users link discourse units by dragging an arrow from the rectangular endpoint of the source unit to the circular endpoint of the target unit (left-hand side of

the boxes in the text view) and then TIARA shows a dialogue box for choosing the link label. Annotators may delete or change the link label by clicking the established link in question.

In TIARA, the difference between directed and undirected link is just a matter of visualisation (i.e., the presence of arrow head) and the interpretation of the relation label in question, but not of computation. This strategy is adopted to eliminate circular links which is not allowed in my scheme.

(d) Reordering
Users may move the position of discourse unit boxes by drag and drop operations.

(e) Text revision
TIARA allows users to edit the text inside boxes. Some notation can be employed to track changes. For example, annotators may modify parts of the text if needed in "[*original expression* | *revised expression*]" notation. An illustration is shown in sentence (1) of Figure 3.3. This feature is useful for the text repair operation following reordering annotation (cf. Section 3.1.3. It can also be used to mark grammatical-error correction in the educational use case.

(f) Adding sentences
This feature is specifically designed to support the potential educational use case in learning-to-write. Students do not always write perfect argumentative texts. For example, they may not provide enough reasons to support their claims. In this case, instructors may recommend adding new reasons or elaborating existing content (Cho and MacArthur, 2010; Crossley and McNamara, 2016); the "Add new sentence" button serves this purpose. This feature can also be useful where students are asked to add more counter-arguments to produce a more balanced or comprehensive argument, considering multiple points of view (Hsin and Snow, 2020; Matsumura and Sakamoto, 2021). Instructors may also recommend merging two or more simple sentences into a single sentence. TIARA supports this use case; students may first drop these simple sentences, and then add a new resulting combined content as a new sentence.

Function\Annotation level	Discourse	Argumentative	Reordering	Content
(a) Dropping discourse units		✓		✓
(b) Discourse unit categorisation	✓	✓		
(c) Linking and link labelling	✓	✓		
(d) Reordering			✓	
(e) Text repair			✓	✓
(f) Adding discourse units				✓

TABLE 3.1: The association between annotation functions in TIARA and annotation levels.

Table 3.1 summarises the association between annotation operation functionalities and the varying levels of annotation I have introduced at the beginning of Section 3.2, that is, discourse structure annotation, argumentative structure annotation, sentence reordering followed by text repair, and content alteration. All these functions will be useful for the educational use case as it contains all levels of annotation.

Visual operation – TIARA provides the following visual operation functionalities.

(g) Indentation
TIARA supports discourse units indentation by clicking the indentation buttons at the right-hand side of boxes (under the "drop" checkbox) in the text view. This is useful to quickly visualise the hierarchical structure of the discourse (De Kuthy et al., 2018) and reduce cluttering. However, the indentation does not alter the discourse structure annotation. This feature is only for readability purposes.

(h) Resize
TIARA allows users to adjust the size of sentence boxes by clicking the "resize" button at the bottom of the text view, should a sentence becomes shorter or longer after the editing operation. This feature is only for readability purposes.

(i) Fold and unfold
Annotators may fold and unfold subtrees in the tree view. This is to reduce clutter in the display when annotating long texts.

(j) Shrink and Enlarge
Annotators may adjust the box and font sizes by clicking "shrink" and "enlarge" buttons in the tree view visualisation. This feature is also for the readability purposes.

(k) Capture Image
Annotators may capture and download the tree view visualisation (analogous to screenshot). The captured image can be printed and shared among annotators to facilitate discussion. In the educational use case, instructors may write comments on the printed image to provide feedback to students.

Miscellaneous – TIARA provides other miscellaneous functionalities as follows.

(l) Loading a file
Users can load an un-annotated text file, where the discourse units must have already been separated by a newline. TIARA can also load an annotated file that was saved in its own internal-format.

(m) Saving annotation
The "save" menu can be used by the users to save the annotation in TIARA's internal format. Users can then load and/or modify the annotation. TIARA also offers exporting the annotation into spreadsheet-friendly formats as follows.

- The "export relation to TSV" option extracts relation information of all combinations of discourse units. The output is useful for calculating inter-annotator agreement (Kirschner et al., 2015).
- The "export annotation to TSV" option converts all annotation as "`.tsv`" file where each row contains <essay ID, unit ID, text, corresponding target unit ID, relation label, dropping flag> information. This option converts all annotated information into a table.

(n) Logging
TIARA records the actions (and timestamps) performed by annotators in background. The log information is stored in each saved annotation-file (TIARA's internal format). Therefore, users are aware of each file's history. This feature is also useful for the analysis of annotation behaviour.

(o) Customisation
Users may customise the sentence categories, relation types, relation labels and their colours by modifying an external configuration script. They can also disable or enable certain functions. During a discourse structure annotation project, for example, a project owner may choose to disable the dropping, reordering, text editing and sentence addition functions. However, they should enable the dropping function for an argumentative structure annotation project. Figure 3.7 shows a configuration script example. During the preliminary trial of the tool, I found that users can modify this script as fast as five minutes on their first try.

```
var enableDropping = true; // flag for "dropping discourse units" function
var enableSentenceCategorisation = true; // flag for "discourse unit categorisation"
    function
var enableLinking = true; // flag for "linking" function
var enableReordering = true; // flag for "reordering" function
var enableEditing = true; // flag for "text editing" function
var enableAddNewSentence = true; // flag for "adding discourse units" function
var enableIntermediarySave = false; // flag for "completeness checking" function;
    set true if you allow annotators to save incomplete annotation;
    false if only allowing complete annotation

var sentenceCategories = ['proponent', 'opponent']; // discourse unit categories
var sentenceCatColours = ['lightseagreen', 'violet']; // the visualisation colour for the
    corresponding unit categories

var relLabels = ['att', 'sup', 'det', '=']; // relation labels
var relColours = ['lightpink', 'lightgreen', 'lightblue', 'lightgray'];
    // visualisation colours for the corresponding relation labels
var relDirections = [true, true, true, false]; // relation types, true if directed (arrow head
     presents in the visualisation) and false if undirected (no head)
```

FIGURE 3.7: Example of TIARA's configuration script (written in JavaScript).

Table 3.2 shows in detail how TIARA is situated in terms of its functionalities amongst other surveyed annotation tools (cf. Section 2.2.3), in particular with respect to its support of AM tasks (1–7) and my additional needs (8–10) which of course it is designed to fulfil. Despite supporting a wide range of tasks, the reordering annotation is the problem for other tools. None of the existing tools support such annotation, which is indispensable in my project. I take OVA as TIARA's strongest competitor among the surveyed annotation tools. It offers almost all features needed in my scheme, except for the discourse unit reordering feature. Still, TIARA is more advantageous for annotating longer texts because it offers features to reduce cluttering on the display, for example, the fold/unfold feature in the tree view, while OVA does not provide such a feature.

Overall, there is no one-for-all discourse or argumentative structure annotation tool, but TIARA, with its middle-ground visual solution, is efficient and a strong

Feature	TreeAnno	RSTTool	GraPat	DiGAT	OVA	TIARA
1. Structure	Tree	Tree	Graph	Graph	Graph	Tree
2. Segmentation		✓			✓	
3. AC and non-AC categorisation			✓	✓	✓	✓
4. Discourse unit categorisation			✓			✓
5. Linking	✓	✓	✓	✓	✓	✓
6. Link labelling		✓	✓	✓	✓	✓
7. Structure visualisation	✓	✓	✓	✓	✓	✓
8. Annotation scheme customisation		✓			✓	✓
9. Discourse unit reordering						✓
10. Text editing		✓			✓	✓

TABLE 3.2: Comparison of features in TIARA and other discourse annotation tools in terms of argument mining tasks (1–7) and my additional needs (8–10).

general tool for relation-focused structural annotation. In particular, it provides versatile visualisation for representing structure (the dual-view and clutter-reducing features); annotators can choose the method that works best for them.

Despite its advantages, TIARA does not provide a text segmentation function (i.e., highlighting a continuous sequence of words as a single unit of analysis) because it is not required in my scheme. This might hinder the use of TIARA for segment-level AM. The possible solution is to use other existing text segmentation tools and then load the segmented text files in TIARA for the structural annotation.

An annotation tool can be used in a learning-to-read scenario by supporting discourse or argumentative structure annotations. TIARA offers a competitive edge by also supporting a learning-to-write scenario compared with other tools. In a classroom setting, students could write argumentative essays and simultaneously draw the intended structures on TIARA in parallel, allowing instructors to interactively and quickly point out and address student mistakes that are visible in TIARA's visualisation. Instructors can then suggest improvements in the overall discourse flow (e.g., by reordering sentences), in the textual realisation (e.g., by editing discourse connectives) and argumentation (e.g., by adding more sentences for a stronger or more balanced argument). Therefore, beyond as an annotation tool, TIARA can also be used to enhance the process of student-instructor communication and feedback.[7]

3.3 Inter-annotator Agreement Metrics

In Section 3.4, I perform an agreement study and build a new language resource with the newly defined annotation scheme (cf. Section 3.1). However, we first need to turn our attention to the question of which agreement metrics would be appropriate for structural annotation such as mine. In addition to the conventional metrics

[7] Readers may refer to Strobl et al. (2019) for a more comprehensive explanation on how to integrate digital technologies into language teaching pedagogies. Suleiman (2000) explains the link between practising writing and the mastery of a language.

(Section 3.3.1), I develop new metrics specifically for the study at hand (Section 3.3.2, and later describe the evaluation of these newly-developed metrics (Section 3.3.3).

3.3.1 Conventional Agreement Metrics

If different annotators produce consistently similar results when working independently, then we can infer that they have internalised a similar understanding of the annotation guidelines, and we can expect them to perform consistently in all similar conditions, in particular with new unseen text. Inter-annotator agreement metrics exist for several types of annotation. My task here is a *categorical* classification, where a fixed set of mutually exclusive categories are used and where we assume that the categories are equally distinct from one another (Artstein and Poesio, 2008). The simplest of these is a plain observed agreement ("agreement ratio"). Chance-corrected agreement measures such as Cohen's κ have also been proven to be particularly useful in computational linguistics (Carletta, 1996).

In the context of this thesis, there are three aspects of the structural agreement which can be expressed in terms of categorical classification:

- **Argumentative component identification (ACI).** Each sentence is categorised as either AC or non-AC.

- **Existence of links between sentences** (**sentence linking**). A binary label (linked vs not linked) is assigned to all non-identical sentence pairs in the text (Kirschner et al., 2015).[8]

- **Relation labelling.** For all sentence pairs that have been confirmed as being connected by annotators, I measure whether annotators agree on the relation label that holds between them.

I report the agreement scores of argumentative structure annotation on these three aspects, using agreement ratio and Cohen's κ (Cohen, 1960). I also report the agreement ratio for the entire structure ("entire agreement ratio") to show how errors propagate. The entire agreement ratio measures whether annotators made the same decisions on all aspects of structural annotation for each sentence (as source): the same component category (AC vs non-AC), the same target sentence and the same relation label. It is analogous to multi-label accuracy.

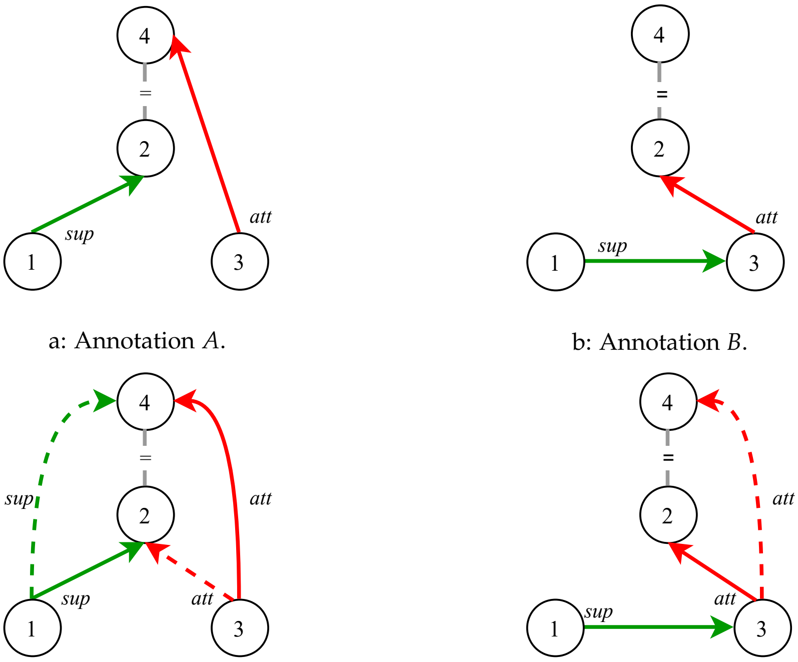
3.3.2 Structure-based Inter-annotator Agreement Metrics

Conventional agreement metrics treat annotated items as independent of each other. However, there are some problems with this assumption for argumentative structure and other types of discourse annotation. Particularly in the sentence linking task, annotation decisions are often structurally dependent on each other; if there is a link from sentence X to sentence Y, other links from sentence X are no longer possible as far as we assume a tree structure. The κ metric does not recognise such dependencies and counts non-linked sentence pairs as correct cases, possibly overestimating the true value.

The second problem concerns implicit links. We have to consider implicit links as the result of the semantics of the RESTATEMENT label as I have argued in Section 3.1.2. Conventional metrics are not suitable for closure structures because they cannot distinguish between explicit and implicit links; instead, they treat implicit links

[8] For a text containing three sentences, there are six possible pairs: $\langle s_1, s_2\rangle$, $\langle s_1, s_3\rangle$, $\langle s_2, s_3\rangle$, $\langle s_2, s_1\rangle$, $\langle s_3, s_1\rangle$ and $\langle s_3, s_2\rangle$; where $\langle s_x, s_y\rangle$ denotes a directed link from s_x to s_y.

as if they are explicit. If implicit links in annotation A do not appear in annotation B, they will be treated as mismatches, and conventional metrics will assign a penalty to the score. Therefore, there might be a large difference in agreement scores between a situation where only explicit links are used and one where both explicit and implicit links are used, which is undesirable. I also think that the fairest treatment of implicit links is to reward in situations where an implicit link is correct, without punishing in situations where the link is incorrect. I will now explain this asymmetry.



a: Annotation A.

b: Annotation B.

c: Closure of A.

d: Closure of B.

FIGURE 3.8: Example of restatement closures. Solid links are explicit and dashed lines are implicit.

Let us consider this point using the two annotations A and B in Figure 3.8. In Figure 3.8a, annotation A marked an explicit link from nodes 3 to 4, which can be expanded by an implicit link from nodes 3 to 2, cf. Figure 3.8c. The fact that the annotators agree that node 3 attacks the restatement cluster $\{2, 4\}$ should be rewarded somehow in my opinion.

Things become more complicated when one annotator links a node into the equivalence cluster when the other annotator links it to a node outside of it. This is illustrated with the links exiting from node 1; A links it to 2 and thus inside the equivalence cluster, whereas B links it to 3 and thus outside the equivalence cluster. It is clear that B should be punished for missing the explicit link 1→2, which is present in annotation A. The question is, should B *additionally* be punished for the lack of the implicit link 1→4, which only arose because node 2 happens to be inside the equivalence cluster? I consider this unfair given that from annotation B's viewpoint, node 1 is not connected to the equivalence cluster. Without a link to the equivalence cluster, B could not possibly have considered the hypothetical implicit link 1→4. Thus, I believe an ideal agreement metric should assign a special treatment to implicit links: (1) to reward implicit links if they match but (2) *not* to punish when implicit links do *not* match.

To allow a more holistic view of structural annotation while alleviating the implicit link problem, I propose a new document-level agreement metric based on the notion of recall, that is, the degree to which each annotation recalls the other annotation in terms of structure. The total number of units for recall calculation normally differs between annotators; this is so because in the earlier AC versus non-AC classification (ACI) step, annotators might have classified different sets of sentences as non-ACs. Consequently, I have to average across the two annotations' recall values and accept that the metric can be defined only for pairs of annotations. My new metric is called "**mean agreement in recall**" (**MAR**). It comes in three variants that differ in how the units are defined: as links (**MAR$^{\mathbf{link}}$**), as paths (**MAR$^{\mathbf{path}}$**) or as descendant sets (**MAR$^{\mathbf{dSet}}$**). The special treatment for implicit links described previously is only applicable to MAR$^{\text{link}}$ and not to the other variants.

When computing structure-based agreement metrics, I need to operationalise undirected links as directed links; if there is a RESTATEMENT link between two nodes A and B, I represent this as $A \rightarrow B$ and $B \rightarrow A$ to account for the equivalence interpretation.[9] I will now describe the metrics in turn.

Link-based MAR

There are two variants of MAR$^{\text{link}}$: (3.1) considering only explicit links and (3.2) also considering implicit links. The implicit version (3.2) rewards implicit links when they appear in another structure but does not punish them when they do not, as described above.

Given two structures A and B with respective sets of explicit links E_A and E_B, MAR$^{\text{link}}$ measures the average recall of links between the two structures as computed in Equation (3.1). Relation labels are disregarded in this metric. For example, MAR$^{\text{link}}$ between annotations A and B in Figure 3.8 is 0.50.[10]

$$\text{MAR}^{\text{link}} = \frac{1}{2}\left(\frac{|E_A \cap E_B|}{|E_A|} + \frac{|E_A \cap E_B|}{|E_B|}\right) \tag{3.1}$$

I modify the formula such that it measures the agreement without giving penalties to implicit links for the closure structures. Given two structures closure(A) and closure(B) with respective sets of link (explicit+implicit) EC_A and EC_B, MAR$^{\text{link}}$ for closure is calculated as in Equation (3.2), as the recall of the closure structure with respect to another explicit structure.

$$\text{MAR}^{\text{link}}(\text{closure}) = \frac{1}{2}\left(\frac{|E_A \cap EC_B|}{|E_A|} + \frac{|EC_A \cap E_B|}{|E_B|}\right) \tag{3.2}$$

For example, MAR$^{\text{link}}$ between closure(A) and closure(B) in Figure 3.8 is 0.75.[11]

Path-based MAR

The second variant is MAR$^{\text{path}}$ that measures the agreement on paths. A path is defined as a sequence of nodes in the argument tree with one or more consecutive edges. For example, the set of path P of annotation A in Figure 3.8 is $\{(4,2,1), (4,2), (2,1), (2,4,3), (2,4), (4,3)\}$. MAR$^{\text{path}}$ between two sets P_A and P_B are calculated as

[9] In contrast, such a link duplication does not happen in the calculation of Cohen's κ, as this metric is not concerned with structure.

[10] $E_A = \{1 \rightarrow 2, 2 \rightarrow 4, 4 \rightarrow 2, 3 \rightarrow 4\}; E_B = \{1 \rightarrow 3, 3 \rightarrow 2, 2 \rightarrow 4, 4 \rightarrow 2\}; E_A \cap E_B = \{2 \rightarrow 4, 4 \rightarrow 2\}$

[11] $EC_A = \{1 \rightarrow 2, 3 \rightarrow 2, 1 \rightarrow 4, 3 \rightarrow 4, 2 \rightarrow 4, 4 \rightarrow 2\}; EC_B = \{1 \rightarrow 3, 3 \rightarrow 2, 3 \rightarrow 4, 2 \rightarrow 4, 4 \rightarrow 2\}$; $E_A \cap EC_B = \{3 \rightarrow 4, 2 \rightarrow 4, 4 \rightarrow 2\}; EC_A \cap E_B = \{3 \rightarrow 2, 2 \rightarrow 4, 4 \rightarrow 2\};$

in Equation (3.3). For example, MAR^{path} between annotation A and B in Figure 3.8 is 0.31.

$$\text{MAR}^{\text{path}} = \frac{1}{2}\left(\frac{|P_A \cap P_B|}{|P_A|} + \frac{|P_A \cap P_B|}{|P_B|}\right) \tag{3.3}$$

When we also consider the implicit links, a path in the closure structure results as a mixture of explicit and implicit links. Unlike MAR^{link}, I treat implicit links the same as explicit links in MAR^{path}. MAR^{path} between closure(A) and closure(B) in Figure 3.8 is 0.57.

Descendant-set-based MAR

The third variant is MAR^{dSet} that measures the agreement based on the existence of the same descendant sets (dSet) in two structures. In contrast with the other two measures, MAR^{dSet} performs its calculations using bigger and more interdependent units. I define the descendant set of node X as the set consisting of the node X itself and its descendants. Figure 3.9 shows an example of the descendant set matching between two annotations. The descendant set in brackets is given below the node ID (which is the sentence position). For example, the descendant set of node 2 of annotation A in Figure 3.9 (left) is $\{2, 3, 4, 5\}$.

I have hypothesised that groups of sentences in an essay operate as one sub-argument. MAR^{dSet} can be seen as a measure of the degree of agreement on such sub-arguments. Two annotations have a high MAR^{dSet} when they group many of the same set of sentences together.

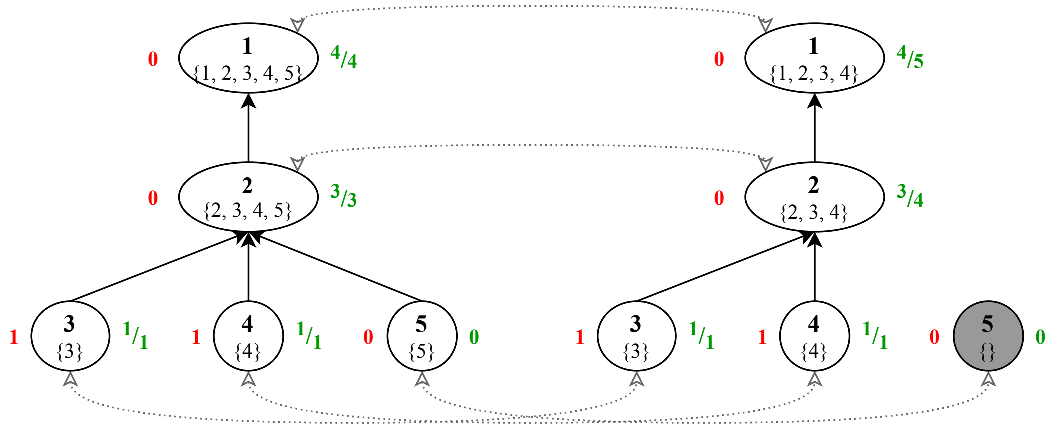


FIGURE 3.9: Example of descendant set matching between annotation A (left) and B (right). Exact-matching scores in red (to the left of each node); partial-matching scores in green to the right. Grey nodes represent non-AC.

There are two types of matching: exact and partial. Under exact matching, a binary score is calculated and two annotations are required to have identical descendant set in order to score a value of 1. For example, the exact matching score for the descendant set rooted in node-2 between annotations A and B in Figure 3.9 is 0. Partial matching, in contrast, returns continuous scores based on the recall of the descendant set of one annotation, calculated with respect to the other annotation. Non-argumentative nodes are counted as a match if they are deemed non-argumentative in both annotations.

A structure is represented by the descendant set matching scores of its nodes in this metric. I define a function f that maps a structure to a vector consisting of descendant set matching scores. For annotation A in Figure 3.9, $f(A) = [0,0,1,1,0]$ when using exact-matching, and $f(A) = [\frac{4}{4}, \frac{3}{3}, \frac{1}{1}, \frac{1}{1}, 0]$ when using partial-matching. MAR^{dSet} is computed as in Equation (3.4), where $\sum$ denotes the summation of vector elements and $|N|$ corresponds to the number of nodes in the structure. It measures the average of average recall.

$$\text{MAR}^{\text{dSet}} = \frac{1}{2}\left(\frac{\sum f(A)}{|N_B|} + \frac{\sum f(B)}{|N_A|}\right) \tag{3.4}$$

MAR^{dSet} scores between annotations A and B in Figure 3.9 are 0.40[12] and 0.76[13] for exact and partial matching, respectively.

I report all three MAR variants because together these structure-based metrics provide us with analytical tools that can measure the agreement on argument paths and descendant sets. For comparison with the literature, I also report the graph-based metric proposed by Kirschner et al. (2015), which is somewhat similar to mine. It measures the extent to which a structure A is included in structure B. The inclusion score I_A is shown in Equation (3.5), where E_A represents the set of links in A; (x, y) denotes two nodes connected by a link; and $\text{SP}_B(x, y)$ is the shortest path between nodes x and y in B.

$$I_A = \frac{1}{|E_A|} \sum_{(x,y) \in E_A} \frac{1}{\text{SP}_B(x, y)} \tag{3.5}$$

The same concept is applicable to measure I_B. This metric measures whether two linked nodes in annotation A also directly or indirectly exist in annotation B. Similar to MAR^{path}, I consider implicit links as if they are explicit when computing Kirschner's metric for closure structures, because a path is a mixture of explicit and implicit links. There are two ways to combine inclusion scores I_A and I_B: by averaging or calculating the F1-score between them. For example, the graph-based agreement scores between two structures in Figure 3.9 are 0.88 (avg.) and 0.86 (F1).

3.3.3 Meta-evaluation of Structure-based Agreement Metrics

If one introduces a new metric, one should evaluate it against human intuition; such an undertaking, as an evaluation of an evaluation metric, is referred to as a "meta-evaluation." I use the crowdsourcing platform Amazon Mechanical Turk (AMT) for the meta-evaluation, and elicit similarity judgements about pairs of human annotations. Workers are asked to judge two different options and to tell us which option represents the higher similarity in this crowdsourcing task. One option compares two argumentative structures for an essay X annotated by two different annotators A and B. The other option compares two structures for a different essay Y, again annotated by A and B. Given these two pairs of two structures (a pair for an essay), workers judge which pair is more similar according to their intuition concerning the composition of the hierarchical structures. They evaluated based on three aspects: placement of nodes in the hierarchical structure, grouping of nodes forming sub-trees and links between nodes.

Figure 3.10 illustrates the AMT task, where numbered nodes represent sentences and arrows represent argumentative relations between sentences. The structures

[12] $\frac{1}{2}\left(\frac{2}{5} + \frac{2}{5}\right)$; average of average sum of the red values in Figure 3.9.

[13] $\frac{1}{2}\,(0.80 + 0.71)$; average of average sum of the green values in Figure 3.9.

Instruction:
You are given two options (option 1 and option 2). Each option contains two figures. Choose the option with more similar figures, considering both the structure and the placement of numbers in the figures.

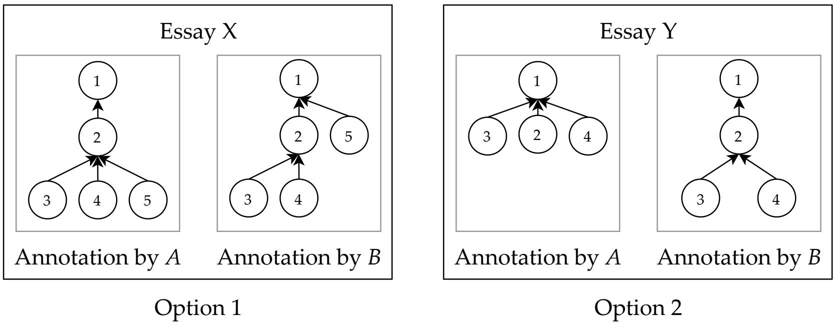


FIGURE 3.10: Illustration of an "AMT task".

shown to workers contain only node IDs and directed links. I replaced undirected links with directed links in order to simplify the task for the crowd workers. I also show the structures without any text or relation labels. This is because the interpretation of the relation labels would require expertise in discourse analysis, which is not available in the crowdsourcing paradigm. Workers therefore also cannot judge whether implicit links should hold or not, and so my evaluation uses scores which are calculated on explicitly annotated links only.

It is difficult to evaluate whether workers provide their responses earnestly in a crowdsourcing experiment. I employ AMT workers who have an approval rating of more than 95% and record a total of 30 votes for each question item. I consider responses that are too fast or too slow to be noises or spams, and to filter them, I remove 5% fastest and slowest responses, leaving us with the 90% of the responses in the middle.

For each AMT task, I count the votes given by crowd workers for Option 1 and Option 2 as V_1 and V_2, respectively. In parallel, I calculate the agreement scores M_1 and M_2, for each option, under each of the metrics M tested here. I compare the agreement ratio and four versions of mine, namely: MAR$^{\text{link}}$, MAR$^{\text{path}}$, MAR$^{\text{dSet}}$ (exact-match) and MAR$^{\text{dSet}}$ (partial-match), and Kirschner's metric.[14] I am the first to provide a meta-evaluation for Kirschner's metric because the original publication did not provide one.

I measure four aspects of evaluation. First, I use *accuracy* to measure whether the metrics' prediction agrees with the majority voting result. When the voting is tied, meaning that the workers have no preference between the two pairs, I also check whether the metric assigns the same score for both pairs. For the second aspect of evaluation, I use *weighted accuracy* (W.Acc.) to simulate the fuzzy nature of human judgement. When a metric assigns a higher score, for example, $M_1 > M_2$, it gains a normalised voting score $\frac{V1}{V1+V2}$. One can interpret this as the probability of the metric being aligned with the workers' preference. Third, I calculate the *minimum squared error* (MSE) between automatically assigned scores and normalised voting differences, that is, between $(M_1 - M_2)$ and $\left(\frac{V_1-V_2}{V_1+V_2}\right)$. This measures whether the metrics can estimate the exact numerical difference of votes. Lastly, I calculate the

[14] It is not possible to evaluate summary-style agreement metrics such as Cohen's κ in this experiment, because κ requires more samples than are available in my experimental setting, as each essay yields only a single data point under κ.

linear correlation between the differences in metric scores and normalised voting differences. A higher score is better for all these evaluations.

I use argumentative structures from 20 randomly chosen ICNALE essays, annotated by two annotators each. Random selection was stratified according to score, country, and prompt. The texts contain 13.3 sentences on average.[15] If each essay's structures are compared with each other essay's structures, $\binom{20}{2}$=190 possible "AMT tasks" results. Given the 30 responses per task, there were a total of 5,700 responses. 5,130 responses remained after I applied the time cutoff described previously.

Metric	Accuracy	W.Acc.	MSE	Correlation
Agreement Ratio	0.65	0.59	0.19	0.43
Kirschner's metric (avg)	**0.75**	**0.64**	0.12	0.67
Kirschner's metric (F1)	**0.75**	**0.64**	0.12	0.67
MAR$^{\text{link}}$	**0.75**	0.63	**0.11**	**0.71**
MAR$^{\text{path}}$	0.74	0.63	0.12	0.64
MAR$^{\text{dSet}}$ (exact-match)	0.71	0.62	0.11	0.68
MAR$^{\text{dSet}}$ (partial-match)	0.70	0.62	0.16	0.59

TABLE 3.3: Meta-evaluation result of structure-based inter-annotator agreement metrics. Best results are written in **bold-face**.

Table 3.3 shows the results of the meta-evaluation. Kirschner's metric and MAR$^{\text{link}}$ achieve the same performance in terms of accuracy. Kirschner's metric achieves the highest performance in W.Acc. (0.64), while my proposed metric, MAR$^{\text{link}}$, achieves the best performance in terms of MSE (0.11) and Pearson's correlation (0.71). The numerical difference between Kirschner's metric (F1) and MAR$^{\text{link}}$ is 0.01 for W.Acc. The difference between MSE of Kirschner's metric (F1) and MAR$^{\text{link}}$ is 0.01. Although MAR$^{\text{link}}$ has a slightly higher correlation value to human judgement compared with Kirschner's metric, the difference is only 0.04. I also note that the agreement ratio performs the worst under all evaluation aspects, with low correlation with human judgements.

MAR$^{\text{link}}$ and Kirschner's metrics are roughly in the same ballpark when it comes to capturing human intuitions, but I still prefer MAR$^{\text{link}}$ because it is able to treat implicit and explicit links differently (although I was not able to test this property in the current experiment). This mechanism is unique among all metrics, and it should be useful for specific purposes.

I have performed a preliminary meta-evaluation of my novel structure-based metrics and Kirschner's metric that shows good results as far as the basic interpretability of these metrics goes; correlation to human judgements is moderate to good. I am now in a position where I can analyse structural agreement using these new metrics, and do so in the rest of this thesis.

3.4 Corpus Construction

This section describes my agreement study and the resulting ICNALE-AS2R corpus from my annotation effort. The corpus consists of 434 ICNALE essays annotated with argumentative structure and sentence reordering.

[15] Meta-evaluation relies on the availability of annotated essays; thus, I performed it chronologically after the agreement studies reported in Section 3.4.1; for this reason, the texts annotated in the agreement studies were reused here.

I report intra- and inter-annotator agreement scores to show that my scheme is stable and reproducible as follows. A scheme is stable if independent annotations by the same person result in a high agreement, and reproducible if independent annotations by different people result in a high agreement.

3.4.1 Intra- and Inter-annotator Structural Agreement

I use the same 20 randomly sampled ICNALE essays as in the meta-evaluation reported in the previous section. They contain a total of 266 sentences, with 3,496 possible pairs of sentences to be linked.

I report agreement scores under closure because, in my opinion, this corresponds most closely to the truth. I also report the scores calculated on explicit links only to allow a comparison with previous argumentation schemes. However, the use of non-closure metrics is not advisable in situations like this where equivalence classes are defined, which negatively affects the metrics' interpretability.

To measure annotation stability, I employ a paid annotator (annotator A), a PhD student in English Education with special expertise in text assessment and discourse analysis and years of experience as an EFL teacher. Although not a native speaker of English, annotator A is very familiar with reading, assessing and improving EFL texts in the course of their daily operations. It is generally accepted that it is not necessary to use English native speakers for experiments in argumentation or discourse studies because the associated tasks require cognition rather than syntactic ability.

I prepared guidelines of 14 pages describing the annotation scheme (cf. Appendix A), which were available to the annotator during annotation, and asked the annotator to annotate 20 essays twice from scratch over the course of a month of interim period. I assumed a month is long enough for the annotator to forget their initial annotation.

Task & Metric	**Explicit**	**Closure**
Argumentative component identification		
Cohen's κ	1.00	-
Agreement Ratio	1.00	-
Linking		
Cohen's κ	0.92	0.89
Kirschner's metric (avg)	0.93	0.91
Kirschner's metric (F1)	0.93	0.91
MAR^{link}	0.92	0.93
MAR^{path}	0.87	0.85
MAR^{dSet} (exact-match)	0.92	0.92
MAR^{dSet} (partial-match)	0.97	0.97
Relation Labeling		
Cohen's κ	0.87	-
Agreement Ratio	0.92	-
Entire Agreement Ratio	0.87	-

TABLE 3.4: Intra-annotator agreement of annotator A.

Table 3.4 shows the the intra-annotation study results. It demonstrates that the annotation is stable.[16] Annotator A has an almost perfect agreement to themselves,

[16] We do not report the linking results using the agreement ratio. It performed badly in the meta-evaluation, and it is known to produce misleadingly high results in tasks where the distribution of

including producing almost exactly the same structures (both explicit and implicit). The confusion matrix in Table 3.5 between the first and second versions of annotations by annotator *A* shows that the only difficulty faced by annotator *A* lay in distinguishing between DETAIL and SUPPORT labels in a few cases.

A(v1) \ A(v2)	RESTATEMENT	ATTACK	DETAIL	SUPPORT
RESTATEMENT	**7**	0	1	0
ATTACK	0	**24**	1	0
DETAIL	0	0	**53**	3
SUPPORT	0	0	12	**121**

Table 3.5: Confusion matrix of annotator *A* in intra-annotator agreement study.

I next perform an inter-annotator agreement study between annotator *A* and myself (annotator *B*) using the same texts as in the intra-annotator study. I compare the first annotation of annotator *A* with annotator *B*'s annotations.

Task & Metric	**Explicit**	**Closure**
Argumentative component identification		
Cohen's κ	0.66	-
Agreement Ratio	0.98	-
Linking		
Cohen's κ	0.53	0.50
Kirschner's metric (avg)	0.63	0.62
Kirschner's metric (F1)	0.63	0.61
MAR^{link}	0.56	0.58
MAR^{path}	0.39	0.37
MAR^{dSet} (exact-match)	0.54	0.54
MAR^{dSet} (partial-match)	0.85	0.85
Relation Labeling		
Cohen's κ	0.61	-
Agreement Ratio	0.77	-
Entire Agreement Ratio	0.47	-

Table 3.6: Inter-annotator agreement results.

A \ B	RESTATEMENT	ATTACK	DETAIL	SUPPORT
RESTATEMENT	**5**	0	0	1
ATTACK	0	**9**	4	0
DETAIL	1	0	**27**	4
SUPPORT	2	1	18	**61**

Table 3.7: Confusion matrix between annotators *A* and *B* in the inter-annotator agreement study.

Table 3.6 shows the inter-annotator agreement scores. The agreement scores on argumentative component identification were measured at Cohen's $\kappa = 0.66$ ($N =$

categories is imbalanced. As is the case in our situation here, the number of sentence pairs that are not linked is far higher than those that are linked.

266, $n = 2$, $k = 2$).[17] There were only 10 ($\sim$4%) and 5 ($\sim$2%) sentences marked as non-ACs by annotators A and B, respectively. Cohen's κ was measured at 0.53 (0.50 on closures; $N = 3,496$, $n = 2$, $k = 2$) for linking, and 0.61 ($N = 133$, $n = 4$, $k = 2$) for relation labelling. Table 3.7 shows that the most frequently confused labels are again DETAIL and SUPPORT.

I manually inspected the cases concerned in the confusion between these labels. One of the possible explanations I discovered is a difficulty in judging whether certain argumentative material is new or not (if it is new, the correct label is SUPPORT; if it is not, DETAIL is correct). Another reason concerns the use of examples, as these can be seen as either elaboration (DETAIL) or actual supporting evidence (SUPPORT). Consider the following excerpt (ICNALE essay W_JPN_PTJ0_005_B2_0_EDIT).

> $_{\text{(S5)}}$ *If they have a part-time job they can learn a lot.* $_{\text{(S6)}}$ *For example: responsibility, hospitability, communication skills, how to solve problems, and so on.*

S6 can be seen as supporting S5 by bringing to light new evidence or elaborating on what can be *learned*, which would make it a DETAIL. One way to mitigate the confusion is to explicitly assign all exemplifications as DETAIL in future guidelines. This would acknowledge that in most cases, examples are used to provide additional detail to the target sentences.

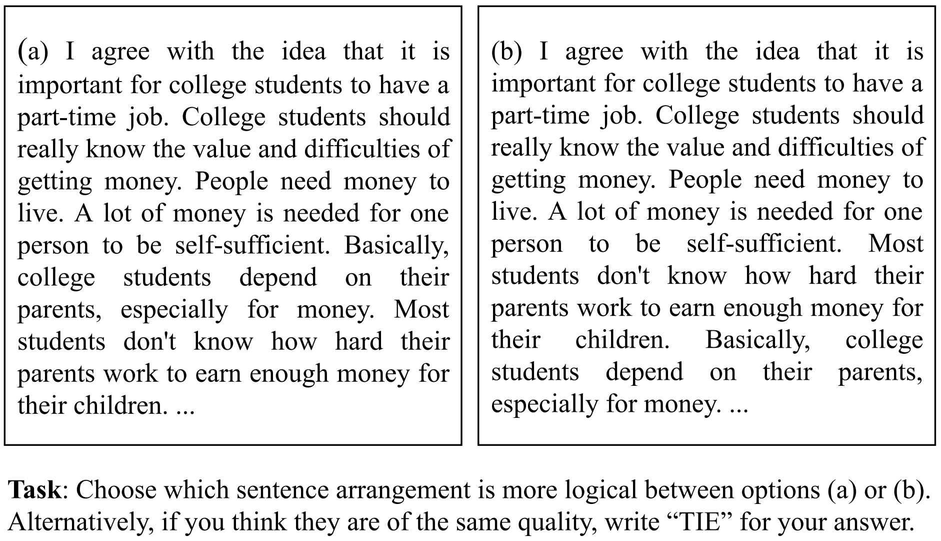
3.4.2 Meta-evaluation of Reordering Annotation

I have described the intra- and inter-annotator agreement analyses for the argumentative structure annotation. Reordering agreement, in contrast, is of secondary importance because existing studies have found that it is not possible to identify a single correct ordering configuration (Barzilay et al., 2002; Todd et al., 2004). Reordering agreement might be low due to the nature of the task but that does not prevent us to improve the essays in one particular way. I perform a secondary meta-evaluation to investigate whether the reordered version produced by annotator A (production annotator) is indeed better than the original one in this section. This gives us insights into whether the reordering operation indeed improve text quality in another way.

In this evaluation, original and reordered versions of the same essay are shown simultaneously. Assessors then judge which version has the more logical sentence arrangement or if they are of the same quality. I need sentences to be as independent as possible of the context in which they originally appear or should be reordered for the evaluation purpose. To this end, I decontextualise sentences by removing connectives at the sentence beginning and resolving intersentential anaphoras with their referents (Wachsmuth et al., 2018). Additionally, I also remove non-AC sentences, since they are discarded from further processing in my scheme (treated as if they do not exist during the reordering annotation). The following snippet shows an example of decontextualisation procedure followed by non-AC removal (essay "W_CHN_ PTJ0_021_A2_0_EDIT"); removed parts are written in strike-through and resolved anaphoras are written in bold. Figure 3.11 illustrates the task.

[17] N denotes the number of items, n is the number of categories and k represents the number of annotators.

(S1:AC)*I agree with the idea that it is important for college students to have a part-time job.* (S2:non-AC)~~*I want to explain the two reasons why I think that college students should have a part time job.*~~ (S3:AC)~~*The first reason is that*~~ *College students should really know the value and difficulties of getting money. ...* (S8:AC)*I think* ~~*they*~~ ***students*** *should know how hard work is needed to get just a little money through a part-time job, as their parents have learned. ...* (S13:AC)~~*For these two reasons,*~~ *I agree with the idea that it is important for college students to have a part-time job.*



(a) I agree with the idea that it is important for college students to have a part-time job. College students should really know the value and difficulties of getting money. People need money to live. A lot of money is needed for one person to be self-sufficient. Basically, college students depend on their parents, especially for money. Most students don't know how hard their parents work to earn enough money for their children. ...

(b) I agree with the idea that it is important for college students to have a part-time job. College students should really know the value and difficulties of getting money. People need money to live. A lot of money is needed for one person to be self-sufficient. Most students don't know how hard their parents work to earn enough money for their children. Basically, college students depend on their parents, especially for money. ...

Task: Choose which sentence arrangement is more logical between options (a) or (b). Alternatively, if you think they are of the same quality, write "TIE" for your answer.

FIGURE 3.11: An illustration of reordering meta-evaluation task (essay "W_ JPN_PTJ0_ 021_B1_2_EDIT.")

Assessor	Original Text	Reordered Text	Tie
X	4	9	11
Y	15	8	1
Z	10	12	2

TABLE 3.8: Meta-evaluation result of reordering annotation. Each row shows how many times the corresponding assessor judges a particular essay version (column) as better than the other version, or if both versions are tied.

I randomly sample 24 reordered essays from the ICNALE-AS2R corpus[18] for this meta-evaluation study, and employ three third-party professional essay assessors with years of experience (X, Y, Z). Table 3.8 shows the evaluation result. In general, assessor X prefers the reordered version of the essays, whereas assessor Y prefers the original version. On the other hand, assessor Z thinks that both versions of essays are similar in quality, although they slightly prefers the reordered version. There is no single essay in which all three assessors agree on their judgement. Inter-annotator agreement (Cohen's κ; $N = 24$, $n = 2$, $k = 3$) scores between assessors are .06, .01 and .05 between X-Y, X-Z and Y-Z pairs, respectively.

[18] This evaluation chronologically happens after the argumentative structure parsing study. In the parsing study, I split the corpus into 80% train and 20% test sets. These 24 reordered essays are taken from the test set.

Consistent with existing studies, I conclude that the best sentence ordering for a text is inherently subjective. This is supported by the fact that third-party assessors do not agree with each other in their judgement. The reordered version of the essays in the ICNALE-AS2R corpus therefore cannot be treated as the best or most correct version. However, there is also no sufficient evidence to reject the reordered versions produced by the expert annotator because assessors X and Z generally go for the reordered version during the meta-evaluation. This suggests that the reordering operation might provide an improvement in text quality to some degree.[19]

A gold standard is required when we consider the development of a computational model. To this end, I train my automatic sentence reordering model to reconstruct the reordered versions of essays as produced by the expert annotator (Chapter 5). The essential research question here is to find the best approach for the reconstruction task.

3.4.3 Description of Resulting Corpus and Qualitative Analysis

Production annotation is performed by annotator *A* on the remaining 414 essays out of 434 ICNALE essays at my disposal (excluding the 20 already used for meta-evaluation and agreement studies).

My final corpus, ICNALE-AS2R, consists of 434 essays: 414 production essays + 20 essays from the inter-annotator study. It is the annotations by annotator *A* that are used throughout, and there are two reasons for this. First, I consider annotator *A* as the expert in the subject area because they are a discourse analyst and an EFL teacher. Second, I expect to avoid my own (or other project members') bias and ensure the consistency of the annotation by employing an external expert annotator.

	All	Max/$_{\text{essay}}$	Min/$_{\text{essay}}$	Avg./$_{\text{essay}}$	SD
Size					
Sentences	6,021	28	6	13.9	3.3
Tokens	111,394	360	191	256.7	32.1
Arg. components	5,799	25	6	13.4	3.1
Non-arg. components	222	6	0	0.5	0.9
Relation and Structure					
Support	3,029	18	1	7.0	2.5
Detail	1,585	14	0	3.7	2.5
Attack	437	6	0	1.0	1.3
Restatement	314	4	0	0.7	0.6
Structure Depth	-	11	1	4.3	1.4

TABLE 3.9: Statistics of the ICNALE-AS2R corpus. Sentences and tokens are automatically segmented using nltk (Bird et al., 2009). SD stands for standard deviation.

The corpus consists of 6,021 sentences in total, containing 5,799 (96.3%) ACs and 222 (3.7%) non-ACs (cf. Table 3.9). Argumentative structures in the corpus have an average depth of 4.3 (root at depth 0). SUPPORT is the most commonly used relationship (3,029 instances–56.5%), followed by DETAIL (1,585–29.5%), ATTACK (437–8.1%) and RESTATEMENT (314–5.9%). This distribution is unsurprising given that students are often explicitly taught to write supporting reasons for their arguments.

[19] To provide a stronger and more definitive claim, we have to conduct a large scale meta-evaluation and see assessors' judgements converge to the preference for the reordered version.

The number of RESTATEMENT relations is lower than the number of essays, meaning that some student arguments do not contain any conclusion statements anywhere.

We next look at how far related sentences are separated from each other. Adjacent links predominate (56.5%) in the ICNALE-AS2R corpus. Short-distance links (2 ≤separation≤ 4) make up 23.7% of the total. On the other hand, long distance links (5 ≤separation≤ 26) make up 19.8%.

Overall, the source sentence succeeds the target sentence in textual order (or in other words, the link was *backward*) in 78.5% of directed relations. The EFL students predominantly tend to use the "claim–support" structure, in which an opinion is stated first and then its evidence is presented afterwards. Again, this is expected, as argumentative writing in English is often taught in this way (Bacha, 2010). Table 3.10 shows the ratio of backward and forward links for each directed relation type. The backward direction is strongly preferred over the forward direction for SUPPORT and ATTACK labels. The DETAIL label stands out because the preference between forward and backward direction is not as strong as the other labels.

	Support		**Detail**		**Attack**	
Backward	2538	(83.8%)	1040	(65.6%)	386	(88.3%)
Forward	491	(16.2%)	545	(34.4%)	51	(11.7%)

TABLE 3.10: Distribution of relation direction in the ICNALE-AS2R corpus.

	Adjacent		**Short-distance**		**Long-distance**	
Student	658	(49.4%)	359	(27.0%)	315	(23.7%)
Expert	750	(56.3%)	293	(22.0%)	289	(21.7%)

TABLE 3.11: Distribution of distance between related sentences before (student version) and after reordering (expert version) in 105 reordered essays.

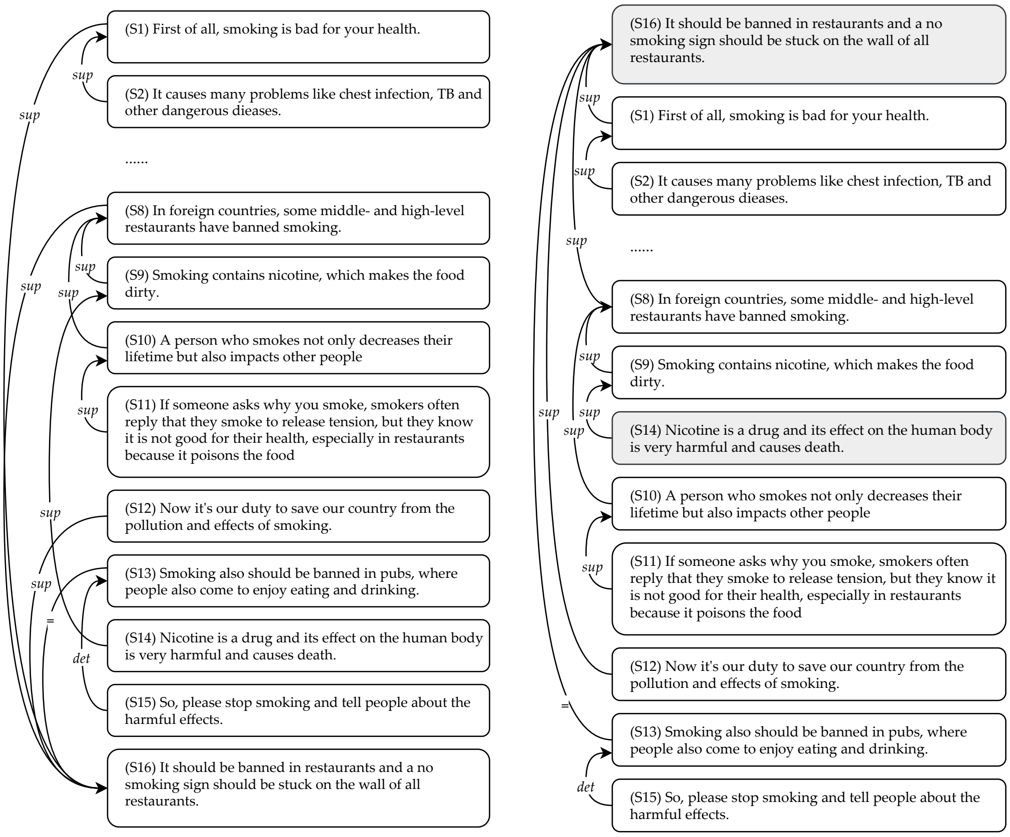
	Support		**Detail**		**Attack**	
Backward	2575	(85.0%)	1042	(65.7%)	394	(90.2%)
Forward	454	(15.0%)	543	(34.3%)	43	(9.8%)

TABLE 3.12: Distribution of relation direction after reordering in the ICNALE-AS2R corpus.

We now turn our attention to how the reordering step was performed by the expert annotator. Annotator *A* performed the reordering operation in 24.2% of the essays (105 out of 434). One to three sentences are usually moved when reordering happens. Table 3.11 shows the change of distribution between related sentences before and after reordering in these 105 essays. The number of adjacent links rose from 49.4% to 56.3% in these essays, whereas the number of short and long-distance links falls. This suggests that reordering brings related sentences closer to each other. The text repair operation is performed on 181 sentences, 123 (68.0%) of which are attempts to repair the prompt-type error of the major claim. The remaining 58 sentences concern changes in connectives and referring expressions. Table 3.12 shows the ratio of backward and forward links for each directed relation type after reordering. Similar to the students, the expert also prefers the backward direction over the forward direction for SUPPORT and ATTACK labels.

3.4.4 Qualitative Analysis

My annotation enables for the identification of potential argument-related problems. For example, 31 essays (7.1%) essays written by EFL students contain more forward than backward relations. This contradicts the typical writing preference for argumentation. These essays tend to present evidence and supporting material at the beginning of the text, followed by the major afterwards. I consider this an example of a potential problem in English argumentative writing. Other cases exist in which a considerable amount of background information is presented before the start of the argument proper, another potential argument-related problem.



(S1) First of all, smoking is bad for your health.

(S2) It causes many problems like chest infection, TB and other dangerous dieases.

......

(S8) In foreign countries, some middle- and high-level restaurants have banned smoking.

(S9) Smoking contains nicotine, which makes the food dirty.

(S10) A person who smokes not only decreases their lifetime but also impacts other people

(S11) If someone asks why you smoke, smokers often reply that they smoke to release tension, but they know it is not good for their health, especially in restaurants because it poisons the food

(S12) Now it's our duty to save our country from the pollution and effects of smoking.

(S13) Smoking also should be banned in pubs, where people also come to enjoy eating and drinking.

(S14) Nicotine is a drug and its effect on the human body is very harmful and causes death.

(S15) So, please stop smoking and tell people about the harmful effects.

(S16) It should be banned in restaurants and a no smoking sign should be stuck on the wall of all restaurants.

(S16) It should be banned in restaurants and a no smoking sign should be stuck on the wall of all restaurants.

(S1) First of all, smoking is bad for your health.

(S2) It causes many problems like chest infection, TB and other dangerous dieases.

......

(S8) In foreign countries, some middle- and high-level restaurants have banned smoking.

(S9) Smoking contains nicotine, which makes the food dirty.

(S14) Nicotine is a drug and its effect on the human body is very harmful and causes death.

(S10) A person who smokes not only decreases their lifetime but also impacts other people

(S11) If someone asks why you smoke, smokers often reply that they smoke to release tension, but they know it is not good for their health, especially in restaurants because it poisons the food

(S12) Now it's our duty to save our country from the pollution and effects of smoking.

(S13) Smoking also should be banned in pubs, where people also come to enjoy eating and drinking.

(S15) So, please stop smoking and tell people about the harmful effects.

a: Original essay.

b: A potential improvement for Figure 3.12a.

FIGURE 3.12: An excerpt of annotation for essay "W_PAK_SMK0_022_B1_1_EDIT."

Figure 3.12a shows an annotation example. Sentence S16 has been identified by annotator *A* as the clearest statement of the major claim in this figure; it, therefore, becames the root of the structure. Prescriptive writing guidance for argumentation (Bacha, 2010; Silva, 1993) would advise putting such a sentence early in the text.[20] However, the EFL student placed it at the end of the essay.

[20] Note that there is also a less clear formulation of the major claim in S13, which also contains some additional argumentative material. The annotator indicated the similarity with a restatement relation between S13 and S16, but decided that S16 is the best major claim. This in a way indicates too that there

Another indicator of a problem is crossing links in the structure. Crossing links might indicate coherence breaks in texts because argumentative relations typically hold between sentences stating similar ideas. Ideally, few or no crossing links should occur if all sentences constituting a sub-argument are presented together. For example, the topic of both sentences S9 and S14 in Figure 3.12a is nicotine, but the discussion on this topic is interrupted by several sentences discussing different topics. If sentences S9 and S14 were placed close to each other, we can expect an improvement in the textual coherence of the essay.

Figure 3.12b shows the reordered version of the essay by the expert annotator in a way that would be consistent with the previous discussion–sentence S16 has been moved to the beginning of the essay,[21] and sentences S9 and S14 are now adjacent. The improved text is more consistent with the argumentative development strategy in prescriptive writing guidelines; it first introduces a topic, then states its stance on that topic, supports its stance by presenting detailed reasons and finally concludes the essay at the end (Bacha, 2010; Silva, 1993).

To investigate whether the presence of crossing link is intolerable in argumentation, I count the number of *projective* (without crossing links) and *non-projective* (with crossing links) structures in the reordered 105 essays in comparison with their original versions.[22] Table 3.13 shows the result, indicating that projective structures are preferred. There are 28 cases in which non-projective structures remain non-projective even after reordering. I found that the number of crossing links decreased in 15 out of those 28 cases.

Original \ Reordered	Projective	Non-projective
Projective	31	1
Non-projective	45	28

TABLE 3.13: The change of projective and non-projective structures before and after reordering.

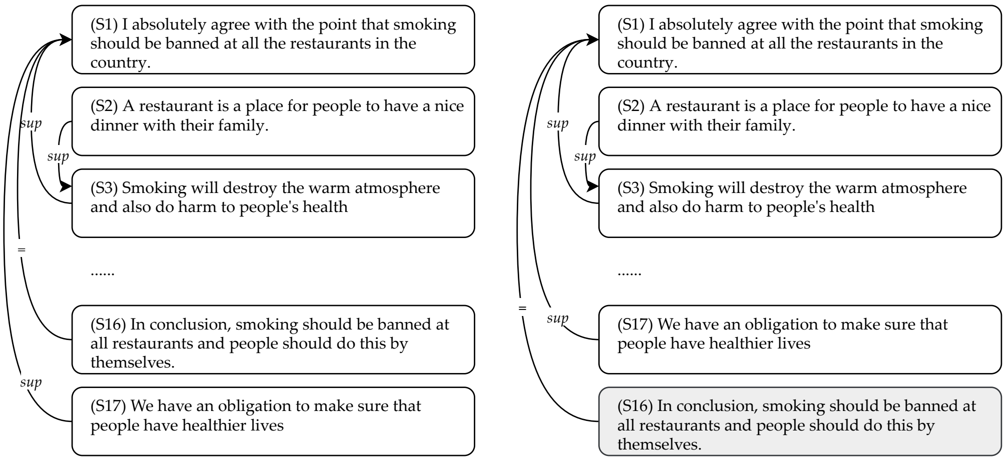
However, an essay is not guaranteed to be problem-free, even if the major claim is placed at the beginning and there are no crossing links. The essay in Figure 3.13a is one such case–S1 is its major claim, and S16 restates it, acting as the conclusion at the end. There are no crossing links. However, sentence S17, which supports the major claim, appears after S16. According to prescriptive guidance, reasons supporting the major claim should be placed before the concluding statement. Therefore, S17 should be placed somewhere between sentences S1 and S16, as shown in Figure 3.13b.

The qualitative analysis revealed that the argumentative structure annotation can provide us with objective means of essay improvement, by indicating both potential problems and better sentence rearrangements that can lead to a better-structured text. This means that the annotated essays in the ICNALE-AS2R can be used as illustrative examples in a teaching session. I observe several reordering strategies that are commonly used in prescriptive classroom teaching by manually inspecting

is a problem; we normally assume that the real major claim occurs before its restatement. However, the directional aspect cannot be explicitly expressed in my notation, as restatements are semantically undirected.

[21] Note that the anaphora starting the sentence should ideally be replaced in the final version too. Because my scheme only instructs repairing potentially ambiguous anaphora, the expert annotator left it as it is.

[22] The terms *projective* and *non-projective* are borrowed from the syntactical dependency parsing field.


(S1) I absolutely agree with the point that smoking should be banned at all the restaurants in the country.
(S2) A restaurant is a place for people to have a nice dinner with their family.
(S3) Smoking will destroy the warm atmosphere and also do harm to people's health
......
(S16) In conclusion, smoking should be banned at all restaurants and people should do this by themselves.
(S17) We have an obligation to make sure that people have healthier lives
sup
sup
=
sup
(S1) I absolutely agree with the point that smoking should be banned at all the restaurants in the country.
(S2) A restaurant is a place for people to have a nice dinner with their family.
(S3) Smoking will destroy the warm atmosphere and also do harm to people's health
......
(S17) We have an obligation to make sure that people have healthier lives
(S16) In conclusion, smoking should be banned at all restaurants and people should do this by themselves.
sup
sup
=
sup

a: Original essay. b: A potential improvement for Figure 3.13a.

FIGURE 3.13: An excerpt of annotation for essay "W_CHN_SMK0_045_A2_0_EDIT."

the changes in reordered texts. For example, background information is moved to a place before the main argument and the conclusion is moved toward the end. Sentences logically belonging to the same sub-argument are gathered together. Within sub-arguments (sub-trees), the root or claim is often moved to the beginning, and further explanations, supporting evidence and examples typically follow after the root.

3.5 Chapter Summary

In this chapter, I presented a new annotation scheme for argumentative structure and sentence reordering in EFL essays. The annotation effort results in a collection of 434 annotated essays, called the "ICNALE-AS2R" corpus. The agreement study showed that my proposed argument annotation scheme is stable, with the near-perfect intra-annotator agreement and reasonable inter-annotator agreement for the structural annotation. I also evaluated whether the reordered version of essays in the ICNALE-AS2R corpus is indeed better than the original ones. A secondary meta-evaluation for the reordering annotation was performed, employing three third-party assessors. One assessor preferred the reordered version, one slightly preferred the reordered version and the other one preferred the original version. Therefore, the reordered version of the essays in the corpus cannot be treated as the best or most correct one. Yet, there is no sufficient evidence to reject the reordered version as well. As far as supervised machine learning is concerned, the reordered version can nevertheless be used for training sentence reordering models.

The annotated corpus also comes with some additional methodological and technical contributions. I developed MAR, a novel structure-based agreement metric to provide a more holistic view of the structural annotation. The metric comes in three variants, which differ in how the unit of analysis was defined. The first variant, MAR^{link}, is useful in the case where we want to differentiate between implicit and explicit links. The second variant, MAR^{path}, measures agreement on argumentation chains. The last variant, MAR^{dSet} calculates agreement based on the presence of the same substructures. A meta-evaluation study via crowdsourcing showed that all MAR variants achieved a high correlation with human judgement.

My qualitative analysis revealed that the argumentative structure annotation can indicate potential problems existing in the texts. The second layer of annotation in the ICNALE-AS2R corpus provides sentence reordering that may lead to a more coherent text. Thus, the corpus in itself is useful to provide examples during a teaching session.

The annotation was entirely performed using a newly developed, web-based (client-side) annotation tool TIARA. It provides versatile visualisation for analysing argumentative structure and reducing clutter in the display. While the tool is of course designed to fulfil my annotation needs, it also supports general discourse structure annotation and educational use cases. This makes it advantageous compared with existing tools. TIARA can also be customised easily via a configuration script to accommodate a wide range of annotation schemes.

I turn to the task of argumentative structure parsing in the next chapter. The argumentative structure is useful for text analysis in itself as explained above. It is also an input to my sentence reordering module (cf. Chapter 5).

Chapter 4

Argumentative Structure Parsing

In this chapter, I describe parsing models for extracting the argumentative structure of essays in the ICNALE-AS2R corpus. The argumentative structure parsing consists of two steps: (i) a ***sentence linking*** step where I identify related sentences that should be linked, forming a tree structure, and (ii) a ***relation labelling*** step, where I label the relationship between the sentences. I do not only evaluate the model performance based on individual links but also perform structural analysis, giving more insights into the models' ability to learn different aspects of argumentation. For a new scheme and corpus, it can be advantageous to look at intermediate results even though a pipeline system may fall prey to error propagation. I first experiment with SotA models used in other studies as base models and analyse their performance. Then, I propose a multi-task learning extension using two structural-modelling auxiliary tasks to improve the sentence linking performance.

My second contribution is exploring the possibility of multi-corpora training for AM. In the past, well-written and less well-structured texts have been treated as two separate domains, and AM systems were trained separately on each domain. Here, I investigate how far the existing labelled corpora for well-written texts can also be useful for training parsers for less well-structured texts. To this end, I train the base models on both in-domain and out-domain texts and evaluate them on the in-domain task. I use the reordered versions of EFL texts as our out-domain texts. Furthermore, I study the possibility of a multi-corpora training strategy for texts annotated using different schemes and of different quality but from the same genre. Here, I use two learner essays corpora: the PEC (cf. Section 2.2.1) and the ICNALE-AS2R corpus.

4.1 Base Models

4.1.1 Sentence Linking Task

Given an essay as a sequence of sentences $s_1, ..., s_N$, a sentence linking model outputs the distance $d_1, ..., d_N$ between each sentence s_i to its target; if a sentence is connected to its preceding sentence, the distance is $d = -1$. I consider those sentences that have no explicitly annotated outgoing links as linked to themselves ($d = 0$); this concerns major claims (roots) and non-ACs. I also operate RESTATEMENT links as directed relations for computational purposes.[1]

Table 4.1 shows the distance distribution between the source and target sentences in the corpus in this formulation, ranging $[-26, ..., +15]$. Adjacent links predominate (50.4%). Short-distance links ($2 \leq |d| \leq 4$) make up 21.2% of the total. Backward

[1] Recall that the difference between the directed and undirected link in TIARA is a matter of visualisation and interpretation, and not of computation. RESTATEMENT links are operationalised as directed relations to eliminate circular links that are not allowed in my scheme.

≤ -5	-4	-3	-2	-1	0	$+1$	$+2$	$+3$	$+4$	$\geq +5$
16.6	3.9	5.2	8.3	37.0	10.9	13.4	2.3	0.9	0.6	1.0

TABLE 4.1: Distribution of distance (in percent) between source and target sentences in the corpus.

long distance links at $d \leq -5$ are 16.6%, whereas forward long distance links are rare (1.0%). Self-loop makes up 10.9% of the total.

Following recent advances in AM (cf. Section 2.3), I model the argumentative structure parsing as sequence tagging and as dependency parsing tasks.

Sequence Tagger Model

Figure 4.1 shows my sequence tagging architecture (SEQTG). I adapt the vanilla BiLSTM with softmax prediction layers (as Eger et al. (2017) similarly did).

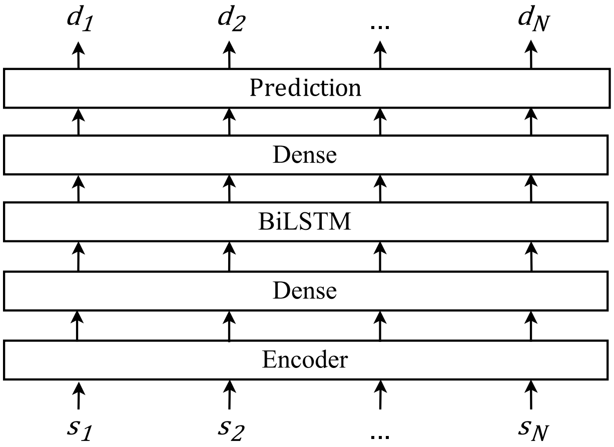


FIGURE 4.1: BiLSTM-softmax (SEQTG).

The input sentences $s_1, ..., s_N$ are first encoded into their respective sentence embeddings, using either BERT (Devlin et al., 2019) or sentence-BERT (SBERT, Reimers and Gurevych (2019)) as encoder.[2] I do not perform fine-tuning when using the BERT encoder because my dataset is too small for it.[3] SBERT is a modified version of pre-trained BERT, which is specifically designed to derive semantically meaningful sentence embedding without further fine-tuning. I use the SBERT variant that is trained on the natural language inference (NLI) task in this thesis. The idea of training embeddings on the NLI task goes back to Conneau et al. (2017). It involves recognising textual entailment (TE), and a TE model has been previously used by Cabrio and Villata (2012) for argumentation. The resulting sentence embeddings are then fed into a dense layer for dimensionality reduction. The results are fed into a BiLSTM layer ($\#stack = 3$) to produce contextual sentence representations and then fed into a prediction layer.

The model predicts the probability of link distances, in the range $[-26, ..., +15]$. I perform a constrained argmax during prediction time to make sure that there is no out-of-bound prediction. For each sentence s_i, I compute the argmax only for distances at $[1-i, ..., N-i]; i \geq 1$. The model is trained using a cross-entropy loss.

[2] By averaging subword embeddings.

[3] I conducted a preliminary BERT fine-tuning experiment on the sentence linking task, but the performance did not improve. This is most probably because of the dataset size.

Biaffine Attention Model

Figure 4.2 shows my dependency parsing architecture (BIAF). I adapt the biaffine attention model (Dozat and Manning, 2017), treating the sentence linking task as sentence-to-sentence dependency parsing (Morio et al., 2020).

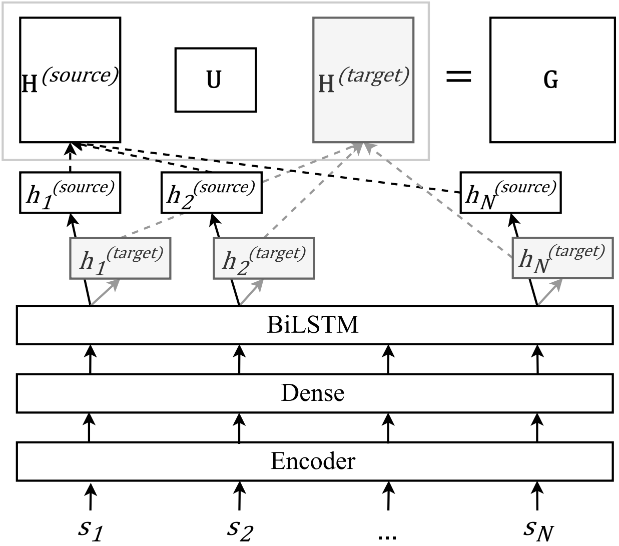


FIGURE 4.2: Biaffine attention model (BIAF).

The first three layers produce contextual sentence representations in the same manner as in the SEQTG model. These representations are then passed into two different dense layers to encode the corresponding sentence when it acts as a source $\left(h^{(source)}\right)$ or target $\left(h^{(target)}\right)$ in a relation. Finally, a biaffine transformation (Dozat and Manning, 2017) is applied to all source and target representations to produce the final output matrix $\mathbf{G} \in \mathbb{R}^{N \times N}$, where each cell $g_{i,j}$ represents the probability (score) of the source sentence s_i pointing to s_j. Equations 4.1 and 4.2 show the detail of the biaffine transformation, where $\mathbf{U}$ and W are weight matrices and b is a bias. I train the BIAF model using a max-margin criterion (Kiperwasser and Goldberg, 2016).

$$\text{Biaff}(x_1, x_2) = {x_1}^{\mathrm{T}}\mathbf{U}x_2 + \mathbf{W}(x_1 \oplus x_2) + b \tag{4.1}$$

$$g_{i,j} = \text{Biaff}\left(h_i^{(source)}, h_j^{(target)}\right) \tag{4.2}$$

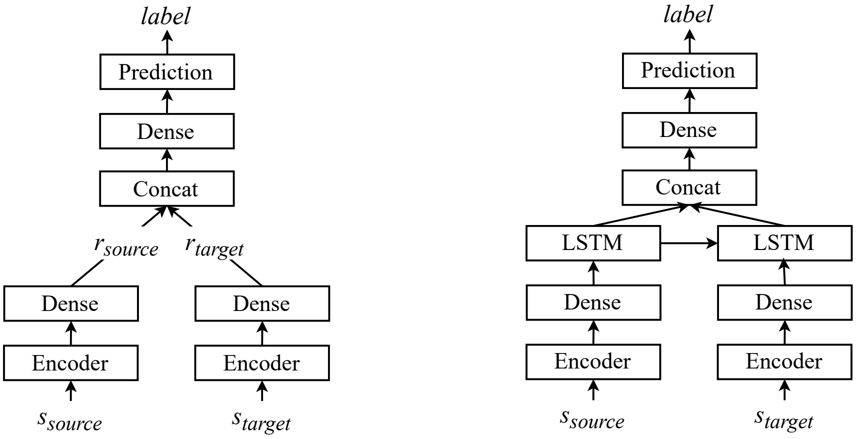
When only considering the highest scoring or most probable target for each source sentence in isolation, the outputs of the models (SEQTG and BIAF) do not always form trees (30-40% non-tree outputs in my experiment). To this end, I apply Chu-Liu-Edmonds algorithm (Chu and Liu, 1965; Edmonds, 1967) to create a minimum spanning tree out of the output.

4.1.2 Relation Labelling Task

In the relation labelling task, given a pair of *linked* source and target sentences $\langle s_{source}, s_{target}\rangle$, $s_{source} \neq s_{target}$, a model outputs the label that connects them, that is, one of {SUPPORT, ATTACK, DETAIL, RESTATEMENT}. I use non-fine-tuning models with feed-forward architecture and fine-tuning transformer-based language models.

Non-fine-tuning Models

In non-fine-tuning models, both source and target sentences $\langle s_{source}, s_{target} \rangle$ are encoded using BERT or SBERT to produce their respective embeddings. I then pass these embeddings into respective dense layers for a dimensionality reduction and transformation step, producing $\langle r_{source}, r_{target} \rangle$. As the first option (FFCON, Figure 4.3a), r_{source} and r_{target} are concatenated, passed to a dense layer for a further transformation, and finally passed into a prediction layer. As the second option (FFLSTM, Figure 4.3b), I feed r_{source} and r_{target} to an LSTM layer, and the hidden units of LSTM are concatenated before being sent to a dense layer (Deguchi and Yamaguchi, 2019).



a: FFCON model.

b: FFLSTM model.

FIGURE 4.3: Non-finetuning relation labelling models.

Fine-tuning Models

Unlike in the sentence linking task, where an entire essay is taken as input, the relation labelling task takes a pair of sentences. There are 5,365 of such pairs in the ICNALE-AS2R. I fine-tune BERT and DISTILBERT (Sanh et al., 2019) for the relation labelling task because there are more instances than in the sentence linking task. The $\langle s_{source}, s_{target} \rangle$ pair is fed into the transformer model, and then the [CLS] token representation is passed into a prediction layer (Figure 4.4). All relation labelling models are trained using cross-entropy loss.

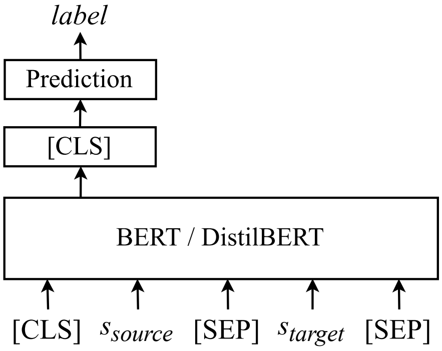


FIGURE 4.4: Fine-tuning relation labelling models.

4.1.3 Experimental Result and Discussion

The ICNALE-AS2R corpus is split into 80% train set (347 essays; 4,841 sentences) and 20% test set (87 essays; 1,180 sentences), stratified according to prompts, scores and country of origin of the EFL learners. I am interested in how the AM models trained on well-written texts would fare on less well-structured texts. To find out, I train the models on both the original EFL texts (in-domain) and the parallel improved texts (out-domain), and then evaluate them on the original EFL texts. The difference between in- and out-domain data lies on the textual surface, that is, sentence rearrangement, the use of connectives, referring expressions and textual repair for major claims. The out-domain data is roughly 76% the same as the in-domain data (81 essays were reordered in the train set, and 24 in the test set) because not all essays undergo any reordering.

The number of hidden units and learning rates (alongside other implementation notes) to train the models can be found in Appendix B. I run the experiment for 20 times,[4] and report the average performance. The relation labelling models are trained and evaluated using sentence pairs according to the gold-standard. In the end-to-end evaluation (Section 4.1.4), however, the input to the relation labelling model is the automatic prediction. Statistical testing, whenever possible, is conducted using the permutation test (Noreen, 1989) on the performance scores of the 20 runs with a significance level of $\alpha = .05$.

Sentence Linking

I first report in-domain before turning to the cross-domain results.

Before going to the standard evaluation metrics, I first evaluate the global shape properties of the models' outputs. The gold standard trees have a particular shape, expressed as average depth of 4.3 (SD $=$ 1.4) and leaf ratio of .439 (SD $=$ 0.11). It is hard to believe that the models did learn the structure of texts when both average depth and leaf ratio of the models' outputs digress too far from the gold standard (we must consider both metrics as they are correlated). Both extremes are undesirable: a leaf ratio close to 0 indicates a linear chain, where each sentence only points to its preceding adjacent sentence; whereas a leaf ratio close to 1 indicates a tree of depth one, where each leaf is directly pointing at the root.

Table 4.2 shows the output shape of the parsing models. All models tend to produce trees that are deeper and narrower than the gold standard, particularly when using the SBERT encoder. Having said that, the deviation of the shape is still within the standard deviation of the gold trees, notably the average depth. Therefore, we can confirm that all models produce structures that are relatively similar to the gold annotations.

Table 4.3 shows the experimental result on the prediction of individual links. The best model is a biaffine model, namely SBERT-BIAF, statistically outperforming the SEQTG models (accuracy .471 vs .444 and F1-macro .323 vs .274; significant difference on both metrics).

To gain deeper insights into model quality, I also considered the models' F1 score per distance (Figure 4.5). All models, and in particular BIAF, are better at predicting long-distance links ($d \leq -5$, BIAF avg. F1 = .41) than short distance links ($2 \leq |d| \leq 4$, BIAF avg. F1 = .17) when using the SBERT encoder (the same trend goes when

[4] 20 experiments are repeated on the same dataset split. This is to account for random initialisation in neural networks. Readers may refer to Reimers and Gurevych (2017) for a more detailed explanation.

Model	Average Depth	Leaf Ratio
Gold	4.3±1.4	.439±.11
BERT-SeqTg	4.4	.430
BERT-Biaf	4.7	.424
SBERT-SeqTg	4.8	.409
SBERT-Biaf	5.1	.404

Table 4.2: Output shape of in-domain sentence-linking models.

Model	Accuracy	F1-macro
BERT-SeqTg	.436	.274
BERT-Biaf	<u>.446</u>	<u>.310</u>
SBERT-SeqTg	.444	.229
SBERT-Biaf	**.471**†	**.323**†

Table 4.3: In-domain results of individual-link predictions in the sentence linking task. The best result is shown in **bold-face**. The † symbol indicates that the difference to the second-best result (underlined) is significant.

using the BERT encoder). Long-distance links tend to happen at the higher tree level, for example, the links from nodes at depth ′ to the root, while short-distance links tend to happen at the deeper level, for example, within a sub-argument at depth ≥ 2. Figure 4.6 shows the model performance across depths, that is, whether the model places each node at the depth it is supposed to be. This plot indicates that the model performance declines as one moves further down the tree. Nodes at depth 0 (major claims) are arguably easy to identify because they are often marked with discourse markers, such as *"in my opinion"*, *"I strongly believe that"* and *"I argue that"*. Supporting opinions at depth 1 are also indicated by discourse markers, such as *"firstly"* and *"secondly"*. We argue that the relatively good performance on depths 0 and 1 c As deep structur ıore.

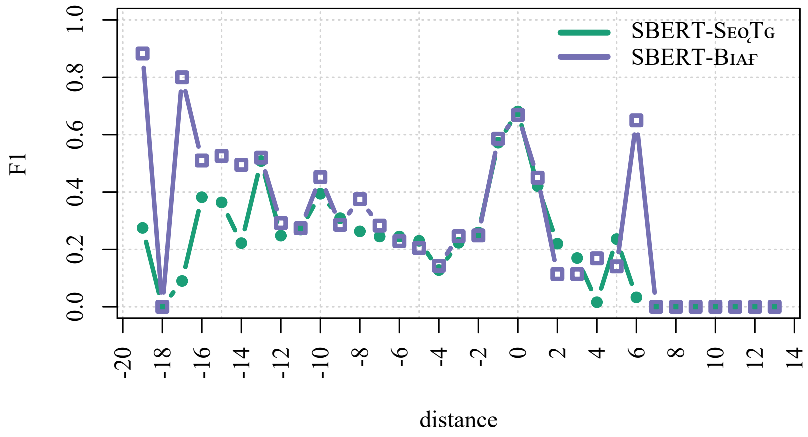


Figure 4.5: Model performance across distances for in-domain evaluation using SBERT encoder.

We next look at the models' ability to perform *quasi* argumentative component type (QACT) classification. My scheme does not assign AC roles per se, but we can

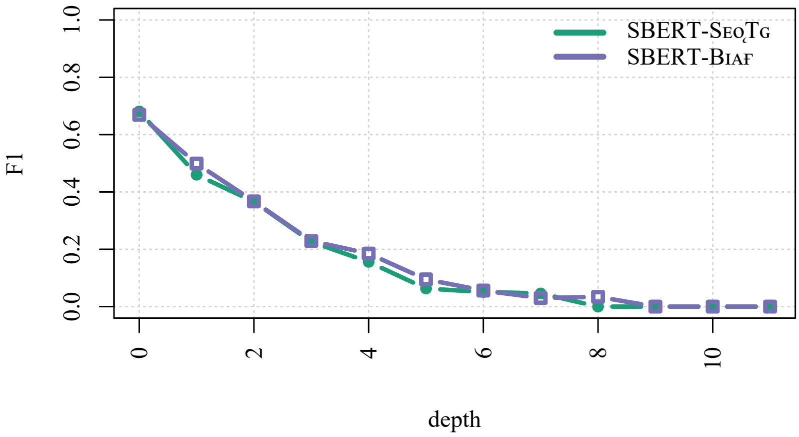


FIGURE 4.6: Model performance across depths for in-domain evaluation using SBERT encoder.

compile the following sentence types from the tree typology:

- *major claim (root)*: only incoming links,
- *AC (non-leaf)*: both outgoing and incoming links,
- *AC (leaf)*: only outgoing links and
- *non-AC*: neither incoming nor outgoing links.

This QACT scheme evaluates whether the models place sentences properly in the hierarchical structure and whether they have the desired link properties. Table 4.4 shows the results.

Model	**Major claim**	**AC (non-leaf)**	**AC (leaf)**	**non-AC**	**F1-macro**
BERT-SeqTg	.695	.603	.584	**.486**	.592
BERT-Biaf	**.730**†	.609	.573	.058	.493
SBERT-SeqTg	.705	.616	.590	.471	.596
SBERT-Biaf	**.730**†	**.639**†	**.599**†	.437	**.601**

TABLE 4.4: In-domain results of *quasi* argumentative component type classification (node labels identified by topology). This table shows F1 score per node label and F1-macro. **Bold-face**, †, and underline as above.

SBERT-Biaf model performed the best (F1-macro = .609). I notice that the Biaf model works only when paired with the SBERT encoder. When using the BERT encoder, it has great difficulty in producing *any* non-AC nodes at all (Non-AC F1 = .058; F1-macro = .493) despite good performance on individual links. This result seems to suggest that SBERT is a better encoder than BERT for non-fine-tuning models. This also demonstrates the necessity of evaluating AM models beyond standard metrics, such as in terms of structural characteristics, as I show here. Individual link prediction performance does not ensure the overall structure's quality.

I next look at the cross-domain performance of the best sentence linking model, namely, SBERT-Biaf. It achieves an accuracy of .459 and an F1-macro of .270 for the prediction of individual links. The F1-macro for QACT classification is .565. These scores are somewhat lower compared with the in-domain performance (significant difference). This means that the modifications of even 25% of essays (in terms of reordering) in the out-domain data may greatly affect the linking performance in the cross-domain setting.

Relation Labelling

	Support	Detail	Attack	Restatement	F1-macro
(B)-FFCon	.698	.433	.282	.594	.502
(B)-FFLSTM	.695	.434	.277	.600	.502
(S)-FFCon	.719	.479	.372	.558	.532
(S)-FFLSTM	.722	**.481**	.396	.574	.543
DistilBERT	.741	.426	.431	.631	.557
BERT	**.760**†	.468	**.478**†	**.673**†	**.595**†

TABLE 4.5: In-domain relation labelling results, showing F1 score per class and F1-macro. "(B)" for BERT and "(S)" for SBERT encoders. **Bold-face**, underline and † as above.

Table 4.5 shows the experimental results for the in-domain relation labelling task when gold-standard links are used. Fine-tuned BERT model achieves the significantly best performance (F1-macro = .595). Non-fine-tuning models performed better when using the SBERT than BERT encoder (F1-macro =.532 vs .502; .543 vs .502; both having significant difference). This further confirms the promising potential of SBERT and might suggest that the NLI task is suitable for pre-training a relation labelling model.

We can see from the results that the ATTACK label is the most difficult one to predict correctly, presumably because of its infrequent occurrence in the dataset. However, the RESTATEMENT label, which is also infrequent, is relatively well predicted by all models. I think that has to do with models' ability to recognise semantic similarity. Recall that the RESTATEMENT label is used when a concluding statement rephrases the major claim. SUPPORT and DETAIL are often confused. Note that they are also the most confusing labels between human annotators. Sentence pairs that should be classified as having ATTACK and RESTATEMENT labels are also often classified as SUPPORT.

I also do a cross-domain experiment for this task. The best relation labelling model, BERT, achieves a cross-domain F1-macro of .587 (the difference is not significant to the in-domain performance). Although it is not currently shown, the change of performance in other models are also almost negligible (up to 2% in F1-macro).

4.1.4 End-to-end Evaluation

For end-to-end evaluation, I combine the best models for each task into a pipeline system: SBERT-Biaf for sentence linking and fine-tuned BERT for relation labelling.

Table 4.6 shows the evaluation results of the average of 20 runs. Accuracy measures whether the pipeline system predicts all of the following correctly for each source sentence in the text: the correct component category (AC vs non-AC), the correct target distance and the correct relation label. In addition, I also calculated the Cohen's κ score between the system's output and the gold annotation for annotation subtasks in my scheme.

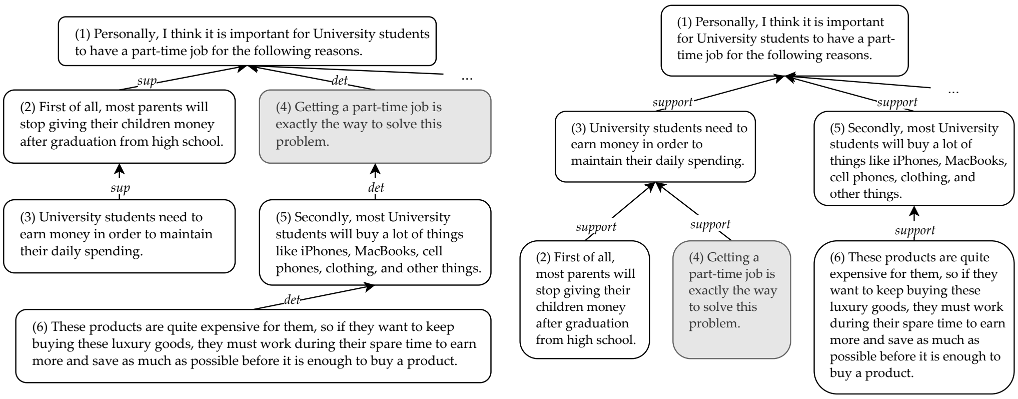
The accuracy of the in-domain system is .341, and that of the cross-domain system .321 (significant difference). There is still a relatively big performance gap when compared with human performance on all metrics (in the agreement study). The

	Accuracy	ACI	SL	RL
Inter-annotator agreement	.474	.66	.53	.61
In-domain	.341	.42	.41	.43
Cross-domain	.321	.36	.40	.39

TABLE 4.6: End-to-end results. Cohen's κ scores are used for ACI (argumentative component identification), SL (sentence linking) and RL (relation labelling).

cross-domain system is able to perform at 94% of the in-domain performance in an end-to-end setting. As this performance drop might well be acceptable in many real-world applications, this signals the potential of training an AM model for less well-structured texts using the annotated corpora for well-written texts alongside those more infrequent annotations for less well-structured texts, at least as long as the genre stays the same.

I also perform an error analysis on several random end-to-end outputs. The system has a tendency of failing to identify the correct major claim when it is not placed at the beginning of the essay. For example, the major claim may be pushed until the middle of the essay when it contains a lot of background information on the discussion topic. Cultural preferences might also be a factor. It has been often observed that reasons for a claim are presented before in writings by Asian students, not after the claim as is more common in Anglo-Saxon cultures (cf. Section 2.1.2). There might be inconsistencies if some EFL learners followed the Anglo-Saxon style and some followed the writing style in their native languages. Recurrent neural networks, which are particularly sensitive to order, can be expected to be thrown off by such effects.


(1) Personally, I think it is important for University students to have a part-time job for the following reasons.

(2) First of all, most parents will stop giving their children money after graduation from high school.

(4) Getting a part-time job is exactly the way to solve this problem.

(3) University students need to earn money in order to maintain their daily spending.

(5) Secondly, most University students will buy a lot of things like iPhones, MacBooks, cell phones, clothing, and other things.

(6) These products are quite expensive for them, so if they want to keep buying these luxury goods, they must work during their spare time to earn more and save as much as possible before it is enough to buy a product.

(1) Personally, I think it is important for University students to have a part-time job for the following reasons.

(3) University students need to earn money in order to maintain their daily spending.

(5) Secondly, most University students will buy a lot of things like iPhones, MacBooks, cell phones, clothing, and other things.

(2) First of all, most parents will stop giving their children money after graduation from high school.

(4) Getting a part-time job is exactly the way to solve this problem.

(6) These products are quite expensive for them, so if they want to keep buying these luxury goods, they must work during their spare time to earn more and save as much as possible before it is enough to buy a product.

a: System's output.

b: Gold structure

FIGURE 4.7: An example snippet of the in-domain system output and its gold structure for essay "W_HKG_PTJ0_021_B1_1."

Another source of error concerns placing a sub-argument into the main argument's sibling position instead of that of its child. In general, the systems also have some issues with clustering, that is, they split a group of sentences that should belong together into separate sub-arguments or, conversely, group together sentences that do not belong together. Figure 4.7 illustrates this problem. In the gold structure, sentence (4) points at (3), forming a sub-argument (sub-tree) of {2, 3, 4}. However, the system puts sentence (4) in the inappropriate sub-tree. This kind of case often happens at group boundaries.

I also found that the system may erroneously use the RESTATEMENT label when connecting claims (at depth = 1) and major claims, particularly when the claims include almost all tokens that are present in the major claim. I suspect that our model learned to depend on lexical overlaps to recognise RESTATEMENT as this type of relation concerns paraphrasing. However, I am unable to perform an error analysis to determine how this has affected the performance on each of the other relation labels, which involve entailment and logical connections.

Looking at the overall results, the main challenge of argumentative structure parsing lies in the sentence linking task. The models seem to stumble when confronted with the hierarchical nature of arguments. Since the outputs of sentence linking prediction will be given as inputs for the subsequent relation labelling model, incorrect link predictions will result in incorrect end-to-end predictions. Thus, the system needs improvement concerning the hierarchical arrangement of sentences in the structure in order to move forward.

4.2 Multi-task and Multi-corpora Training Strategies to Enhance Sentence Linking Performance

I propose several approaches to improve the sentence linking performance, particularly in terms of the grouping of sentences as sub-arguments. I propose to extend the biaffine attention model using a novel set of tasks in the multi-task learning (MTL) setup. I also propose a multi-corpora training strategy using the persuasive essay corpus (PEC, cf. 2.2) to increase training data. This experiment is conducted using the same ICNALE-AS2R 80% train (347 essays) and 20% test (87 essays) splits as previously mentioned in Section 4.1.3.

4.2.1 Multi-task Learning Extension

Figure 4.8 shows the BIAF architecture in the MTL setup. Similar to the base BIAF model, sentences are first encoded in their vector form (embedding). I only use SBERT as the choice of the encoder in the current experiment as it generally worked better than the BERT encoder for the BIAF model in the previous experiment. I also use a sentence position (spos) as an input feature because it has been proved to be useful in other studies (e.g., Song et al., 2020). The spos encoding is calculated by dividing a sentence position by the essay length. Sentence embeddings and spos encoding are then concatenated, passed to a dense and then a BiLSTM (#*stack* = 3) layer.

There are three tasks for the BIAF MTL model, the sentence linking as the main task and two structural-modelling related auxiliary tasks. In contrast with existing studies (cf. Section 2.3), I opt for a new type of auxiliary tasks instead of the more common joint AM formulation (MTL of AM subtasks) or discourse and rhetorical auxiliary tasks. My auxiliary tasks are advantageous in that they do not require additional annotation. The first auxiliary task is the prediction of *quasi*-argumentative-component type (QACT) for each input sentence, that is, one of major claim, AC (non-leaf), AC (leaf) and non-AC, which is automatically identified from the tree topology (cf. Section 4.3). The QACT prediction task should help the model to learn the placement of sentences in the hierarchical structure, as well as the property of links for each sentence.

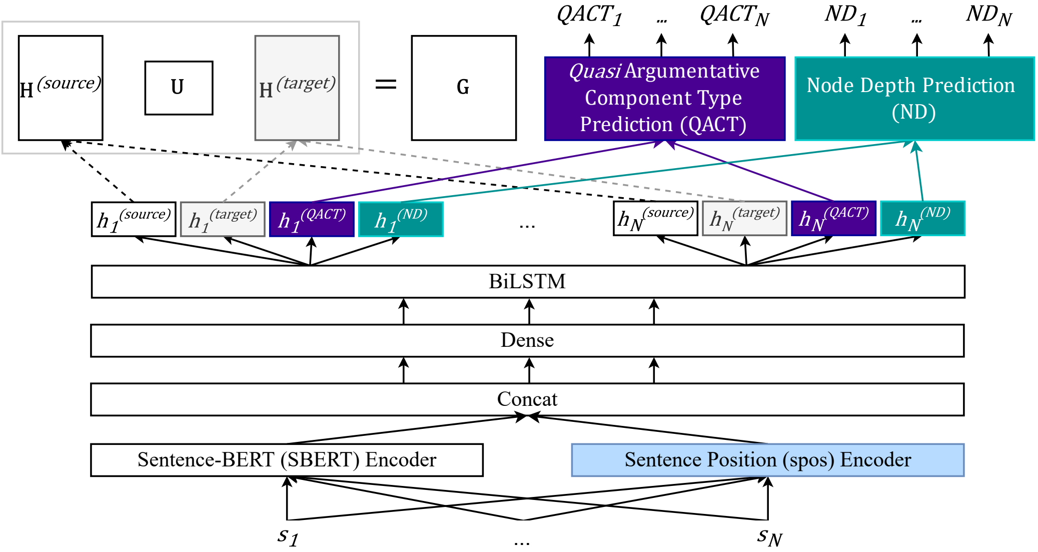


FIGURE 4.8: Multi-task learning extension for the biaffine attention model (BIAF). Newly added modules are coloured.

The second auxiliary task concerns node depth (ND) prediction. There are six depth categories employed: depth 0 to depth 4, and depth 5+. The argumentative structure is hierarchical, and there are no relations between nodes of the same depth. The ND prediction task should also help the model to learn the role of sentences in the hierarchical structure, and provide a guidance where each sentence should point at, that is, sentences at depth X point at sentences at depth $X - 1$.

The MTL loss is defined in Equation (3.1), where the loss L_t of each task t is dynamically weighted and controlled by a learnable parameter σ_t (Kendall et al., 2018). The loss for the main task is computed using the max-margin criterion, while losses for auxiliary tasks are computed using the cross-entropy.

$$L = \sum_t \frac{1}{2\sigma_t^2} L_t + \ln(\sigma_t) \tag{4.3}$$

4.2.2 Multi-corpora Training

Training deep learning models requires a huge amount of data. To this end, I consider the use of the PEC essays (cf. Section 2.2) as additional training data. PEC also provides argumentative essays written by students and represents argumentative structures as trees. However, different to the ICNALE-AS2R, there is no information on the essay authors proficiency (L1 or L2) nor the observed quality of the PEC essays. Hence, these two corpora might be of different quality. Nevertheless, the previous iteration of the parsing experiment (cf. Section 4.3) has shown that a cross-domain (cross-quality) system can perform at 94% of the in-domain system. This signals that we can use other existing corpora to train a parser that works for the ICNALE-AS2R corpus despite the possible difference in essay quality.

There are two settings when training using multiple corpora. The first is to use the entire 402 essays in the PEC, on top of 347 essays (80% train split) from the ICNALE-AS2R, resulting in 749 training essays (12,162 sentences, "[P+I]" setting). However, PEC and ICNALE-AS2R are different in terms of essay length and annotation scheme. The PEC essays have 18.2 sentences on average (15.1 ACs and 3.1

non-ACs), whereas ICNALE-AS2R essays have 13.9 sentences on average (13.4 ACs and 0.5 non-ACs). The difference in non-ACs proportion between these corpora is likely caused by the difference in the set of relation labels employed. The essays in PEC were annotated using two relation types: SUPPORT and ATTACK, while ICNALE-AS2R additionally uses DETAIL and RESTATEMENT. To this end, there is more information in the ICNALE-AS2R corpus. Particularly, some sentences that should have been annotated as non-ACs (hence not linked to other sentences) in the PEC might have been annotated as ACs in the ICNALE-AS2R by using the additional relations.

Because of the differences in annotation scheme and statistical properties, I suspect that the model might not properly learn the distribution of ICNALE-AS2R in the [P+I] setting, that is, the *distributional shift* problem (cf. Section 2.3). To this end, I also propose a second *selective sampling* ("[SS]" setting) strategy, to account for the differences in essay length and annotation scheme, that is, minimising the distributional shift. I only use PEC essays that are "somewhat similar" to those of ICNALE-AS2R's considering the following heuristics in this setting.

- Having 17 sentences at maximum (ICNALE-AS2R avg. 13.9 + 3.3 SD)
- Containing 2 non-ACs at maximum (ICNALE-AS2R avg. 0.5 + 0.9 SD)

There are 110 remaining PEC essays after selective sampling. Hence, the number of training instances for the [SS] setting is $110 + 347 = 457$ essays (6,418 sentences).

ADU in the ICNALE-AS2R is also annotated at the sentence level. However, PEC is annotated at the clause level. Therefore, I convert the PEC annotation to the sentence level, following the strategy described by Song et al. (2020). I use the whole sentence as an AC if a sentence contains only one AC; I split it into multiple sentences if a sentence contains two or more ACs, while including the preceding connective to each AC. The following example (Song et al., 2020) illustrates the splitting procedure (PEC essay075), where a sentence containing three ACs is split into three sentences (annotated AC segments are written in bold).

> $_{\text{(S1)}}$*[To conclude, **art could play an active role in improving the quality of people's lives,**]* $_{\text{(S2)}}$*[but I think that **governments should attach heavier weight to other social issues such as education and housing needs**]* $_{\text{(S3)}}$*[because **those are the most essential ways enable to make people a decent life.**]*

4.2.3 Experimental Result and Discussion

The current experiment aims to improve the sentence linking performance, particularly concerning the grouping of sentences as sub-argument. A metric is needed to quantify the improvement on this aspect, and I propose to use MAR$^{\text{dSet}}$ (exact match, cf. Section 3.3.2) to do so. Recall that MAR$^{\text{dSet}}$ quantifies the similarity of two structures based on the presence of the same substructures. Therefore, it is suitable for the goal of the analysis.

In the current experiment, I also perform an ablation study using the forward selection method to understand the contribution of newly proposed components (i.e., MTL tasks, spos feature and multi-corpora training) to the overall performance. The experiment is conducted in the in-domain setting (using essays arranged by students). I consider the SBERT-BIAF from the previous experiment as a baseline (hereafter, simply referred to as "BIAF" because I use SBERT as the encoder for all models).

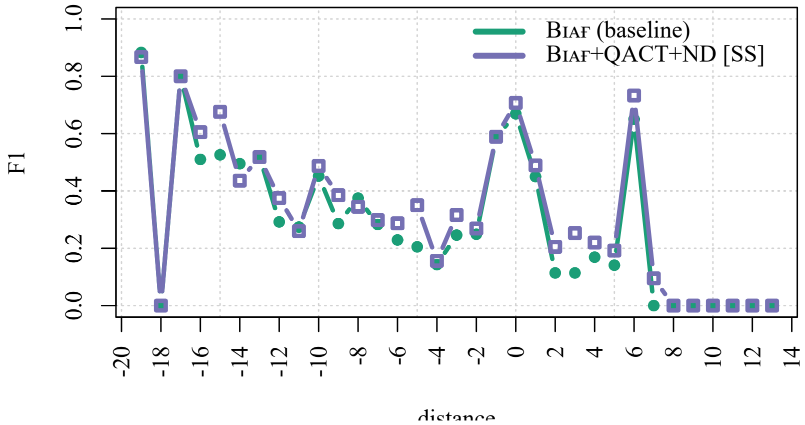
Model	**Accuracy**	**F1-macro**	**MAR$^{\text{dSet}}$**
Baseline			
BIAF	.471	.323	.419
MTL			
BIAF+QACT	.473	.333	.422
BIAF+QACT+ND	.472	.338	.423
Spos			
BIAF+QACT+spos	.472	.327	.421
BIAF+QACT+ND+spos	.475	.336	.426
Multi-corpora Training			
BIAF+QACT+ND [P+I]	.468	.360	**.455**
BIAF+QACT+ND [SS]	**.489**†	**.374**†	.452

TABLE 4.7: Results of individual link (accuracy and F1-macro) and substructure predictions in the sentence linking experiment (with rich supervision signals). The best result is shown in **bold-face**. The † symbol indicates that the difference to the second-best result (underlined) is significant.

Table 4.7 shows the experimental result in terms of prediction of individual links and MAR$^{\text{dSet}}$. Training the BIAF model using the QACT auxiliary task results in improvement of performance over the baseline, particularly in terms of F1-macro (not significant difference). In addition, using both QACT+ND auxiliary tasks significantly improved the performance over the baseline in terms of F1-macro. This signals that the proposed MTL setting benefits model performance.

We next look at how spos encoding affects the model performance. Introducing spos to the BIAF+QACT+ND model improves accuracy and MAR$^{\text{dSet}}$. However, the difference is not significant. Similarly, the difference between BIAF+QACT+spos and BIAF+QACT is not significant across all metrics. There are two possible explanations for this phenomenon. First, students may organise their texts inconsistently, that is, some texts may have been written in the "claim-support" structure, but some are written in the "support-claim" structure. Sentences on the same topic might also be separated to each other (cf. Section 3.4.3). These inconsistencies and noise might have negated the effect of the spos encoding. Second, the output of the BIAF model is a graph **G** which considers the directed link between all pairs of sentences. The spos feature might not affect the biaffine transformation much in this context.

The models trained using multiple corpora attain the best performance for individual link and substructure predictions (Table 4.7). BIAF+QACT+ND [P+I] achieves the best performance of .455 in terms of MAR$^{\text{dSet}}$, and BIAF+QACT+ND [SS] achieves the best performance of .489 and .374 in terms of accuracy and F1-macro, respectively. The [SS] model also achieves the second-best performance of .452 in terms of MAR$^{\text{dSet}}$. These improvements are significant over the baseline and the BIAF+QACT +ND MTL model. Note that when using the [SS] setting, the model performance is consistently improved concerning all metrics, whereas the accuracy of the [P+I] model is lower than the the baseline. In general, the [SS] model attains a better performance compared with the [P+I] model, despite fewer training instances. This means that when training a model using multiple corpora, it is essential to consider training instances having the same properties as our goal. Simply having more training instances does not guarantee improvements.



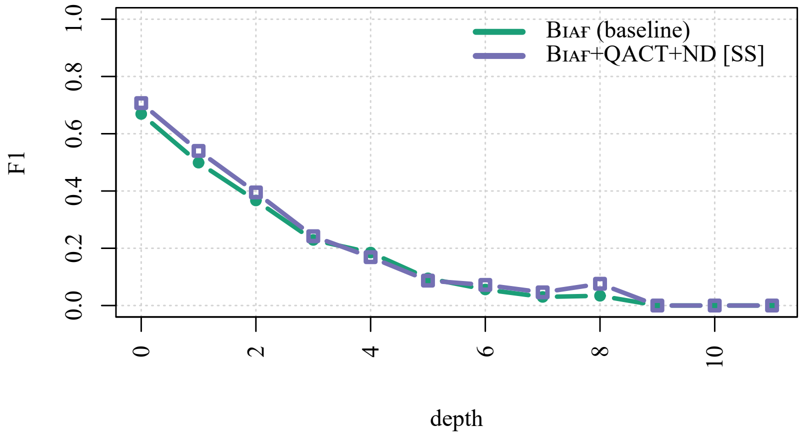


FIGURE 4.10: Model performance across depths.

To gain deeper insights into the link-prediction improvement brought by the [SS] model over the baseline, Figure 4.9 shows F1 score per target linking distance. BIAF+QACT+ND [SS] is better than the baseline model, particularly at predicting short-distance links ($2 \leq |d| \leq 4$, avg. F1 = .24 vs .17). Yet, this is still the weakest range even for the [SS] model. Figure 4.10 shows the model performance across depths, that is, whether the model places each node at the proper depth in the predicted structure. BIAF+QACT+ND [SS] performs better than the baseline particularly in [0,3] and [6,8] ranges. The performance of both models still degrades at the deeper tree levels.

We next look at the models' ability to perform the QACT prediction in the structure predicted by the main task. Table 4.8 shows the result. The MTL models perform better compared with the baseline model. Both BIAF+QACT and BIAF+QACT +ND achieve a significant improvement over the baseline in terms of non-AC prediction and F1-macro. This reconfirms that both of my proposed MTL tasks are useful to improve sentence linking performance. Similar to the previous result on individual links, the spos encoding does not provide much help. I notice that the BIAF+QACT+ND [P+I] model performs worse compared with the baseline, particularly in terms of AC (non-leaf) and non-AC predictions. On the other hand, BIAF+ QACT+ND [SS] achieves the best performance of .622 in F1-macro, which is a significant improvement over other configurations. This confirms my hypothesis that the

Model	**Major Claim**	**AC (non-leaf)**	**AC (leaf)**	**Non-AC**	**F1-macro**
Baseline					
BIAF	.730	.639	.599	.437	.601
MTL					
BIAF+QACT	.739	.639	.601	.453	.608
BIAF+QACT+ND	.734	.636	.601	.454	.606
Spos					
BIAF+QACT+spos	.725	**.641**	.602	.438	.602
BIAF+QACT+ND+spos	.738	.638	.603	.460	.610
Multi-corpora Training					
BIAF+QACT+ND [P+I]	748	.606	**.634**†	.420	.602
BIAF+QACT+ND [SS]	**.767**†	.633	.628	**.462**	**.622**†

TABLE 4.8: Results of *quasi* argumentative component type classification (based on the predicted topology). This table shows F1 score per node label and F1-macro. **Bold-face**, † and underline as above.

difference between annotation schemes and statistical properties between PEC and ICNALE-AS2R affects the model in terms of the distribution learned. The proposed solution to perform selective sampling helps to alleviate this problem.

Model	**Average Depth**	**Leaf Ratio**
Dataset		
ICNALE-AS2R	4.3±1.4	.439±.11
PEC [ALL]	2.8±.6	.540±.09
PEC [SS]	2.7±.5	.565±.08
Baseline		
BIAF	5.1	.404
MTL		
BIAF+QACT	5.1	.410
BIAF+QACT+ND	5.0	.418
Spos		
BIAF+QACT+spos	5.2	.407
BIAF+QACT+ND+spos	5.0	.412
Multi-corpora Training		
BIAF+QACT+ND [P+I]	4.1	.486
BIAF+QACT+ND [SS]	**4.5**	**.446**

TABLE 4.9: Structural-output quality of sentence-linking models with rich supervision signals. The closest value to the ICNALE-AS2R gold standard is written in bold.

I also analyse the overall shape of the predicted structures by all models, as shown in Table 4.9. The gold standard trees in ICNALE-AS2R have a particular shape, expressed as the average depth of 4.3 (SD = 1.4) and the leaf ratio of .439 (SD = 0.11). The baseline model tends to produce trees that are deeper and narrower than the ICNALE-AS2R gold standard, and this is still true for models trained in the MTL setting and with spos encoding. My MTL auxiliary tasks help to improve the leaf ratio to become closer to the gold standard, while spos embedding does not provide additional improvement. When I introduce the multi-corpora training strategy, the predicted structures become shallower compared with the baseline. I believe this is due to the shallower trees in the PEC. When using the [P+I] setting, the model predicts trees that are shallower and wider compared with the essays in the ICNALE-AS2R corpus (the distributional shift problem). However, this is less problematic for the [SS] model, as it produces the most similar structure to the ICNALE-AS2R essays. I conclude BIAF+QACT+ND [SS] as the best model in this experiment because it produces consistently better performance compared with other configurations across all evaluation aspects.

	Accuracy	**ACI**	**SL**	**RL**
Inter-annotator agreement	.474	.66	.53	.61
SL:BIAF & RL:BERT	.341	.42	.41	.43
SL:BIAF+QACT+ND [SS] & RL:BERT	.357	.44	.43	.45

TABLE 4.10: End-to-end results. Cohen's κ scores are used for ACI (argumentative component identification), SL (sentence linking) and RL (relation labelling).

Finally, I analyse how the improvement in the sentence linking task affected the overall argumentative structure parsing task. To this end, I combine BIAF+QACT +ND [SS] for sentence linking and fine-tuned BERT model for relation labelling into a pipeline system. Table 4.10 shows that the improved pipeline achieves better performance compared with the base parser (accuracy .357 vs .341, significant difference). Since a better sentence linking model provides more correct inputs for the subsequent relation labelling model, the agreement between automatic prediction and the gold standard has also been improved from .43 to .45 for the relation labelling task.

4.3 Chapter Summary

In this chapter, I proposed several deep learning models for argumentative structure parsing tasks in EFL essays. I used a pipelined neural approach, consisting of sentence linking and relation labelling steps. Experimental result shows that the biaffine model combined with the SBERT encoder performs the best in the sentence linking task at the F1-macro of .323. The fine-tuned BERT model achieved the best performance at the F1-macro of .595 in the relation labelling task. I also evaluated my base parser on a cross-domain setting, where training is performed on both in-domain (students' original essays) and out-domain (reordered texts) data, and evaluation is performed on the in-domain test data. I found that the best cross-domain system achieved 94% of the in-domain system in terms of end-to-end performance. This signals the potential to use well-written texts together with less well-structured texts to increase the size of training data.

In the subsequent step, I investigated multi-task and multi-corpora training strategies for the sentence linking task. I proposed structural-modelling-related auxiliary tasks that require no additional annotation, to provide a richer supervision signal. Also, I proposed a multi-corpora training strategy to increase training data size. However, it has to be noted that simply increasing the training data does not guarantee improved performance. We need to ensure that the system indeed models the desired target distribution. To this end, corpora of different genres or annotated using different schemes have to be used and adapted selectively. Both these strategies improved the sentence linking model performance to the F1-macro of .374 from .323 for individual link predictions. The parsing performance was also improved to .357 from .341 in terms of end-to-end accuracy.

In the next chapter, I describe how the argumentative structure can be utilised in the downstream task of sentence reordering.

Chapter 5

Automatic Sentence Reordering

I present a novel computational task of sentence reordering to provide discourse-level feedback to language learners. I will train different reordering models on the essays in the ICNALE-AS2R corpus. The goal is to rearrange sentences into a well-structured text for a given sequence of sentences in sub-optimal order. In our context, the well-structured text is defined as the final version of the text produced by the expert annotator. This final version could include reordering or not. The model should also recognise whether the input essay is already well-structured, and do nothing if so; that is, retain the original order.

The sentence reordering task is different from the existing sentence ordering task (cf. Section 2.4). The sentence ordering task aims to find a coherent sequence for a given set of randomised sentences, whereas the sentence reordering task assumes prior order information in the input. A sentence ordering model's output will be better or at least equal to the randomised input in terms of the overall quality. However, a reordering system needs to select the right sentences and move them in appropriate positions to generate an output of higher quality than the input. Failure to do so renders the reordering system impractical.

In the CL community, the validity of using a single gold standard has been questioned in tasks where there is a substantial amount of subjectivity, such as summarisation, machine translation and discourse analysis. There are more than one acceptable solution in these circumstances. The meta-evaluation of reordering annotation (cf. Section 3.4.2) confirmed that this is the case for the sentence reordering task as well. However, it is unclear how to form an evaluation strategy in this situation. Ideally, one would collect as many different solutions as there are, and evaluate systems by proximity to either one of the solutions. However, it is often not feasible to collect all possible solutions from human annotators. Therefore, many CL tasks simply rely on a single gold standard out of necessity, despite knowing that this situation is less than ideal. For instance, acquiring an exhaustive list of all possible variations of translation for a given text is arguably not possible because of real-life constraints (Fomicheva et al., 2020).[1] The same holds in summarisation tasks, where many studies evaluate their systems using only a single reference summary (e.g., Rush et al., 2015). I do the same here, so I do not propose that I can solve this problem in this thesis. Considering the final version of annotated essays in the ICNALE-AS2R corpus as one possible gold standard, I investigate the ability of computational approaches to perform the sentence reordering task as defined by *this particular* gold standard.

I propose a novel method to reorder sentences in EFL essays based on the results of a previous step of argumentative structure analysis. Following Grosz and

[1] The most important of these constraints is the cost involved in creating multiple reference translations.

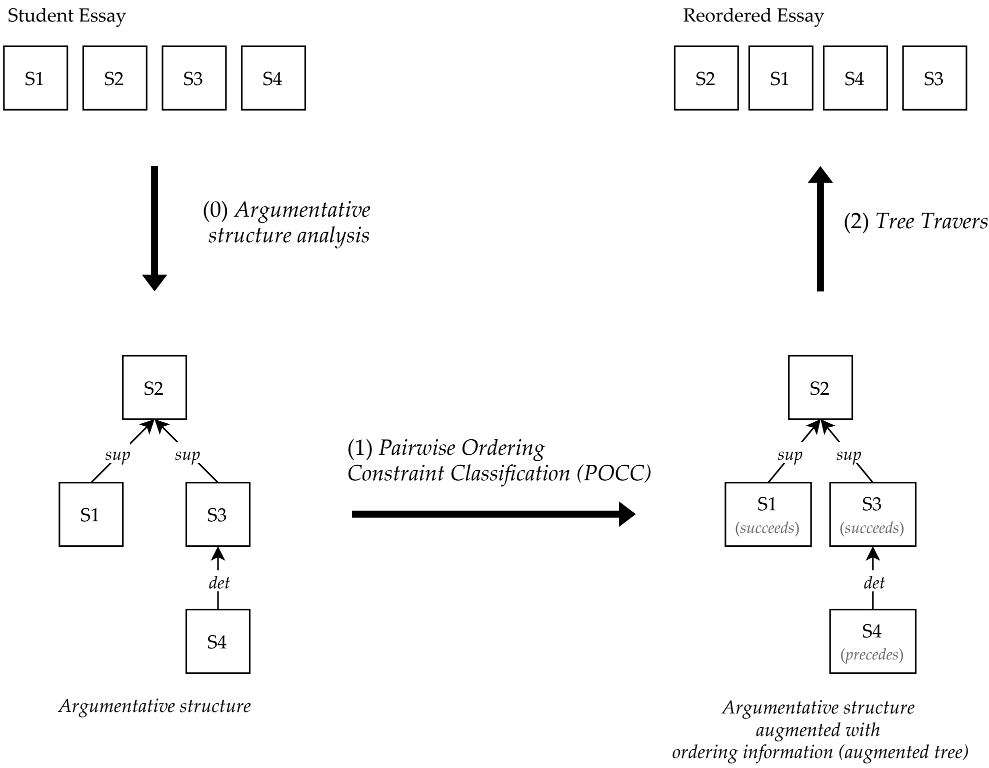

Student Essay
S1
S2
S3
S4
Reordered Essay
S2
S1
S4
S3
(0) *Argumentative structure analysis*
(2) *Tree Traversal*
S2
sup
sup
S1
S3
det
S4
(1) *Pairwise Ordering Constraint Classification (POCC)*
S2
sup
sup
S1 (*succeeds*)
S3 (*succeeds*)
det
S4 (*precedes*)
Argumentative structure
Argumentative structure augmented with ordering information (augmented tree)

FIGURE 5.1: My sentence reordering approach.

Sidner's (1986) theory of coherence (cf. Section 2.1.1), I hypothesise that the argumentative structure provides a guidance to arrange sentences in text. The argumentative structure tells us which sentences should be moved to some other location in order to improve the overall quality of the text (cf. Section 3.4.3). I also investigate how the quality of automatic argumentative structure analysis affects the pipelined reordering step in this chapter. To this end, I perform experiments where the inputs are automatically predicted argumentative structures, gold standard structures and randomly generated structures.

I formulate the sentence reordering task as a tree-traversal problem. Given a text and its corresponding argumentative structure, reordering is performed in two steps. The first step is ***pairwise ordering constraint classification*** (POCC), which considers a pair of sentences that are connected by an argumentative relation, and decides the relative order between them. This relation is analogous to Grosz and Sidner's *satisfaction-precedence* relation. The second step is ***tree traversal***, where I generate a text by traversing the argumentative structure that has been augmented with the pairwise ordering information. Figure 5.1 illustrates my approach. The following section explains each step of my reordering pipeline in detail.

5.1 Proposed Architecture

5.1.1 Pairwise Ordering Constraint Classification

Given a pair of source and target sentences that are connected by an argumentative relation $\langle s_{source}, s_{target} \rangle$, this step decides whether the source sentence *precedes* or *succeeds* the target sentence in linear order. The classification results are then recorded in the argumentative structure. The bottom-right structure in Figure 5.1 shows such an augmented tree; the additional node labels "(*succeeds*)" and "(*precedes*)" indicate whether the particular node should precede or succeed its target sentence. I consider neural methods to perform this classification task and compare the performance with doing nothing.

Neural Network Classifier

I present several deep learning classifiers to perform the POCC task. Given a pair of sentences $\langle s_{source}, s_{target} \rangle$, the models are trained to decide upon the pairwise ordering between the input pair, using the final version of essays as training material.

I fine-tune transformer-based language models, BERT and ALBERT (Lan et al., 2020), on the sentence pair classification task. ALBERT is a particularly great choice for the task at hand since it is pre-trained on a future sentence prediction objective. Specifically, given two segments $\langle s_a, s_b \rangle$, this pre-training objective aims to predict whether s_b will appear after s_a at some point in the same text. This objective is undoubtedly related to the POCC task and ALBERT should therefore be useful for my task. I also train the neural models in an MTL setup.[2] As an auxiliary task, the models also predict the relation label that connects the input pair. The MTL setup should help us analyse whether relation labels do matter for pairwise ordering. Figure 5.2 shows my architecture. Section 5.3.1 will show the performance of these models for the task.

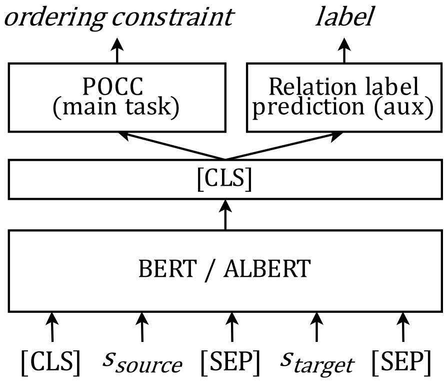


FIGURE 5.2: My architecture for fine-tuned POCC models.

Retain Original Pairwise Ordering

As a comparison to the neural methods, I also propose to retain the original pairwise ordering ("ROPO" strategy) between sentences in the POCC step. Based on my knowledge of the expert annotator's behaviour, in particular how rarely they chose to reorder (and only 1-3 out of 14 sentences were moved when reordering happens), I expect retaining the original pairwise ordering in the POCC step to result in a reasonably well-performing pipeline. Table 5.1 shows that there is a very high degree of similarity between the original and reordered versions of essays. For instance, 98.6% of argumentatively connected sentence pairs in the SUPPORT relations were not swapped in relative order even after the reordering annotation.

[2] I use the dynamic loss as described in Kendall et al. (2018).

	Support		Detail		Attack		Restatement	
Same PO	2385	(98.6%)	1309	(99.5%)	338	(98.5%)	247	(99.6%)
Different PO	33	(1.4%)	6	(0.5%)	5	(1.5%)	1	(0.4%)

Table 5.1: The number and percentage of cases where the pairwise ordering (PO) between sentences are kept or changed after annotation, in the ICNALE-AS2R train split (347 essays). Here, we operationalise RESTATEMENT as a directed relation type.

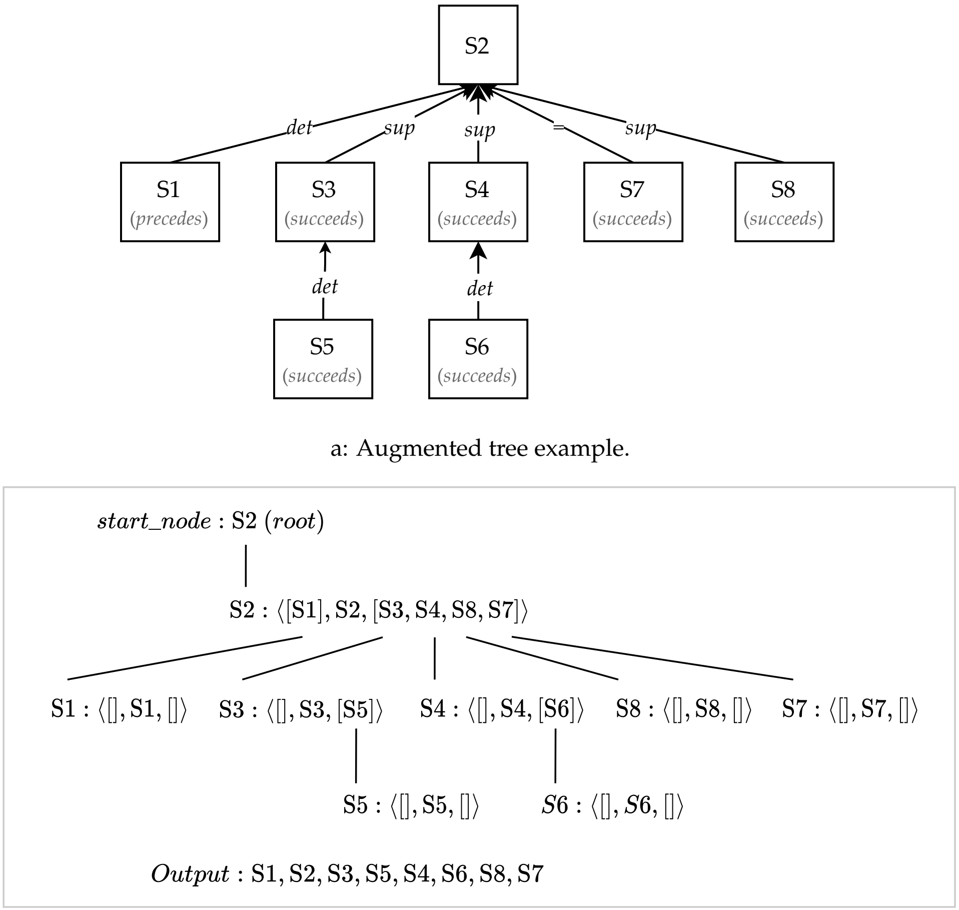
I will show the performance of the entire reordering pipeline when using neural and ROPO strategies in the POCC step in Section 5.3.2.

5.1.2 Traversal Algorithm

Given an argumentative structure that has been augmented with pairwise ordering information, the final step in the reordering module generates an output text by traversing the augmented tree. My traversal algorithm is inspired by prescriptive writing guidance for argumentation (Bacha, 2010; Silva, 1993), stating that a successful argumentative essay typically introduces the discussion topic and its major claim on the topic, discusses the topic in more depth, and then concludes the essay at the end. This tendency was also observed in the reordering annotation of the ICNALE-AS2R corpus by the expert annotator (cf. Section 3.4.4). Algorithm 1 describes how I formulate this strategy computationally.

Algorithm 1: Traversal Algorithm

```
function traversal($augmented\_tree$) begin
    queue_formation($augmented\_tree$)
    $start\_node \leftarrow root$ of $augmented\_tree$
    $output = []$
    recursive_util($start\_node, output$)
    return $output$
end
procedure queue_formation($augmented\_tree$) begin
    for $v$ in $augmented\_tree$ do
        $v.preceding\_queue \leftarrow$ sort($preceding\_children$ of $v$)
        $v.succeeding\_queue \leftarrow$ sort($succeeding\_children$ of $v$)
        if restatement(s) exist in $v.succeeding\_queue$ then
            move restatement(s) to the end of the queue
        end
    end
end
procedure recursive_util($v, output$) begin
    for $i \leftarrow 1$ to length($v.preceding\_queue$) do
        traversal_util($v.preceding\_queue[i], output$)
    end
    $output.append(v)$
    for $i \leftarrow 1$ to length($v.succeeding\_queue$) do
        traversal_util($v.suceeding\_queue[i], output$)
    end
end
```



a: Augmented tree example.

b: The illustration of the recursive_util function call for Figure 5.3a

FIGURE 5.3: An illustration of my traversal algorithm.

The traversal order of nodes in my algorithm depends on three aspects: (1) children-parent structure, (2) the pairwise ordering between them and (3) the order between siblings. The argumentative structure represents the first aspect, whereas the POCC step represents the second aspect. The traversal algorithm consists of two steps: queue formation (lines 8–16) and recursive (lines 17–25) steps. The queue formation step in my traversal algorithm processes the third aspect. It aims to create a data structure for each node v : $\langle preceding\_queue, v, succeeding\_queue \rangle$, which would be used in the following recursive step. The recursive step first traverses nodes in *preceding_queue* that should appear before v, then adding the node v itself into the output buffer, and finally traverses the nodes in *succeeding_queue*.

Given a node v in the queue formation step, I split its children into two sets of siblings: those that should precede (*preceding_children*) and succeed (*succeeding_children*) v in linear order based on the POCC results. For example, the *preceding_children* for node S2 in Figure 5.3a is {S1}, and its *succeeding_children* is {S3, S4, S7, S8}. I then sort these sibling sets based on their original order in the student essay (lines 10–11 in Algorithm 1). This process forms queues of nodes, *preceding_queue* and *succeeding_queue*, that should be visited during the recursive step. A special treatment is implemented for restatements: children nodes involved in RESTATEMENT relations are moved to the end of the queue (lines 12–14).[3] According to the prescriptive teaching of argumentation, there should not be any discussion following a concluding statement. This special treatment becomes necessary to prevent outputting the concluding statement in the middle of the reordered essay. For example, the *succeeding_queue* for node S2 is initially [S3, S4, S7, S8] at line 11 in Algorithm 1. Upon the special treatment for the restatement node S7 (lines 12–13), the *succeeding_queue* for node S2 becomes [S3, S4, S8, S7].

I then perform the recursive step starting from the root node (equivalent to the major claim in my scheme, lines 3–5). Nodes are added one by one to the output buffer. Figure 5.3a shows an augmented tree example, in which the pairwise ordering relation between child and parent nodes is determined using the ROPO strategy (cf. Section 5.1.1). Figure 5.3b illustrates the traversal process for this structure, starting from the root. There are eight times of calls to the recursive function here.

5.2 Sentence Ordering Models

Sentence reordering and sentence ordering tasks are different, as we have seen in the introduction of this chapter. The sentence reordering objective is to find a better sequence for sub-optimally ordered input sentences. The sentence ordering task, on the other hand, aims to find a coherent sequence for a given set of unordered sentences. I am interested to see how well sentence ordering models would perform in my task. Therefore, I am empirically evaluating them for general interest.

Here, I employ a maximum local coherence model that has been commonly used in natural language generation studies (cf. Section 2.4) and SotA topological sorting model (Prabhumoye et al., 2020). To make it possible for these models to run on my task, I provide randomised input sentences to the models. The models are trained to reconstruct the final version of essays in the ICNALE-AS2R corpus from scratch.

5.2.1 Maximum Local Coherence

The maximum local coherence model (MLCM) generates a sequence of sentences that maximises the local transition score between two adjacent sentences (Lapata, 2003). I adapt El Baff et al.'s (2019) approach and calculate the local transition score based on two aspects: (1) semantic similarity between sentences and (2) the transition of outgoing relation label.[4] The local transition score $P(s_i \mid s_j)$ from sentence s_j to s_i is calculated as in Equation (5.1), where sim denotes the semantic similarity between sentences and T denotes the probability of transitioning from outgoing relation-label l_j to l_i. The semantic similarity score is measured as the cosine similarity of SBERT sentence embeddings, and the relation-label transition bigram is calculated on ICNALE-AS2R train set.

$$P(s_i \mid s_j) = sim(s_i, s_j) \times T(l_i \mid l_j) \tag{5.1}$$

[3] Although restatements are logically equivalence classes and not directed relations, it is nevertheless sometimes convenient to represent them as directed links. This is so because I do not allow circular links in implementation. Recall that I did the same in Chapter 4

[4] El Baff et al. (2019) originally used the semantic similarity between sentences and the probability of rhetorical category transition for computing the local transition probability. However, ICNALE-AS2R corpus was not annotated with rhetorical categories, and I use relation labels instead.

Given an input text and its corresponding argumentative structure, I choose the root of the structure as the first sentence to be outputted by this method. The next sentence is then greedily chosen among other remaining pool of sentences based on the transition score to the first sentence. This is performed iteratively until there is no remaining sentence in the pool.

5.2.2 Topological Sorting

Prabhumoye et al. (2020) formulated the sentence ordering task as a constraint learning problem. Their model operates by first deciding the relative ordering between all pairs of sentences. An output is then generated by using the topological sorting algorithm (Tarjan, 1976).

Given a set of N sentences as input, there are $\binom{N}{2}$ possible sentence pairs. For example, if a text has four sentences $s_1, ..., s_4$, then there are six combinations of pairs: (s_1, s_2), (s_1, s_3), (s_1, s_4), (s_2, s_3), (s_2, s_4) and (s_3, s_4). For each pair, we randomly choose which sentence acts as the *source* and *target*.[5] A classifier then decides whether the *source* precedes or succeeds the target sentence in linear order. Here, I fine-tune BERT and ALBERT language models. This step is similar to the POCC step in my approach, except that Prabhumoye et al. used all combinations of sentences while I use only argumentatively connected pairs. Note that non-AC sentences are excluded in both approaches.

The POCC step for all pairs of sentences results in a topological graph, where a directed edge $u \rightarrow v$ from node u to v denotes that u should come before v in the output text. The output is then generated using the topological sorting algorithm, returning a sequence of nodes where each node appears before all the nodes it points to in the topological graph. For this algorithm to work, the topological graph has to be in the form of a directed acyclic graph (DAG). Otherwise, this approach produces no output. This means that the topological sorting approach has limited coverage and usability. We show that this is the case in Section 5.3.

5.3 Experimental Result and Discussion

The ICNALE-AS2R corpus is split into 80% train set (347 essays, 4,841 sentences) and 20% test set (87 essays, 1,180 sentences), stratified according to prompts, scores and country of origin of the EFL learners. This split is the same as I used in the argumentative structure parsing experiment (cf. Section 4). There are 81 reordered essays in the train set, and 24 in the test set. I am reporting two experiments here: the POCC and end-to-end reordering evaluations.

I evaluate POCC models for two types of input: (1) pairs of sentences connected by gold argumentative relations (“ArgPairs”, for my proposed approach) and (2) all pairs of sentences (“AllPairs”, for the topological sorting approach). Table 5.2 shows the number of train and test instances for the POCC task. I run the experiment for 20 times and report the average performance in Section 5.3.1. Statistical testing is conducted using the permutation test on the performance scores of the 20 runs with a significance level of $\alpha = .05$ whenever possible.

I also perform an evaluation for reordering systems. In this evaluation, I analyse which approach performs best in reconstructing the reordered version of essays in the ICNALE-AS2R corpus. Additionally, I also evaluate to which degree automatic

[5] This is to create balanced training data and randomised input. I follow the original implementation at https://github.com/shrimai/Topological-Sort-for-Sentence-Ordering.

Input Type	Train		Test	
	Precedes	Succeeds	Precedes	Succeeds
ArgPairs	856	3,468	184	857
AllPairs	13,422	17,472	3,023	4,025

TABLE 5.2: The number of training and test instances for the POCC task. Each column denotes the number of source sentences that precedes and succeeds their corresponding target sentences.

systems can recognise (and *not* reorder) the learners' essays that are already well-structured. I evaluate the entire reordering pipeline in an end-to-end fashion: given a student essay, my system automatically predicts the argumentative structure of the essay. This prediction will be used in the POCC and traversal steps that follow. I use the best model run for each step in the pipeline to perform this.[6] I also evaluate my traversal algorithm independently; given gold standard answers for argumentative parsing and POCC steps, an important research question is whether the traversal algorithm can properly reconstruct the final version of essays in the ICNALE-AS2R corpus. In this thesis, I hypothesised that the argumentative structure is important for sentence reordering. I also compare the reordering pipeline performance when fed with automatically predicted structures, gold structures and random structures to verify this hypothesis.[7] In the evaluation of reordering performance, statistical testing is conducted using the permutation test on the performance scores across test essays with a significance level of $\alpha = .05$ whenever possible.

5.3.1 Pairwise Ordering Constraint Classification: Results

Pairs of Sentences Connected by Argumentative Relations

Model	F1-precedes	F1-succeeds	F1-macro
BERT	.506	.903	.705
BERT [MTL]	.509	.901	.705
ALBERT	.559	.914	.737
ALBERT [MTL]	<u>.574</u>	<u>.916</u>	<u>.745</u>
ROPO	**.953**†	**.989**†	**.971**†

TABLE 5.3: Pairwise ordering classification results for pairs of sentences connected by argumentative relations. The best result is shown in **bold-face**. The † symbol indicates that the difference to the second-best result (underlined) is significant.

Table 5.3 shows the experimental results for the POCC task using pairs of sentences connected by argumentative relations. We can see that the ROPO strategy attains the best performance by a large margin. It achieves an F1-macro score of .971, whereas the performance of neural models is in the .7 range (significant difference).

[6] For the argumentative structure parsing step, I use the BIAF+QACT+ND [SS] model for sentence linking and the fine-tuned BERT model for relation labelling tasks, which were previously developed in Chapter 4.

[7] Random spanning trees are generated using the networkx (Python) library. Relation labels are randomly assigned to edges, following the observed distribution in the ICNALE-AS2R corpus.

Among the neural models, ALBERT (.737) performs slightly but significantly better than BERT (.705). Training ALBERT with MTL slightly improved the F1-score to .745, but the difference is not significant to the single-task model. Neural models are performing poorly in predicting source sentences that should precede their corresponding target sentences. This is unsurprising given there are fewer preceding cases than succeeding cases in the dataset.

In the end-to-end reordering evaluation (Section 5.3.2), I use both ROPO and neural-based models for the POCC step. My preference here is to use the ALBERT model (single-task) when using the neural-based model because the MTL model is not significantly better.

All Pairs of Sentences

We next take a look into the neural models' performance on the POCC task for all pairs of sentences. This is the POCC step for the topological sorting approach (cf. Section 5.2.2). Table 5.4 shows the result, where ALBERT significantly outperforms the BERT model in terms of F1 scores (.689 vs. .677). Since Prabhumoye et al.'s approach relies on the topological sorting algorithm, I also evaluate the ratio of the models' outputs that form DAG (denoted as the "DAG ratio"). Both models have a relatively low DAG ratio, 14.0% for BERT and 10.9% for ALBERT. This means that the topological sorting approach indeed has limited coverage and usability as I suspected–it will not always generate a final output for any given input.

Model	**F1-precedes**	**F1-succeeds**	**F1-macro**	**DAG ratio**
BERT	.713	.641	.677	**14.0%**†
ALBERT	**.732**†	**.647**†	**.689**†	10.9%

TABLE 5.4: Pairwise ordering classification results for all pairs of sentences. **Bold-face**, † and underlined as above.

In this experiment, I prioritise the performance in terms of F1-macro and take ALBERT as the best performing POCC model for the topological sorting sentence ordering approach.

5.3.2 End-to-end Reordering

I consider four evaluation metrics for the end-to-end reordering evaluation. The first is the *perfect match ratio* (PMR), calculating the ratio of essays for which the entire gold ordering sequence is correctly predicted (Chen et al., 2016). I am also interested in finding out about partial matches, as such metrics tell us how far the model output is from the gold standard. Partial metrics are useful to distinguish models' performance in the situation where the models cannot recreate the gold ordering perfectly. I evaluate partially correct predictions using the *Longest Common Subsequence Ratio* (LCSR) and the *Kendall Tau* (Tau). LCSR measures the longest common subsequent between the predicted order and the gold standard (Gong et al., 2016). The third metric, Tau, calculates the distance between the predicted order and the gold standard in terms of the number of pairwise inversions (Kendall, 1938).[8] It is computed as Tau $= 1 - \frac{2I}{N(N-1)/2}$, where I is the number of pairs in the predicted

[8] Lapata (2006) showed the empirical evidence that Tau correlates with human judgements for evaluating sentence ordering tasks.

order with incorrect relative order and N denotes the number of ACs in gold order.[9] Finally, I compute the *minimum edit distance* (MED) to quantify how many insertion, deletion or replacement operations are needed to transform the predicted order into the gold order. LCSR and Tau show the relative similarity between prediction and gold order, while MED tells us how many sentences have to be moved when the prediction is not the same as the gold standard. A higher score is better for PMR, Tau and LCSR, while a lower score is better for MED. The significance test is conducted only on LCSR and Tau. This is because PMR scores are typically so low that the likelihood to detect a significant difference is very low. On the other hand, MED scores are usually high so that the likelihood to detect a significant difference is very high.

Automatic Systems Run on Reordered Essays

Table 5.5 shows the evaluation results on the 24 reordered essays in the test split. For reference, I also include the performance scores when the input essay does not undergo any reordering ("leave untouched").

The best sentence reordering pipeline is achieved by using the ROPO strategy during the POCC step, attaining LCSR of .827 and Tau of .754. ALBERT pipeline follows in the second place, attaining LCSR of .787 and Tau of .739. The difference between ROPO and ALBERT pipelines is significant in terms of LCSR. The topological sorting approach only works on those cases where the graph resulting from the POCC step is DAG; in the current experiment, this happened in only 3 out of 24 cases. The scores for this approach are calculated only on those 3 cases. The topological sorting approach is therefore not comparable to other models, and I report the performance strictly for general interest. This limitation makes the topological sorting approach inherently impractical compared with the other approaches. The MLCM model shows the worst performance in the current experiment, attaining LCSR of .475 and Tau of .168 (significantly worse than other models in both metrics).

In general, the sentence reordering models perform better at reconstructing the gold standard than sentence ordering models. This shows that the presence of prior ordering information matters (recall that sentence ordering models are fed with randomised input).

Model	**PMR**	**LCSR**	**Tau**	**MED**
Sentence reordering				
ALBERT	.000	.787	.739	4.83
ROPO	**.125**	**.827**†	**.754**	**3.54**
Sentence ordering				
MLCM	.000	.475	.168	10.00
Topological Sorting	.000	.640	.323	7.33
Leave untouched	.000	.903	.873	2.58

TABLE 5.5: End-to-end paragraph reconstruction performance of automatic systems on the 24 reordered test essays. **Bold-face**, † and underlined as above. Statistical testing is conducted only for LCSR and Tau metrics.

[9] Non-ACs are not considered here because they are discarded from further processing. The difference between AC versus non-AC classification between gold and automatically predicted structures results in a penalty during the end-to-end reordering evaluation for all metrics.

It is better to always leave the input essay untouched rather than reordering it through the automatic systems. Thus, we can interpret that the automatic systems generate worse texts instead of better ones in the current stage. However, there is no gain without risk. We have shown evidence that some essays can be improved if reordered (cf. Section 3.4.4). Therefore, a system that does nothing is pointless. An essential question here is how we can improve the pipeline, so that it may produce better texts. I next perform an ablation study of my pipeline to identify possible improvements.

Ablation Study on Reordered Essays

Table 5.6 shows the reordering systems' performance when fed with automatically predicted argumentative structures, gold structures and random structures. I conduct the statistical test between all pairs of models and summarise the results here.[10] Let us first take a look at the upper bound performance. When feeding the system with gold argumentative structures and gold POCC answers (G-AS G-POCC), my traversal algorithm can reconstruct the reordered essays relatively well (LCSR of .961 and Tau of .952). Several essays are perfectly reconstructed too, given by the PMR score of .625. In non-perfect-reconstruction cases, the output is 1.08 (MED) sentences off from the gold standard, on average.

Model	**PMR**	**LCSR**	**Tau**	**MED**
G-AS G-POCC [*upper bound*]	**.625**	**.961**†	**.952**†	**1.08**
A-AS ALBERT	.000	.787	.739	4.83
G-AS ALBERT	.250	.825	.854	3.58
R-AS ALBERT	.000	.452	.203	10.71
A-AS ROPO	.125	.827	.754	3.54
G-AS ROPO	.208	.925	.897	1.92
R-AS ROPO	.000	.550	.394	9.58
Leave untouched	.000	.903	.873	2.58

TABLE 5.6: An ablation test of my proposed sentence reordering systems on the 24 reordered test essays. "G-AS" denotes gold argumentative structure, "A-AS" denotes automatically predicted argumentative structure and "R-AS" denotes random structure. **Bold-face**, † and underlined as above.

We next look at the performance of automatic systems. A-AS ALBERT achieves the LCSR of .787 and Tau of .739. When given gold argumentative structures, the ALBERT pipeline achieves LCSR of .825 and Tau of .854. A significant difference is observed between G-AS ALBERT and A-AS ALBERT in terms of Tau. On the other hand, the performance of the pipeline becomes very poor when using random structures, given by the LCSR of .452 and Tau of .203 (significant difference in both metrics to the A-AS model). The trend is the same for the ROPO pipeline. G-AS ROPO performs significantly better than A-AS ROPO in terms of LCSR and Tau. A-AS ROPO also performs significantly better than R-AS ROPO in both metrics. Another observation is that that the upper bound and G-AS ROPO systems produce texts that are closer to the gold standard than texts produced by the strategy of never reordering

[10] Detailed statistical testing results are provided in Appendix C.

(the upper bound is significantly better than leave untouched). This shows that it is worth taking the risk to reorder sentences rather than doing nothing at all.

These results prove that the quality of the argumentative structures is critical for the reordering pipeline. A further improvement in argumentative structure parsing is the most critical for improving the performance of the sentence reordering models. However, even if the outputs from the current parsing module are sub-optimal, automatically predicted structures still capture meaningful discourse relationships to some degree and can be useful for downstream tasks.

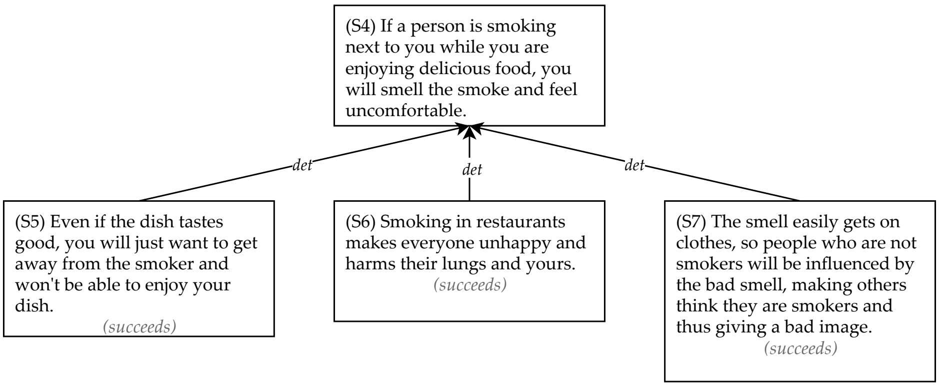

(S4) If a person is smoking next to you while you are enjoying delicious food, you will smell the smoke and feel uncomfortable.
det
det
det
(S5) Even if the dish tastes good, you will just want to get away from the smoker and won't be able to enjoy your dish.
(succeeds)
(S6) Smoking in restaurants makes everyone unhappy and harms their lungs and yours.
(succeeds)
(S7) The smell easily gets on clothes, so people who are not smokers will be influenced by the bad smell, making others think they are smokers and thus giving a bad image.
(succeeds)

FIGURE 5.4: Augmented tree snippet for essay "W_TWN_SMK0_003_B1_1_EDIT" (gold argumentative structure and gold POCC answers).

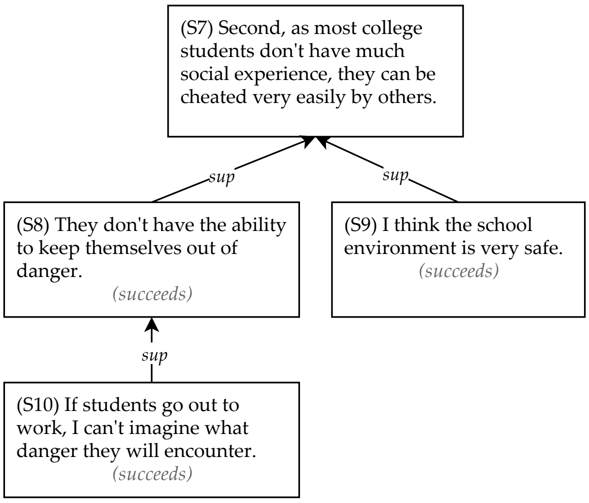

(S7) Second, as most college students don't have much social experience, they can be cheated very easily by others.
sup
sup
(S8) They don't have the ability to keep themselves out of danger.
(succeeds)
(S9) I think the school environment is very safe.
(succeeds)
sup
(S10) If students go out to work, I can't imagine what danger they will encounter.
(succeeds)

FIGURE 5.5: Augmented tree snippet for essay "W_CHN_PTJ0_041_A2_0_EDIT" (gold argumentative structure and gold POCC answers).

I next manually inspect the cases in which my traversal algorithm could not reconstruct the reordered essays perfectly, and found two directions in which the algorithm could be improved. The first direction is to consider the order between siblings. Recall that the traversal algorithm creates queues based on the original ordering between siblings. However, this does not always lead to optimal texts. Consider the augmented tree snippet in Figure 5.4. My traversal algorithm outputs

$[S4, S5, S6, S7]$ in this case. However, the final order produced by the expert annotator is $[S4, S5, S7, S6]$. One possibility is to try to arrange topically similar siblings close to each other to ensure a smooth transition of sentences. This could be achieved with some form of topic modelling.

The second direction is to consider the relation between nodes to their parents' siblings. This case is illustrated in Figure 5.5, where the gold order is $[S7, S8, S9, S10]$, whereas my algorithm outputs the sequence of $[S7, S8, S10, S9]$. To reproduce the gold order, it seems necessary to analyse the pairwise ordering relation between S9 and S10. However, since my approach only consider pairs of sentences connected by argumentative relations, the pairwise ordering between S9 and S10 cannot be expressed.

Automatic Systems Run on All Test Essays

Model	PMR	LCSR	Tau	MED
Sentence reordering				
ALBERT	.080	.810	.760	4.23
ROPO	**.448**	**.926**†	**.879**†	**1.79**
Sentence ordering				
MLCM	.000	.465	.132	10.28
Topological Sorting	.000	.676	.499	5.80
Leave untouched	.724	.973	.965	.71

TABLE 5.7: End-to-end paragraph reconstruction performance of automatic systems on the entire test set of 87 essays. **Bold-face**, † and underlined as above.

I next evaluate the automatic systems' performance on a larger test set of 87 essays, consisting of 24 reordered and 63 non-reordered final texts. Table 5.7 shows the gold standard reconstruction performance, where the main trends are still the same:

(a) Sentence reordering models can reconstruct the gold answers better than sentence ordering models.

(b) The topological sorting approach is inferior to other models in that it can only be applied in those 10 out of 87 instances (limitation from the DAG requirement), whereas other models can produce outputs all the time.

(c) If only considering the scores from automatic evaluation metrics, then it is better not to use any sentence reordering models at all, as they are outperformed by the simple strategy of never reordering.

Let us discuss point (c) in more detail. The four metrics used in Table 5.7 show the performance on gold standard reconstruction. However, they cannot show if the systems can perform the reordering task selectively, that is, reorder only when necessary and retain the original order when it is already optimal. This is an important ability for sentence reordering systems to minimise the odds of producing outputs of inferior quality compared with the inputs.[11] A good system should have a high accuracy in its judgement to reorder or retain the input.

[11] Sentence ordering systems do not have such ability because they assume randomised input. Randomised texts are always different from the gold answers, and thus, the models learn to always reorder the input texts.

		Prediction		Total
		Retain	Reorder	
Gold	Retain	**46**	17	63
	Reorder	2	**22**	24
	Total	48	39	87

TABLE 5.8: Confusion matrix of reordering operation for the upper bound system.

I first evaluate the upper bound performance on this aspect when using gold argumentative structures and gold POCC answers, given in Table 5.8. The upper bound system achieves an accuracy of .782.[12] Table 5.9 shows the confusion matrix for the reordering operation of the ALBERT pipeline where it tends to reorder too often. It attains an accuracy of .345 $\left(\frac{8+22}{87}\right)$, which is very far from the upper bound. The difference between both models is significant according to the two-tailed McNemar test (McNemar, 1947) with a significance level of $\alpha = .05$.

		Prediction		Total
		Retain	Reorder	
Gold	Retain	**8**	55	63
	Reorder	2	**22**	24
	Total	10	77	87

TABLE 5.9: Confusion matrix of reordering operation for the ALBERT pipeline.

		Prediction		Total
		Retain	Reorder	
Gold	Retain	**51**	12	63
	Reorder	11	**13**	24
	Total	61	25	87

TABLE 5.10: Confusion matrix of reordering operation for the ROPO pipeline.

Table 5.10 shows the confusion matrix for the ROPO pipeline, where it tends to retain the original order, compared with the ALBERT pipeline and the upper bound. It achieves an accuracy of .736 $\left(\frac{51+13}{87}\right)$. Although this score is significantly worse than the upper bound according to the McNemar test, it is still relatively good. When we look at it in an entire system context, we have to keep in mind that this step (decision to reorder or not) is only one aspect of the pipeline and that the actual end-to-end performance (cf. Table 5.8) will of course drop.

Ablation Study on All Test Essays

Table 5.11 shows the ablation study of the systems' performance when fed with automatically predicted structures, gold structures and random structures on all test

[12] The accuracy is measured by dividing the sum of the diagonal numbers by the total essay. For example, it is $\frac{46+22}{87} = .782$ for Table 5.8.

essays.[13] The trend is the same as in the previous ablation study on the reordered test essays: (1) the systems with gold structures perform significantly better than the corresponding systems using automatically predicted structures and (2) the systems with automatically predicted structures perform significantly better than those with random structures. This further reconfirms my hypothesis that the quality of the argumentative structure matters. Surprisingly, A-AS ROPO attains higher scores than G-AS ALBERT despite using automatic structures (significant difference on LCSR). This may have to do with the tendency of the pipeline to carry out the reordering operation when employing ALBERT in the POCC step, while the ROPO strategy enables the pipeline to do it more selectively. As I suspected, the ability to perform reordering selectively is important to minimise the risk of producing bad outputs.

Model	**PMR**	**LCSR**	**Tau**	**MED**
G-AS G-POCC [*upper bound*]	**.701**	**.962**	**.941**	**.99**
A-AS ALBERT	.080	.810	.760	4.23
G-AS ALBERT	.161	.850	.829	3.46
R-AS ALBERT	.000	.454	.210	10.49
A-AS ROPO	.448	.926	.879	1.79
G-AS ROPO	.586	.952	.925	1.22
R-AS ROPO	.000	.544	.413	9.64
Leave Untouched	.724	.973	.965	.71

TABLE 5.11: An ablation test of my proposed system on the entire test set of 87 essays. **Bold-face**, † and underlined as above.

5.4 Chapter Summary

In this chapter, I proposed a novel task of sentence reordering. The task aims to find a better sequence for possibly sub-optimally ordered input sentences and, otherwise, retain the original order if it is already optimal. My method to reorder sentences in EFL essays consists of three steps: (1) argumentative structure analysis, (2) pairwise ordering constraint classification (POCC) and (3) tree traversal. I employed the automatic argumentative structure parsing system previously developed in Chapter 4 to address the first step. In the second step, I employed a neural classifier; the strategy of always retaining the original pairwise ordering (ROPO) between sentences acts as a baseline. In the third step, I developed an original traversal algorithm that was inspired by prescriptive writing guidance for argumentation.

The ROPO pipeline attained the best performance in the experiment. It achieves .827 and .754 in LCSR and Tau scores on the reconstruction of the 24 reordered essays in the test set, and .926 and .879 in LCSR and Tau scores on the entire test set of 87 essays. Another crucial aspect is to evaluate whether automatic reordering systems are able to perform the reordering operation only when necessary. I verified that the proposed pipeline could perform it selectively, that is, it reorders input sentences when required and retains the original order when not required.

Here, I am also interested in the performance of sentence ordering models on my task, and therefore, I implemented several models. I found that the sentence

[13] Detailed statistical testing results are provided in Appendix C

ordering models perform poorly in my task. This suggests that the presence of prior order in the input plays an essential part in the reordering task.

An ablation study of the reordering system using automatically predicted argumentative structures, gold structures and random structures revealed that the quality of argumentative structure matters for the reordering task. The reordering system performance dropped when using automatic structures compared with using gold structures. The system also performed considerably worse when using random structures. Hence, a further improvement in the argumentative structure parsing module is foremost to enhance the reordering performance. It is also necessary to propose a better traversal algorithm, particularly considering the order between sibling nodes and between nodes to their parents' siblings.

Chapter 6

Conclusion

Writing coherent argumentative essays is a difficult task for EFL learners. A coherent argumentative text has to contain the desired argumentative elements; ideas should be clearly stated, connected to each other and supported by relevant reasons. Ideas also have to be organised in a way that is coherent in the eyes of native readers. If learners use the customs, reasoning patterns and rhetorical strategies of their first language when writing in the second language, there is a danger that the different organisation of ideas may violate the cultural expectations of native speakers; this makes learners' texts perceived as poorly organised (hence incoherent) in the eyes of native readers. This thesis described a study on analysing argumentative structure in EFL essays. It is motivated by the need and potential of argumentation analysis systems to enhance English-as-a-foreign-language learning by providing discourse level feedback. Such a system could develop learners' ability to evaluate the effectiveness of their writings, compare the structures of their writings with native writings and then subsequently improve their writings to closely resemble native-level productions. This thesis included three tasks: (i) constructing a new language resource for training automated systems, (ii) argumentative structure parsing, (iii) utilising argumentative structure for automatic sentence reordering.

In the first task, I presented ICNALE-AS2R, a corpus of 434 argumentative essays written by EFL students from various Asian countries. The corpus contains two layers of annotation: argumentative structure and sentence reordering. This corpus is unique among other argumentative text corpora in that it contains the argumentative structures of intermediate-quality texts and models text improvement. I employed four relation labels for the argumentative structure, namely SUPPORT, DETAIL, ATTACK and RESTATEMENT. I proposed to encode the semantics of RESTATEMENT as an equivalence class. The agreement study showed that my proposed argument annotation scheme is stable, with near-perfect intra-annotator agreement and reasonable inter-annotator agreement. Inter-annotator agreement results for the three argumentative-structure-annotation steps are as follows: Cohen's $\kappa = 0.66$ for argumentative component identification, Cohen's $\kappa = 0.53$ for linking argumentative components and Cohen's $\kappa = 0.61$ for four-way argumentative relation labelling. I also conducted a meta-evaluation to confirm whether the reordered version of essays is indeed better than the original ones by students. Three third-party professional essay assessors were employed, and they do not agree with each other in their judgement. The quality of sentence order is inherently subjective as there is no strict definition of the ideal sentence arrangement. The reordered version of the essays in the ICNALE-AS2R corpus, therefore, cannot be treated as the only correct version. But in adopting machine learning techniques, the reordered version was used as a gold standard to train computational models.

My qualitative analysis revealed that the argumentative structure annotation can indicate potential problems existing in the learners' texts. The reordering annotation then provides alternative sentence order configurations that lead to texts with a better logical flow. By manually inspecting the changes in reordered texts, I observe several reordering strategies that are commonly used in prescriptive classroom teaching. For instance, background information is moved to a place before the main argument and the conclusion is moved toward the end of the essay. Sentences logically belonging to the same sub-argument are gathered together. Within sub-arguments (sub-trees), the root or claim is often moved to the beginning, and further explanations, supporting evidence and examples typically follow after the root. Hence, beyond for training automated systems, the annotated essays in the ICNALE-AS2R corpus can also be used to support the practical teaching of how to argue and facilitate the study on contrastive rhetoric.

The annotated corpus comes with some additional methodological and technical contributions. First, I proposed a novel structure-based agreement metric called "mean agreement in recall" (MAR) that provides a more holistic approach to the evaluation of structural agreements. It comes in three variants, offering different insights depending on which unit of analysis is of interest (link, path or substructure). A large-scale meta-evaluation using 5,130 structure similarity judgements showed that the simplest variant, MAR$^{\text{link}}$, was on par with the structural metric proposed by Kirschner et al. (2015) in achieving a high correlation with human judgement. That being said, MAR$^{\text{link}}$ is more advantageous because of its unique mechanism in treating explicit and implicit links differently. Second, the annotation of the ICNALE-AS2R corpus is performed by an external expert annotator using my newly developed web-based annotation tool TIARA. The tool provides versatile visualisation to enhance structural annotation. Particularly, annotators can analyse the texts from both logical-sequencing and overall structure viewpoints using TIARA's dual-view display. TIARA also implements clutter-reducing features that are particularly helpful when annotating long texts. The tool is easily customisable via a configuration script. Besides has being used to annotate hundreds of texts in this thesis, TIARA has also been adopted for a secondary discourse annotation study by Matsumura and Sakamoto (2021). They studied how the visualisation of argumentative structures might help to detect argumentation-related problems in EFL writings and facilitate text assessment.

In the second topic, I extracted the argumentative structures of EFL essays using a pipelined neural approach, consisting of sentence linking and relation labelling steps. Experimental results showed that the biaffine attention model combined with the SBERT encoder performed the best in the sentence linking task at the F1-macro of .323. In the relation labelling task, the fine-tuned BERT model performed the best at the F1-macro of .595. The pipeline system achieved an end-to-end accuracy of .341. I also evaluated the parser in a cross-domain setting, where training is performed on both in-domain (original) and out-domain (reordered) data, and evaluation is performed on the in-domain test data. Experimental result shows that the cross-domain system achieved 94% (accuracy of .321) of the in-domain system. This signals the potential to use well-written texts together with less well-structured texts to increase the size of training data. I identified the sentence linking task as the main challenge of argumentative structure parsing; the system seems to stumble when confronted with the hierarchical nature of arguments. To improve the sentence linking performance, I extended the biaffine model using a novel set of structural-modelling related auxiliary tasks that require no additional annotation. Additionally, I also proposed a multi-corpora training strategy using the persuasive essay corpus (PEC) to increase training data. It has to be noted that simply increasing the training data does not guarantee improved performance. We need to ensure that the system indeed models our desired target distribution. To this end, I filtered PEC essays using the selective sampling technique. These two strategies provided a richer supervision signal to the sentence linking model and improved its performance on individual link predictions to the F1-macro of .374 from .323, amongst other improvements. The end-to-end parsing performance was also improved to the accuracy of .357 from .341.

In the third topic, I developed an automatic sentence reordering system as a downstream application of argumentative structure analysis. Given a text and its corresponding argumentative structure, I formulate the sentence ordering task as a tree traversal problem consisting of two steps. The first step is a pairwise ordering constraint classification (POCC) step where I identify the ordering constraint between argumentatively connected sentence pairs. The best performance for the POCC step is achieved by retaining the original pairwise ordering between sentences, achieving the F1-macro of .971. The second step is a tree traversal step where I generate output reordered texts by traversing the argumentative structure that has been augmented with pairwise ordering information. My best end-to-end reordering system achieved a Kendall's Tau of .879 on the entire test set. The system could also perform the reordering operation selectively, that is, it reorders sentences when necessary and retains the original input order when it is already optimal. An ablation study comparing the system's performance when using automatically predicted structures, gold structures and random structures revealed that the quality of argumentative structure matters for reordering. Hence, a further improvement in the argumentative structure parsing module is foremost for enhancing the reordering performance.

Future Directions

There are several possible directions for future work as follows.

- **Dataset**: In the current machine learning paradigm, it is common to have only a single correct answer for a given input. However, there might be multiple correct interpretations of argumentation and multiple acceptable reordering configurations for a given EFL essay. It is necessary to investigate whether it is possible to exhaustively annotate all possible better reordering variations for a given essay in the future. We also have to propose novel machine learning approaches that account for multiple gold standards.

- **TIARA**: Future versions of TIARA will improve the visualisation by easy comparisons of original and edited texts. In addition, I plan to allow relations between nodes and edges, for example, the *undercut* relation. On the purely technical side, the current version of TIARA is appropriate for relatively small-scale projects, whereas for bigger and more complex projects, an additional management feature would improve the annotation experience substantially. Therefore, I consider the provision of two parallel versions of TIARA: a light-weight client-side TIARA versus one with more extensive management, collaboration and monitoring features.

- **Meta-evaluation of MAR metric**: It was not possible to test all aspects of the MAR metric within the crowdsourcing paradigm. I am particularly curious whether the intuitions concerning the implicit links following from the equivalence class property of RESTATEMENT are borne out in practice. Another meta-evaluation could provide this assessment in the future but it would require judges with expertise in discourse analysis.

- **Argumentative structure parsing**: EFL essays are different from standard English texts because they are often noisy, for example, containing unnatural lexical choices and collocations. Existing sentence encoders are commonly trained on news or Wikipedia texts, which are arguably higher in quality than EFL texts. One direction to improve the parsing performance is by using a sentence encoder that is robust on noisy texts. Another direction is providing a richer supervision signal using various multi-task and transfer learning techniques. Recurrent neural networks are also particularly sensitive to order and can be expected to be thrown off by inconsistencies in learners' texts development strategies. Hence, it is better to process learners' texts using techniques that are invariant to input order, such as using the transformer architecture (Vaswani et al., 2017).

- **Sentence reordering evaluation**: A sentence reordering system has to rearrange sentences into a better-structured text. However, the model should retain the original order if the input sentences are already well-structured. This means that there is a risk of producing an output of inferior quality compared with the input when passing any text into the reordering system. However, there is no gain without risk, as we have discussed in Chapter 5. To properly consider such a risk into evaluation, we need a metric that encourages risk-taking behaviour. For example, giving higher rewards to successful reordering cases, and lower rewards on safer cases (do nothing and retains the original input order).

Appendix A

Annotation Guideline

Thank you for participating in our discourse annotation study. You will be given a set of argumentative essays on two different topics. The topic is given to the writers (students of the English language) in the form of a "prompt"; i.e., a sentence giving a statement to be discussed, for instance *"smoking should be banned at all restaurants in the country."* The students are told to produce a stand-alone text that can be read without knowing the prompt. We would like you to perform the following three tasks on these texts.

1. Annotating relations or dropping sentences
   For each sentence in the text, determine another sentence that is most closely related to it and indicate what their relationship is. Alternatively, remove the sentence if it does not contribute to the overall argument.

2. Reordering sentences
   If necessary, reorder the sentences to improve the overall logical flow. The reordered text should be a more well-structured argument than the original one.

3. Repairing text
   If it is necessary for understanding the reordered text and only then, you may change the referring and connective expressions.

The output of your work are two things.

- The structure of the text, expressed in form of relation links between sentences.
- The final text that results from you performing the above reorder and repair operations (the relations will be stripped away). Aim for the highest quality of text that can be produced with the above methods given to you.

For the automatic task that motivates this annotation, we care about both outputs equally.

A.1 Annotation Procedure

Roughly, an argumentative essay can be divided into three main parts: *introduction*, *body* and *conclusion*. Figure A.1 shows an illustration.

Introduction typically presents a general background about the discussion topic. It also contains the main claim that begins the argumentation. Since an argumentative essay aims to persuade the readers to adhere to the main claim, a deeper level of argumentation usually follows in the body. The body contains one or several ideas that *support* or *attack* the main claim. For example, one main reason argues why students should (or not) have a part-time job from the economics and education viewpoints. The essay's author may also describe a viewpoint on a deeper level of argumentation. For example, he/she might argue about economics from the viewpoint of practising financial management and lessening family's burden. Finally, the conclusion part sums up the entire argument, most often, by restating the main claim. Please note that the conclusion part does not strictly consist of only one sentence. It might be composed of several sentences.

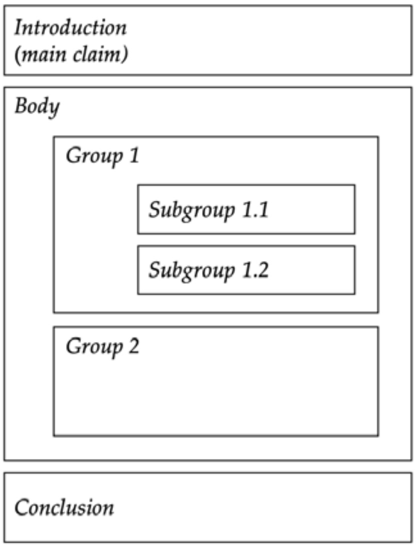


FIGURE A.1: General structure of an argumentative essay

The followings are **the sequential steps you should perform** when annotating an essay.

1. Read through the whole text at least once to understand its content.

2. Find the statement that expresses the author's opinion at the highest level of abstraction, i.e., the main claim.

3. Iteratively, determine the parts or *groups* existing in the text, i.e., *introduction*, *body* and *conclusion*. A part (especially body) might be recursively divided into several subgroups denoting deeper level of argumentation. Each group is represented by a representative sentence that you will choose. The group is connected to the rest of the argument only via this representative sentence. In a logical representation, relations in argumentative texts form hierarchical structures. You will annotate these relations by choosing from a set of four relation labels.

4. Determine relations existing in the text and drop sentences that are not connected to the argument (and ignore them from now on). We recommend that you start by determining the relations among sentences in the smaller groups.

5. If necessary, reorder sentences in such a way that a logically better-structured text results.

6. Reordering may cause changes in how people and things are referred to, and how sentences are connected. If necessary, edit the referring and connective expressions.

7. Read through the entire text, again, at least once to assess whether the current annotation is already the most proper annotation you can think of. If it is not, repeat the process from Step 2.

A.2 Annotating Relations or Dropping Sentences

There are two steps. First, find the text's main claim (as will be explained in Section A.2.1). Then, for each sentence X other than the main claim, determine another sentence Y that is strongly related to it. Express the relationship between X (which we call the *source*) and Y (which we call the *target*) as a link labelled with one of four possible relations.

There are four relation labels you can choose: *support*, *detail*, *attack* and *restatement*. The definitions of these relations are shown in Table A.1. The first three of them are directional: they hold between a *source* (lower in the hierarchy) and a *target* (higher in the hierarchy). The last relation, restatement, is non-directional, meaning that the *source* and *target* are not in a hierarchical relationship. The relations you can use are explained in more detail in Section A.2.2-A.2.5.

After the annotation process, the resulting relations in the text should form a hierarchical structure in which the main claim (conclusion, in the absence of the main claim) is at the top of the hierarchy. The main claim is then supported or attacked at a deeper level of argumentation, forming the hierarchical structure. Figure A.2 shows an example.

Label	Name	Description
`sup`	support	The *source* sentence asserts or justifies reasons and ideas for supporting the *target* sentence; it contains evidence or examples for the target sentence.
`det`	detail	The *source* sentence further explains, describes, elaborates or provides background for the concept(s) mentioned in the *target* sentence.
`att`	attack	The *source* sentence considers counter-arguments that argue for the opposite opinion.
`=`	restatement	The *source* sentence repeats high-level argument material that has been previously described. Note that the *source* sentence should not add a new idea into the discourse.

TABLE A.1: Relation labels

A.2.1 Finding Main Claim

The first step of the relation annotation is to find the statement that expresses the author's opinion at the highest level of abstraction, i.e., the main claim. It expresses the author's overall stance toward a discussion topic. After determining the main claim, you can proceed to identifying all remaining relations existing in the text. Consider the following example.

```
(Prompt) Smoking should be banned at all the restaurants in the country.
```

(1) Supported by the utilitarian perspective, I believe smoking should be completely banned at all restaurants.
(2) This is because there is overall harm if smoking is not prohibited.

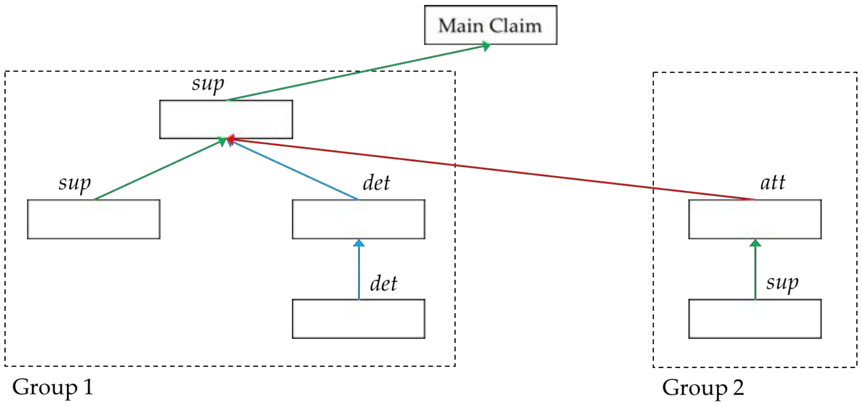


FIGURE A.2: Illustration of (logical) hierarchical structure

In this example, sentence (1) is the author's opinion at the highest level of abstraction denoting the author's stance in response to the discussion topic. Sentence (1) is the main claim, and other sentences will be connected according to their stance towards it.

Tip Sometimes, main claims might be marked with indications such as "*In my opinion,*" "*I strongly believe*" or "*I feel that.*" You can take such expressions into account, but your judgement should always be based on the context.

A.2.2 Support

A statement in an argumentative essay can be supported by several reasons/ideas. They assert why readers of the essay should believe the statement, in general, by providing *new* argumentative material. Ideally, readers are becoming more willing to accept the *target* sentence, provided reinforcement from the support sentences. Consider the following example.

(1) From my point of view, banning smoking in all restaurants is necessary.
(2) First, I think it is essential to protect the citizens' health.
(3) It is well known that smoking causes cancer.
(4) Second, banning smoking also allows all diners to eat in peace.

Sentence (1) is a statement about banning smoking in restaurants. It is further reinforced by two different ideas, and therefore, two different groups. The first group consisting of sentences (2) and (3) concerns health, and has sentence (2) as its representative sentence. Sentence (3) supports sentence (2) since it gives evidence for it. The second group consisting of sentence (4) concerns dining. Sentences (2) and (4) support the main claim, i.e., making the opinion in sentence (1) become more convincing. Therefore, both sentences (2) and (4) point at sentence (1) via the sup label, and sentence (3) points at sentence (2) via the sup label.

Tip The support relationships are sometimes indicated by, but not necessarily, the presence of list markers ("*first,*" "*second*"), exemplification expressions ("*for example*") or reasoning expressions ("*it is because,*" "*for this reason*").

A.2.3 Detail

The `det` relation label is applied in the following two cases.

- Sentences which present *additional* details (further explanations, descriptions or elaborations) about a particular sentence in question, but *without providing new argumentative material*.
- Sentences that introduce the topic of the discussion in a neutral way by providing general background, but *without any argumentative material*.

Consider the following example for the first case.

> (1) It is difficult to balance studying and working.
> (2) Especially if the students cannot manage their time well, because it only breaks the focus of their studying.

By reading sentence (1) on its own, we can infer that "*difficult to balance*" is talking about the time management between studying and working. When we read sentence (2) afterwards, we understand that sentence (2) elaborates the information which we inferred from sentence (1). Sentence (2) provides the author's explanation for the phrase "*difficult to balance*." Because no new argumentative material is introduced, and because sentence (2) provides additional explanation to sentence (1), you should annotate sentence (2) pointing at sentence (1) via the `det` label.

Consider the following example for the second case.

> (1) Today, more and more college students are taking part-time jobs.
> (2) I think having a part-time job is a good thing for them.

Sentence (1) is an introduction to a discussion topic in a neutral way. It enables the readers to comprehend sentence (2) by giving some contextual information. In this case, the correct annotation is to relate sentence (1) to sentence (2) via the `det` label; i.e., sentence (1) is now pointing *forward*[1] at sentence (2).

Tip Sometimes, `sup` and `det` labels might be in competition with each other. The main decision criterion is whether *new* argumentative material ("a new idea") is introduced or not. A new idea is a new reason for the target sentence, so `sup` is the correct relation. We can test whether this is the case by placing the word "*because*" between the target and the source, in this order. If the resulting sentence sounds odd, it is more likely to be a `det` relation, for instance an explanation or a definition.

A.2.4 Attack

The `att` relation label denotes sentences arguing for the opposite opinion. Consider the following example.

> (1) From my point of view, banning smoking in all restaurants is necessary.
> . . .
> (4) On the other hand, I admit that some restaurants are popular because men are allowed to smoke.

Sentence (1) is an example of the main claim in a smoking-themed essay. It states that smoking should be banned. However, sentence (4) argues against banning smoking

[1] Note that *backward* direction is more common in texts.

because smoking makes restaurants popular. In this example, the correct annotation is sentence (4) pointing at sentence (1) via the `att` label.

Tip The attack relations might, but do not have to, be indicated using the following expressions: *"on the other hand,"* *"but,"* *"however,"* *"in contrast,"* *"contrary to"* and *"in another way."*

A.2.5 Restatement

Two sentences are connected with "=" label if they are restatements of each other. Consider the following example.

> (1) I agree that college students should have a part-time job.
> . . .
> (4) Second, having a part-time job is a valuable way to pick up communication skills that will be needed in the workforce.
> . . .
> (8) Therefore, it is better for college students to have a part-time job to exercise communication skills.

Sentence (1) is an example of the author's opinion at the highest level of abstraction which is in favour of a part-time job. Sentence (4) states one of the reasons why students need a part-time job, i.e., to acquire communication skills. After further elaboration, the entire meaning of sentence (4) is restated as sentence (8). Notice that sentence (4) does not say anything about agreement or disagreement towards the question of part-time jobs while sentence (8) explicitly states it. However, we understand from reading sentence (1) that sentence (4) is implicitly in favour of a part-time job. This means that sentence (4) is basically restated as sentence (8). You should therefore connect sentence (4) and (8) with the "=" label. Restatements often happen in a situation such as the one above, where large parts of an argument are summarised for the second time. The two restatement sentences are treated as an equivalence class with respect to all outgoing and incoming relations they participate in.

Three properties must hold for a sentence to be a restatement.

1. There is a lot of repeated material from the high-level *target* sentence, and maybe, some other material from other sentences.
   A restatement reminds the readers of the strengths of its *target* sentence (an argument). Given that the readers have been provided reasons and explanations to the argument, restatement repeats the opinion and is central to *reinforce* the persuasion to accept the argument.

2. Restatement happens at a high-level argumentation.
   This means that the *source* sentence in a restatement relation fulfils an important function in the discourse. Ideally, the *target* sentence of a restatement relation is a major claim, group representative or subgroup representative (cf. Section A.1).

3. If there is non-repeated material in the *source* sentence, it should *not* add a new idea into the discourse.
   There should be no reasoning step between the *source* and *target* sentence of a restatement relation.

Figure A.3 shows a useful flowchart for judging a restatement.

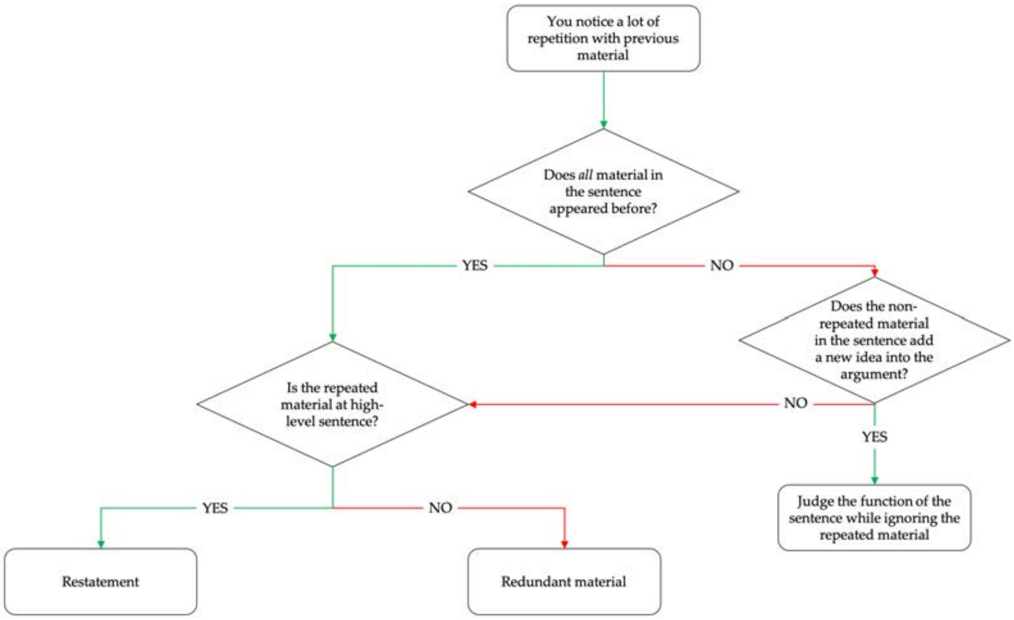

You notice a lot of repetition with previous material
Does *all* material in the sentence appeared before?
YES
NO
Does the non-repeated material in the sentence add a new idea into the argument?
Is the repeated material at high-level sentence?
NO
YES
Judge the function of the sentence while ignoring the repeated material
YES
NO
Restatement
Redundant material

FIGURE A.3: Restatement flowchart

Look at Figure A.4 to understand the difference between a restatement, redundant material and a *detail*. Sentence (3) in Figure A.4 (a) is a restatement. It repeats and reinforces a high-level argument. Furthermore, it does not add any new idea into the discourse. In contrast, sentence (4) in Figure A.4 (b) shows an example of the repeated information that does not fulfil any argumentative function in the discourse (can be safely dropped–Section A.2.8).

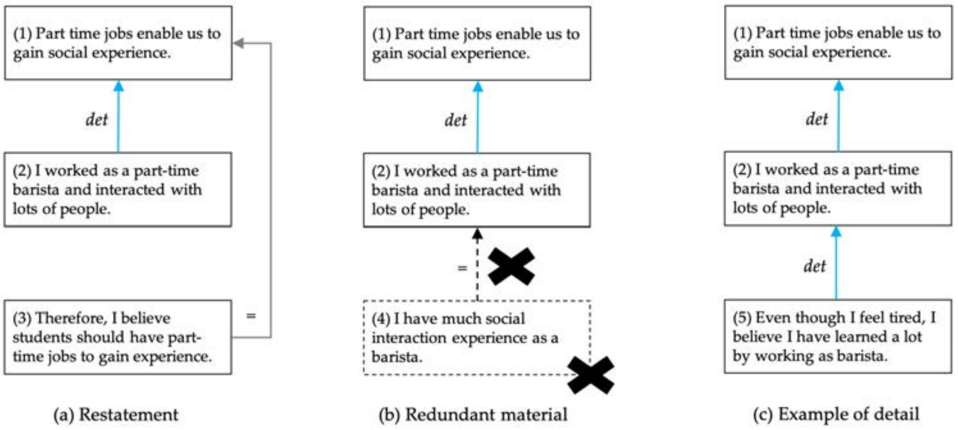

(1) Part time jobs enable us to gain social experience.
det
(2) I worked as a part-time barista and interacted with lots of people.
(3) Therefore, I believe students should have part-time jobs to gain experience.
=
(a) Restatement
(1) Part time jobs enable us to gain social experience.
det
(2) I worked as a part-time barista and interacted with lots of people.
=
(4) I have much social interaction experience as a barista.
(b) Redundant material
(1) Part time jobs enable us to gain social experience.
det
(2) I worked as a part-time barista and interacted with lots of people.
det
(5) Even though I feel tired, I believe I have learned a lot by working as barista.
(c) Example of detail

FIGURE A.4: The difference between restatement, non-relevant material and detail

You may get confused, in cases such as sentence (5) in Figure A.4 (c). Sentence (5) may look *like* a restatement as it states that the author (of the essay) learned a lot by working as a barista. However, the repeated part of sentence (5) does not fulfil the first criteria of restatement, i.e., it does not happen at a high-level argument, while the other part provides some additional description that working as a barista is *tiring*. Therefore, the proper analysis is that sentence (5) points at sentence (2) via the *det* relation.

Tip Restatement sentences might, but do not have to, be indicated by the following expressions: *"in conclusion," "to conclude," "therefore," "for all those reasons,"* and *"to sum up."*

A.2.6 Handling sequence and conjunctive arguments

Our scheme does not treat *sibling relations* in the hierarchy. But sometimes, you will come across strong sibling relations. For instance, sequences and conjunctive arguments. In the case of sequence, you should connect each component of the sequence to its preceding element via the `det` relation. The head acts as the representative for the sequence; therefore, you should connect the representative to the rest of the argument using the appropriate relation. Figure A.5 shows an example.

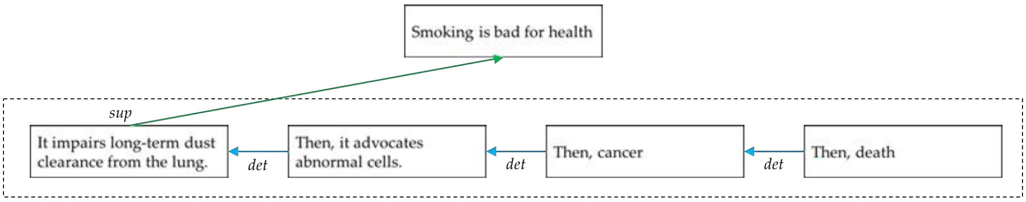

Smoking is bad for health
sup
It impairs long-term dust clearance from the lung.
det
Then, it advocates abnormal cells.
det
Then, cancer
det
Then, death

FIGURE A.5: Example of a sequence

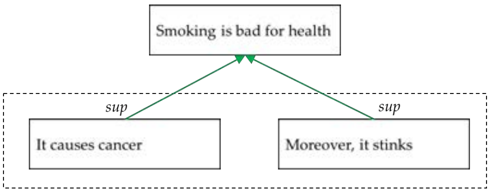


FIGURE A.6: Example of conjunctive arguments

In the case of conjunctive arguments, we assume that each argument is equally important, and that both of them support their target. We therefore annotate direct links between *each* member of the conjunctive argument and the target (see Figure A.6).

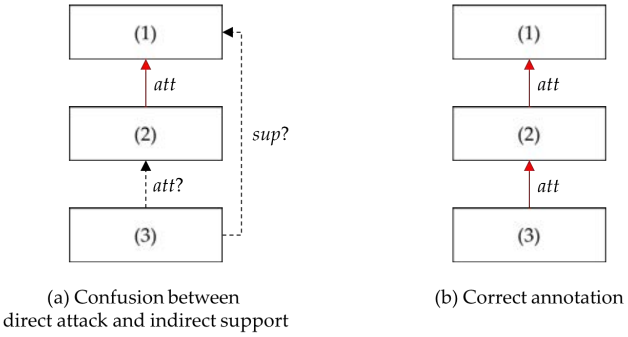
A.2.7 Relation Selection

A sentence can relate to many sentences at once. For example, a sentence might elaborate on two ideas at once. But in this task, you have to choose which sentence it relates to the most, and you cannot split sentences, either. There are several factors to consider.

1. Closeness in position.
   A sentence tends to relate more to those sentences that are close to it.

2. Directness of relation.
   A direct hierarchical relation is preferred over an indirect relation. For example, consider three sentences (1), (2) and (3). Sentence (3) attacks sentence (2), and sentence (2) attacks sentence (1). In this case, sentence (3) also indirectly supports sentence (1) by attacking sentence (2). However, since we prefer a direct relation, you should annotate the relation between sentence (3) and (2),

not the one between sentence (3) and (1). This situation is illustrated in Figure A.7.



(a) Confusion between direct attack and indirect support

(b) Correct annotation

FIGURE A.7: Direct vs indirect relation

3. Preferential ordering.
   You should choose the `sup` relation over `det` when a sentence both explains another sentence further and contains a *new* idea. This is because the new idea is more informative.

Sometimes, a relation can hold between the parts of a long sentence. But relationships inside a sentence cannot be expressed in our scheme, so do not worry about them. In this case, please annotate only the *function of the entire sentence as a whole*.

A.2.8 Dropping Criteria

You may find it hard to connect some sentences by using the four relations mentioned. In this case, you can also consider dropping them. We list the criteria to judge whether a sentence should be dropped.

1. Meta-information.
   You should drop sentences which only make statements about other sentences, without adding any real material. For example, *"I have two reasons for supporting this opinion."* Unlike details, this kind of sentences contributes nothing substantial toward the argument.

2. Redundant material.
   For example, a student may state twice that smoking is dangerous as it causes lung cancer. Please note the difference to restatements, which contain the same information, but typically at a higher level of argumentation (claims, not facts) and with a real function in the overall argument. Unlike redundant material, dropping a restatement might affect the flow of argumentation. So, the restatement cannot be dropped. The material considered as redundant here typically consists of mere facts, rather than real argumentative material such as claims or conclusions.

3. Truly disconnected sentences with no proper connection to the argument.
   Sometimes a sentence is logically isolated; i.e., it does not really relate to any other sentence. In this case, you should drop it.

Note that we want you to try to include as many sentences as possible. Do not drop an entire *group* (Section A.1) just because the function of the whole group is *anecdotal*. You should evaluate each individual sentence and only drop problematic sentences (as stated above) that do not contain any useable arguments. You should actively search for any useable parts of arguments. The remaining sentences after dropping should be logically connected to each other via relations. In the following, you can improve the text in order to reflect this.

A.3 Reordering Sentences

A good text usually places semantically related sentences close to each other, forming semantically consistent segments. If the sentences in the current text are not already in the best order they could be, arrange them into a logically well-structured argument; i.e., the best arrangement of sentences that you can think of. Remember that you can also drop sentences. Please make sure that you keep the original meaning of the text intact while doing so. Consider the following example.

> (1) If people smoke in the restaurant, other people may think the food isn't delicious.
> (2) At restaurants, people enjoy eating and talking.
> (3) They might have a sore throat and be unable to enjoy talking.

This text talks about the effect of smoking in restaurants, then talks about how dining should be an enjoyable experience while moving back again to the effect of smoking in restaurants. A better order is to place sentence (2) before sentence (1) as below.

> (2) At restaurants, people enjoy eating and talking.
> (1) If people smoke in the restaurant, other people may think the food isn't delicious.
> (3) They might have a sore throat and be unable to enjoy talking.

A.4 Repairing Text

After reordering the sentences in the previous step, you might have made the text itself harder to understand in certain superficial ways. In order to revert these negative changes, you are allowed to perform the following operations.

1. Change the text material used to connect two sentences or sentence parts.
   Examples of what we mean are "*however*," "*therefore*," "*but*" or "*eventhough*."

2. Change the text material used to identify people or things.
   Examples of what we mean are "*she*," "*the woman*," "*Maria*" or "*Sister of Kim*."

Please only make minimal repairs necessary for keeping the meaning the same. You should edit, i.e., add, delete, substitute, parts of the text (cf. material 1 and 2 above) only if it is needed for understanding the reordered text correctly.

Example 1

> (1) I think it is okay when poor students have part-time jobs.
> (2) Generally speaking, there are challenges in part-time jobs.
> (3) For instance, my girlfriend cannot focus on her studies.
> (4) I don't think she needs part-time jobs as she is not in a dire state for money.

After reordering the sentences in a more natural way, you might have the following text.

> (2) Generally speaking, there are challenges in part-time jobs.
>
> (3) For instance, my girlfriend cannot focus on her studies.
>
> (4) I don't think she needs part-time jobs as she is not in a dire state for money.
>
> (1) I think it is okay when poor students have part-time jobs.

After moving sentence (1) at the end of the text, it has a better flow. In this case, it might be useful to use the expression "*however*" at the beginning of sentence (1) as well. Thus, the final text looks as follows.

> (2) Generally speaking, there are challenges in part-time jobs.
>
> (3) For instance, my girlfriend cannot focus on her studies.
>
> (4) I don't think she needs part-time jobs as she is not in a dire state for money.
>
> (1) However, I think it is okay when poor students have part-time jobs.

Example 2

> (1) I don't like when my girlfriend is smoking.
>
> (2) She doesn't look cute while doing so.
>
> (3) But I think my grandmother doesn't care about looking cute.
>
> (4) It is okay if she smokes.

When you reorder the text, you might end up with the following order.

> (1) I don't like when my girlfriend is smoking.
>
> (2) She doesn't look cute while doing so.
>
> (4) It is okay if she smokes.
>
> (3) But I think my grandmother doesn't care about looking cute.

In this example, sentences (3) and (4) are swapped in position to make a better text; the sentences are now arranged in the form of opinions followed by reasons. But after reordering, the expression "*she*" in sentence (4) wrongly refers to "*my girlfriend*" instead of "*my grandmother*." To preserve the meaning of the statement, it is therefore necessary to replace "*she*" with "*my grandmother*" in sentence (4), as in the following text.

> (1) I don't like when my girlfriend is smoking.
>
> (2) She doesn't look cute while doing so.
>
> (4) It is okay if my grandmother smokes.
>
> (3) But I think my grandmother doesn't care about looking cute.

Note that in sentence (3), the repetition of "*my grandmother*" now sounds a bit unnatural while the meaning of the text is not affected. As we ask you to make only minimal changes, please leave "*my grandmother*" in sentence (3) as it is. However, the way of connecting sentences is unnatural in a different way too. To make the structure of "opinions followed by reasons" apparent, we can modify the text as follows.

> (1) I don't like when my girlfriend is smoking.
>
> (2) She doesn't look cute while doing so.
>
> (4) But it is okay if my grandmother smokes.
>
> (3) ~~But~~ I think my grandmother doesn't care about looking cute.

By introducing *"but"* at the beginning of sentence (4) and deleting *"but"* from sentence (3), the text now expresses the contrast relationship better.

Special Case: Repairing Main Claim

You may also have to fix the main claim if the author makes the error of assuming that the prompt is read alongside the text. It is because we consider that the prompt is not a part of the text. For example, he/she may write the following main claims.

1. I think *so*.
2. I *agree with the prompt*.
3. *But*, I do not think $\cdots$
4. $\cdots$ is bad *indeed*.

The example sentences above shows the case when the main claim appears in the beginning of the text. As the writers (students) are supposed to produce stand-alone texts, we should assume that readers do not read the prompt. You should repair the examples above by including some information from the prompt and/or editing phrases indicating discourse connection to the prompt. The final text should not refer to the prompt; it should not even mention the word "prompt." Some possible repairs for sentences above are as follows.

1. I think *smoking should be banned at all restaurants in the country*.
2. I *strongly believe smoking should be banned*.
3. I do not think $\cdots$
4. $\cdots$ is bad, *I think*.

Formatting

Editing should be done by placing the edited part inside a bracket "[*before* | *after*]." The *"before"* part denotes the expression before edit while the *"after"* part denotes the expression after edit. We will now give a formatting example of each operation.

1. **Addition**. Suppose you want to add an expression *"therefore,"* before the phrase *"the old man."* You rewrite this phrase as "[| *therefore,*] *the old man,"* leaving the *"before"* part as blank (space).
2. **Deletion**. Suppose you want to delete the word *"old"* from the phrase *"the old man."* You rewrite this phrase as *"the* [*old* |] *man,"* leaving the *"after"* part as blank (space).
3. **Substitution**. Suppose you want to substitute the word *"instead"* with *"but"* in the phrase *"I don't have a pen. Instead, I have a pencil."* You rewrite this phrase as *"I don't have a pen.* [*Instead* | *But*]*, I have a pencil."* You put the original phrase in the *"before"* part and the new phrase in the *"after"* part.

A.5 Annotation Illustration

To illustrate the whole annotation process, read the text below and follow the step-by-step illustration of its annotation with full attention. The text below is used through the rest of this section.

`(Prompt) Smoking should be banned at all restaurants in the country.`

(1) I agree with the previous statement.
(2) If somebody smokes in the restaurant, other people may not be able to enjoy their meal.
(3) At restaurants, customers enjoy eating and talking.
(4) However, if we ban smoking in restaurants, then those restaurants might lose some customers.
(5) Some restaurants are indeed popular, especially among old men, because they allow people to smoke.
(6) But, I firmly support banning smoking in restaurants because we need to prioritise health.
(7) In conclusion, I encourage banning smoking in all restaurants.

Step 1

Read through the whole text at least once to understand its content.

Step 2

The main claim is sentence (1).

Step 3

The text contains *introduction*, *body* and *conclusion* parts. The *introduction* consist of sentence (1), the *body* consists of sentence (2)–(6) and the *conclusion* consists of sentence (7). The *body* part can be divided into three groups. The first group, consisting of sentence (2)–(3), is about "*enjoyment of eating and talking*." The second group, composed of sentence (4)–(5), is about "*smoking and the number of customers*." Lastly, sentence (6), which forms the third group, argues from a "*health*" viewpoint. The grouping is illustrated in Figure A.8.

Step 4

We first consider relations existing in the smaller groups. In the "*enjoyment of eating and talking*" group, there are two sentences. Sentence (3) gives background for the phrase "*their meal*" in sentence (2). Therefore, sentence (3) points at sentence (2) via the `det` label. Sentence (2) acts as the representative of the group since it is the main statement of the group. As it supports sentence (1) by arguing for it, it receives a `sup` relation to sentence (1).

In the "*smoking and number of customers*" group, sentence (5) supports sentence (4) by presenting an opinion to increase readers' belief on it. Therefore, sentence (5) points at sentence (4) via the `sup` label. Sentence (4), the group representative, points at sentence (1) via the `att` label.

Sentence (6) presents an opposing opinion of sentence (4) by saying we should prioritise health. In this sense, sentence (6) supports sentence (1) by attacking sentence (4). However, as we prefer a more direct relation, sentence (6) is annotated as pointing at sentence (4) with the `att` label (cf. Section A.2.7).

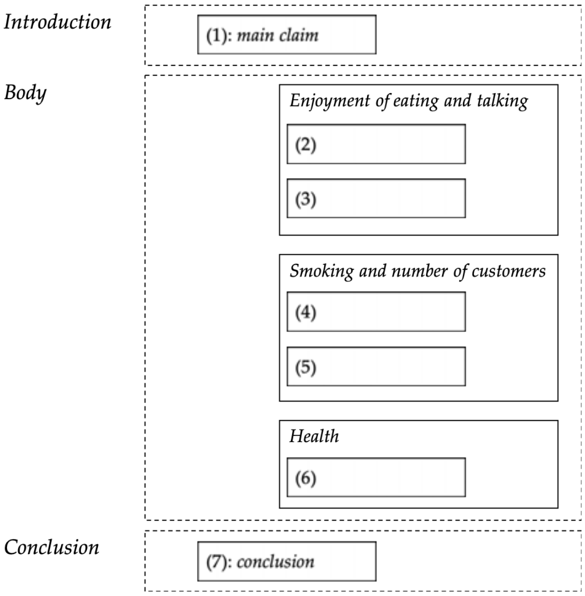


FIGURE A.8: Illustration of recognising *groups* in text

Finally, sentence (7) sums up the whole argument by basically restating the author's main claim. Even though sentence (7) is not the same as sentence (1), we understand that both expressions mean the same thing. Therefore, we annotate sentence (7) as a restatement of sentence (1) via the bidirectional "=" label. In this text, all sentences participate in the argument, and thus no sentence should be dropped. The relations we have established so far are illustrated in Figure A.9. As you can see, the relations form a hierarchical structure in which the main claim is placed at the top.

Step 5

We can improve the arrangement of sentences by swapping sentence (2) and (3). Sentence (2) talks about customers' meals, but sentence (3) gives basic information as background to sentence (2) and therefore, sounds more natural if it is stated first.

(1) I agree with the previous statement.
(3) At restaurants, customers enjoy eating and talking.
(2) If somebody smokes in the restaurant, other people may not be able to enjoy the experience.
(4) However, if we ban smoking in restaurants, then those restaurants might lose some customers.
(5) Some restaurants are indeed popular, especially among old men, because they allow people to smoke.
(6) But, I firmly support banning smoking in restaurants because we need to prioritise health.
(7) In conclusion, I encourage banning smoking in all restaurants.

Step 6

The author of the example text has made an error assuming that the prompt is read alongside the text (cf. Section A.4). It is indicated by the expression "*with the previous statement*" in sentence (1). Furthermore, it is necessary to improve the transition from sentence (4) and (5). For example, we can append "*This is because*" at the beginning of sentence (5). The result of this final step is given as follows.

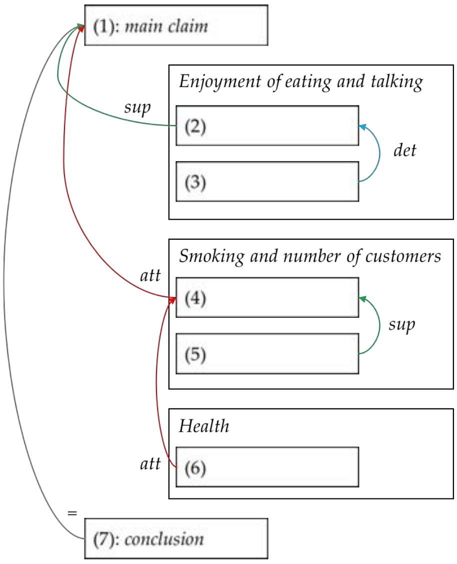


FIGURE A.9: Illustration of annotating relations

(1) I agree [with the previous statement | that smoking should be banned at all restaurants in the country].
(3) At restaurants, customers enjoy eating and talking.
(2) If somebody smokes in the restaurant, other people may not be able to enjoy their meal.
(4) However, if we ban smoking in restaurants, then those restaurants might lose some customers.
(5) [| This is because] some restaurants are indeed popular, especially among old men, because they allow people to smoke.
(6) But, I firmly support banning smoking in restaurants because we need to prioritise health.
(7) In conclusion, I encourage banning smoking in all restaurants.

Step 7

Read through the whole text, again, at least once to assess whether the current annotation is already the most proper annotation you can think and whether you can accept the text as it is. If this is the case, you can stop your annotation.

A.6 General Comment

We appreciate your work and patience. After you annotate the essays, we would like to hear about your experience with and observations about the texts, the tool and the task.

Appendix B

Implementation Notes

BERT encoder A sentence embedding was created by averaging subword embeddings composing the sentence in question. I used bert-base-multilingual-cased (`https://github.com/google-research/bert#pre-trained-models`) and bert-as-a-service (`https://github.com/hanxiao/bert-as-service`).

SBERT encoder I used SBERT encoder that was fine-tuned on the NLI dataset ("bert-base-nli-mean-tokens"), `https://github.com/UKPLab/sentence-transformers`.

Sequence Tagger (SeqTg) Dropout was applied between each layer, except between the encoder and the dimensionality reduction layer to prevent losing embedding information.

Biaffine (Biaf) Dropout was applied between all layers, following Dozat and Manning (2017).

Relation Labelling Models Dropout was applied between the final dense layer and the prediction layer for non-finetuning models.

Hyperparameter Tuning Before training all my models, I first performed a hyperparameter tuning step. To find the best hyperparameter set (e.g., batch size, dropout rate, training epoch) for each architecture, in combination with each encoder (BERT/SBERT) and each input type (in- or out-domain), I performed 5-fold-cross validation on the train set for five times. Then, I and selected the hyperparameter set that produced the best F1-macro score (in individual link predictions for the sentence linking models).

Hidden Units and Learning Rates The number of hidden units and learning rates to train my models are shown in Table B.1. All models were trained using Adam optimiser (Kingma and Ba, 2015), and implemented in PyTorch (Paszke et al., 2019) and AllenNLP (Gardner et al., 2018).

	Dense1	LSTM	Dense2	LR
Sentence linking				
SEQTG	512	256	256	.001
BIAF	512	256	256	.001
Relation labelling				
FFLSTM	256	128	256	.001
FFCON	256	-	256	.001
(DISTIL)BERT	-	-	-	$2e^{-5}$
Pairwise Ordering Constraint Classification				
BERT	-	-	-	$2e^{-5}$
ALBERT	-	-	-	$2e^{-5}$

TABLE B.1: The number of hidden units and learning rates (LR) of my models. “Dense1” denotes the dimensionality reduction layer (after encoder). “Dense2” denotes the dense layer after BiLSTM (before prediction).

Appendix C

Statistical Test Results

This appendix shows the detailed statistical test results for the ablation studies in Section 5.3. Table C.1 and C.2 show the p-values for pairs of models evaluated in the ablation study on the reordered test essays. Table C.3 and C.4 show the p-values for pairs of models evaluated in the ablation study on the entire test set.

		System Y							
		G-AS G-POCC	A-AS ALBERT	G-AS ALBERT	R-AS ALBERT	A-AS ROPO	G-AS ROPO	R-AS ROPO	Leave Untouched
System X	G-AS G-POCC	.507	1.000	.994	1.000	1.000	.967	1.000	.987
	A-AS ALBERT	.000	.504	.023	1.000	.407	.001	1.000	.006
	G-AS ALBERT	.007	.979	.496	1.000	.943	.173	1.000	.352
	R-AS ALBERT	.000	.000	.000	.499	.000	.000	.000	.000
	A-AS ROPO	.000	.587	.058	1.000	.499	.006	1.000	.023
	G-AS ROPO	.034	.999	.832	1.000	.993	.499	1.000	.723
	R-AS ROPO	.000	.000	.000	1.000	.000	.000	.504	.000
	Leave Untouched	.011	.993	.650	1.000	.974	.273	1.000	.497

TABLE C.1: P-values of one-tailed permutation test between all models in Table 5.6 for the Tau metric. This shows whether the mean score of system Y is higher than system X. Significant difference (p-value< 0.05) is marked in red font.

		System Y							
		G-AS G-POCC	A-AS ALBERT	G-AS ALBERT	R-AS ALBERT	A-AS ROPO	G-AS ROPO	R-AS ROPO	Leave Untouched
System X	G-AS G-POCC	.496	1.000	1.000	1.000	1.000	.983	1.000	1.000
	A-AS ALBERT	.000	.510	.074	1.000	.025	.000	1.000	.000
	G-AS ALBERT	.000	.923	.502	1.000	.362	.005	1.000	.027
	R-AS ALBERT	.000	.000	.000	.509	.000	.000	.001	.000
	A-AS ROPO	.000	.973	.641	1.000	.509	.010	1.000	.063
	G-AS ROPO	.013	1.000	.996	1.000	.990	.494	1.000	.939
	R-AS ROPO	.000	.000	.000	1.000	.000	.000	.498	.000
	Leave Untouched	.000	1.000	.975	1.000	.941	.067	1.000	.498

TABLE C.2: P-values of one-tailed permutation test between all models in Table 5.6 for the LCSR metric. This shows whether the mean score of system Y is higher than system X. Significant difference (p-value< 0.05) is marked in red font.

		System Y							
		G-AS G-POCC	A-AS ALBERT	G-AS ALBERT	R-AS ALBERT	A-AS ROPO	G-AS ROPO	R-AS ROPO	Leave Untouched
System X	G-AS G-POCC	.517	1.000	1.000	1.000	.991	.767	1.000	.089
	A-AS ALBERT	.000	.502	.018	1.000	.000	.000	1.000	.000
	G-AS ALBERT	.000	.982	.504	1.000	.050	.000	1.000	.000
	R-AS ALBERT	.000	.000	.000	.501	.000	.000	.000	.000
	A-AS ROPO	.009	1.000	.946	1.000	.501	.044	1.000	.000
	G-AS ROPO	.229	1.000	1.000	1.000	.959	.504	1.000	.015
	R-AS ROPO	.000	.000	.000	1.000	.000	.000	.507	.000
	Leave Untouched	.912	1.000	1.000	1.000	1.000	.985	1.000	.502

TABLE C.3: P-values of one-tailed permutation test between all models in Table 5.11 for the Tau metric. This shows whether the mean score of system Y is higher than system X. Significant difference (p-value< 0.05) is marked in red font.

		System Y							
		G-AS G-POCC	A-AS ALBERT	G-AS ALBERT	R-AS ALBERT	A-AS ROPO	G-AS ROPO	R-AS ROPO	Leave Untouched
System X	G-AS G-POCC	.490	1.000	1.000	1.000	.996	.791	1.000	.125
	A-AS ALBERT	.000	.509	.014	1.000	.000	.000	1.000	.000
	G-AS ALBERT	.000	.986	.497	1.000	.000	.000	1.000	.000
	R-AS ALBERT	.000	.000	.000	.505	.000	.000	.000	.000
	A-AS ROPO	.004	1.000	1.000	1.000	.503	.029	1.000	.000
	G-AS ROPO	.196	1.000	1.000	1.000	.970	.504	1.000	.016
	R-AS ROPO	.000	.000	.000	1.000	.000	.000	.508	.000
	Leave Untouched	.872	1.000	1.000	1.000	1.000	.988	1.000	.503

TABLE C.4: P-values of one-tailed permutation test between all models in Table 5.11 for the LCSR metric. This shows whether the mean score of system Y is higher than system X. Significant difference (p-value$<$ 0.05) is marked in red font.

Bibliography

Accuosto, Pablo and Horacio Saggion (Aug. 2019). 'Transferring Knowledge from Discourse to Arguments: A Case Study with Scientific Abstracts'. In: *Proceedings of the 6th Workshop on Argument Mining*. Florence, Italy: Association for Computational Linguistics, pp. 41–51. DOI: 10.18653/v1/W19-4505. URL: https://www.aclweb.org/anthology/W19-4505.

Al-Khatib, Khalid et al. (2016a). 'A News Editorial Corpus for Mining Argumentation Strategies'. In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan: The COLING 2016 Organizing Committee, pp. 3433–3443. URL: https://www.aclweb.org/anthology/C16-1324.

Al-Khatib, Khalid et al. (June 2016b). 'Cross-Domain Mining of Argumentative Text through Distant Supervision'. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, pp. 1395–1404. DOI: 10.18653/v1/N16-1165. URL: https://www.aclweb.org/anthology/N16-1165.

Al-Khatib, Khalid et al. (2017). 'Patterns of Argumentation Strategies across Topics'. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 1351–1357. DOI: 10.18653/v1/D17-1141. URL: https://www.aclweb.org/anthology/D17-1141.

Aristotle and George Kennedy (1991). *On Rhetoric: A Theory of Civil Discourse*. Oxford University Press. ISBN: 9780195064872.

Artstein, Ron and Massimo Poesio (2008). 'Survey Article: Inter-Coder Agreement for Computational Linguistics'. In: *Computational Linguistics* 34.4, pp. 555–596. DOI: 10.1162/coli.07-034-R2. URL: https://www.aclweb.org/anthology/J08-4004.

Ashley, Kevin D. (1990). *Modeling legal argument - reasoning with cases and hypotheticals*. Artificial intelligence and legal reasoning. MIT Press. ISBN: 978-0-262-01114-3.

Bacha, Nahla Nola (2010). 'Teaching the academic argument in a university EFL environment'. In: *Journal of English for Academic Purposes* 9.3, pp. 229 –241. ISSN: 1475-1585. DOI: https://doi.org/10.1016/j.jeap.2010.05.001. URL: http://www.sciencedirect.com/science/article/pii/S1475158510000378.

Bamberg, Betty (1983). 'What Makes a Text Coherent'. In: *College Composition and Communication* 34.4, pp. 417–429.

Barzilay, Regina, Noemie Elhadad, and Kathleen R. McKeown (2002). 'Inferring Strategies for Sentence Ordering in Multidocument News Summarization'. In: *Journal of Artificial Intelligence Research* 17.1, 35–55. ISSN: 1076-9757. URL: https://jair.org/index.php/jair/article/view/10306.

Barzilay, Regina and Mirella Lapata (2008). 'Modeling Local Coherence: An Entity-Based Approach'. In: *Computational Linguistics* 34.1, pp. 1–34. DOI: 10.1162/coli.2008.34.1.1. URL: https://www.aclweb.org/anthology/J08-1001.

Bird, Steven, Ewan Klein, and Edward Loper (2009). *Natural Language Processing with Python*. 1st. O'Reilly Media, Inc. ISBN: 0596516495.

Blair, J. Anthony (2012). *Groundwork in the Theory of Argumentation*. Springer.

Blanchard, Daniel et al. (2013). 'TOEFL11: A Corpus of Non-native English'. In: *ETS Research Report Series* 2.

Bobek, Eliza and Barbara Tversky (2016). 'Creating Visual Explanations Improve Learning'. In: *Cognitive Research: Principles and Implications* 1.1. DOI: 10.1186/s41235-016-0031-6.

Bollegala, Danushka, Naoaki Okazaki, and Mitsuru Ishizuka (2006). 'A Bottom-Up Approach to Sentence Ordering for Multi-Document Summarization'. In: *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*. Sydney, Australia: Association for Computational Linguistics, pp. 385–392. DOI: 10.3115/1220175.1220224. URL: https://www.aclweb.org/anthology/P06-1049.

Britt, M. Anne and Aaron A. Larson (2003). 'Constructing representations of arguments'. In: *Journal of Memory and Language* 48, pp. 794–810.

Bryant, Christopher and Ted Briscoe (2018). 'Language Model Based Grammatical Error Correction without Annotated Training Data'. In: *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 247–253. DOI: 10.18653/v1/W18-0529. URL: https://www.aclweb.org/anthology/W18-0529.

Cabrio, Elena and Serena Villata (2012). 'Combining Textual Entailment and Argumentation Theory for Supporting Online Debates Interactions'. In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Jeju Island, Korea: Association for Computational Linguistics, pp. 208–212. URL: https://www.aclweb.org/anthology/P12-2041.

— (2018). 'Five Years of Argument Mining: A Data-Driven Analysis'. In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. IJCAI'18. Stockholm, Sweden: AAAI Press, 5427–5433. ISBN: 9780999241127. URL: https://doi.org/10.24963/ijcai.2018/766.

Carletta, Jean (1996). 'Assessing Agreement on Classification Tasks: The Kappa Statistic'. In: *Computational Linguistics* 22.2, pp. 249–254. URL: https://www.aclweb.org/anthology/J96-2004/.

Carlile, Winston et al. (2018). 'Give Me More Feedback: Annotating Argument Persuasiveness and Related Attributes in Student Essays'. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, pp. 621–631. DOI: 10.18653/v1/P18-1058. URL: https://www.aclweb.org/anthology/P18-1058.

Carstens, Lucas and Francesca Toni (2015). 'Towards relation based Argumentation Mining'. In: *Proceedings of the 2nd Workshop on Argumentation Mining*. Denver, CO: Association for Computational Linguistics, pp. 29–34. DOI: 10.3115/v1/W15-0504. URL: https://www.aclweb.org/anthology/W15-0504.

Chen, Xinchi, Xipeng Qiu, and Xuanjing Huang (2016). 'Neural Sentence Ordering'. In: *CoRR* abs/1607.06952. arXiv: 1607.06952. URL: http://arxiv.org/abs/1607.06952.

Cho, Kwangsu and Charles MacArthur (2010). 'Student revision with peer and expert reviewing'. In: *Learning and Instruction* 20.4. Unravelling Peer Assessment, pp. 328 –338. ISSN: 0959-4752. DOI: 10.1016/j.learninstruc.2009.08.006.

Christensen, Janara et al. (2013). 'Towards Coherent Multi-Document Summarization'. In: *Proceedings of the 2013 Conference of the North American Chapter of the*

Association for Computational Linguistics: Human Language Technologies. Atlanta, Georgia: Association for Computational Linguistics, pp. 1163–1173. URL: https://www.aclweb.org/anthology/N13-1136.

Chu, Chenhui, Raj Dabre, and Sadao Kurohashi (July 2017). 'An Empirical Comparison of Domain Adaptation Methods for Neural Machine Translation'. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 385–391. DOI: 10.18653/v1/P17-2061. URL: https://www.aclweb.org/anthology/P17-2061.

Chu, Y. and T. Liu (1965). 'On the shortest arborescence of a directed graph'. In: *Science Sinica*, 1396–1400.

Cohen, Jacob (1960). 'A Coefficient of Agreement for Nominal Scales'. In: *Educational and Psychological Measurement* 20.1, pp. 37–46. DOI: 10.1177/001316446002000104.

Conneau, Alexis et al. (2017). 'Supervised Learning of Universal Sentence Representations from Natural Language Inference Data'. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 670–680. DOI: 10.18653/v1/D17-1070. URL: https://www.aclweb.org/anthology/D17-1070.

Connor, Ulla (2002). 'New Directions in Contrastive Rhetoric'. In: *TESOL Quarterly* 36.4, pp. 493–510. ISSN: 00398322. URL: http://www.jstor.org/stable/3588238.

Crossley, Scott A. and Danielle S. McNamara (2016). 'Say more and be more coherent: How text elaboration and cohesion increase writing quality'. In: *Written Research* 7.3, pp. 351–370. URL: https://eric.ed.gov/?id=ED565450.

Cui, Baiyun et al. (2018). 'Deep Attentive Sentence Ordering Network'. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, pp. 4340–4349. DOI: 10.18653/v1/D18-1465. URL: https://www.aclweb.org/anthology/D18-1465.

Cullen, Simon et al. (2018). 'Improving analytical reasoning and argument understanding: a quasi-experimental field study of argument visualization'. In: *NPJ Science of Learning* 3.21. DOI: 10.1038/s41539-018-0038-5.

Daxenberger, Johannes et al. (Sept. 2017). 'What is the Essence of a Claim? Cross-Domain Claim Identification'. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 2055–2066. DOI: 10.18653/v1/D17-1218. URL: https://www.aclweb.org/anthology/D17-1218.

De Kuthy, Kordula, Nils Reiter, and Arndt Riester (2018). 'QUD-Based Annotation of Discourse Structure and Information Structure: Tool and Evaluation'. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA). URL: https://www.aclweb.org/anthology/L18-1304.

Deguchi, Mamoru and Kazunori Yamaguchi (2019). 'Argument Component Classification by Relation Identification by Neural Network and TextRank'. In: *Proceedings of the 6th Workshop on Argument Mining*. Florence, Italy: Association for Computational Linguistics, pp. 83–91. DOI: 10.18653/v1/W19-4510. URL: https://www.aclweb.org/anthology/W19-4510.

Devlin, Jacob et al. (2019). 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding'. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

Technologies, Volume 1 (Long and Short Papers). Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://www.aclweb.org/anthology/N19-1423.

Dozat, Timothy and Christopher D. Manning (2017). 'Deep biaffine attention for neural dependency parsing'. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. URL: https://openreview.net/forum?id=Hk95PK9le.

Edmonds, Jack (1967). 'Optimum Branchings'. In: *Journal of Research of the National Bureau of Standards - B. Mathematics and Mathematical Physics* 71B, pp. 233–240. URL: http://dx.doi.org/10.6028/jres.071B.032.

Eger, Steffen, Johannes Daxenberger, and Iryna Gurevych (2017). 'Neural End-to-End Learning for Computational Argumentation Mining'. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 11–22. DOI: 10.18653/v1/P17-1002. URL: https://www.aclweb.org/anthology/P17-1002.

El Baff, Roxanne et al. (2019). 'Computational Argumentation Synthesis as a Language Modeling Task'. In: *Proceedings of the 12th International Conference on Natural Language Generation*. Tokyo, Japan: Association for Computational Linguistics, pp. 54–64. DOI: 10.18653/v1/W19-8607. URL: https://www.aclweb.org/anthology/W19-8607.

Ermakova, Liana, Josiane Mothe, and Anton Firsov (2017). 'A Metric for Sentence Ordering Assessment Based on Topic-Comment Structure'. In: *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '17. Association for Computing Machinery, 1061–1064. ISBN: 9781450350228. DOI: 10.1145/3077136.3080720.

Evans, Karla K. et al. (2011). 'Visual attention'. In: *WIREs Cognitive Science* 2.5, pp. 503–514. DOI: 10.1002/wcs.127.

Feng, Steven Y. et al. (2021). 'A Survey of Data Augmentation Approaches for NLP'. In: *CoRR* abs/2105.03075. arXiv: 2105.03075. URL: https://arxiv.org/abs/2105.03075.

Feng, Vanessa Wei, Ziheng Lin, and Graeme Hirst (2014). '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*. Dublin, Ireland: Dublin City University and Association for Computational Linguistics, pp. 940–949. URL: https://www.aclweb.org/anthology/C14-1089.

Fomicheva, Marina, Lucia Specia, and Francisco Guzmán (2020). 'Multi-Hypothesis Machine Translation Evaluation'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 1218–1232. DOI: 10.18653/v1/2020.acl-main.113. URL: https://aclanthology.org/2020.acl-main.113.

Fries, Peter H. (1994). 'On theme, rheme, and discourse goals'. In: *Advances in Written Text Analysis*. Ed. by Malcolm Coulthard. London: Routledge, pp. 229–249.

Fujiwara, Yumi (2018). 'The Role of Grammar Instruction in Japanese EFL context: Towards Communicative Language Teaching'. In: *Journal of the Academic Society for Quality of Life (JAS4QoL)* 4.4, pp. 1–11.

Gardner, Matt et al. (2018). 'AllenNLP: A Deep Semantic Natural Language Processing Platform'. In: *CoRR* abs/1803.07640. arXiv: 1803.07640. URL: http://arxiv.org/abs/1803.07640.

Garing, Alphie G. (2014). 'Coherence in Argumentative Essays of First Year College of Liberal Arts Students at De La Salle University'. In: *DLSU Research Congress*.

Gong, Jingjing et al. (2016). 'End-to-End Neural Sentence Ordering Using Pointer Network'. In: *CoRR* abs/1611.04953. arXiv: 1611.04953. URL: http://arxiv.org/abs/1611.04953.

Gontijo-Lopes, Raphael et al. (2021). 'Tradeoffs in Data Augmentation: An Empirical Study'. In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=ZcKPWuhG6wy.

Grosz, Barbara J. and Candace L. Sidner (1986). 'Attention, Intentions, and the Structure of Discourse'. In: *Computational Linguistics* 12.3, pp. 175–204. URL: https://www.aclweb.org/anthology/J86-3001.

Habernal, Ivan, Judith Eckle-Kohler, and Iryna Gurevych (2014). 'Argumentation Mining on the Web from Information Seeking Perspective'. In: *Proceedings of the Workshop on Frontiers and Connections between Argumentation Theory and Natural Language Processing*. URL: http://ceur-ws.org/Vol-1341/paper4.pdf.

Han, Na-Rae, Martin Chodorow, and Claudia Leacock (2006). 'Detecting errors in English article usage by non-native speakers'. In: *Natural Language Engineering* 12.2, 115–129. DOI: 10.1017/S1351324906004190.

Hearst, Marti A. (1997). 'Text Tiling: Segmenting Text into Multi-paragraph Subtopic Passages'. In: *Computational Linguistics* 23.1, pp. 33–64. URL: https://www.aclweb.org/anthology/J97-1003.pdf.

Hirst, Graeme and Alexander Budanitsky (2005). 'Correcting real-word spelling errors by restoring lexical cohesion'. In: *Natural Language Engineering* 11.1, 87–111. DOI: 10.1017/S1351324904003560.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). 'Long Short-Term Memory'. In: *Neural Computation* 9.8, pp. 1735–1780.

Hofmockel, Carolin, Anita Fetzer, and Robert M. Maier (2017). 'Discourse relations: Genre-specific Degrees of Overtness in Argumentative and Narrative Discourse'. In: *Argument & Computation* 8, pp. 131–151. DOI: 10.3233/AAC-170021.

Hovy, Eduard H. (1991). 'Approaches to the Planning of Coherent Text'. In: *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Ed. by Cécile L. Paris, William R. Swartout, and William C. Mann. Boston, MA: Springer US, pp. 83–102. ISBN: 978-1-4757-5945-7. DOI: 10.1007/978-1-4757-5945-7_3.

Hsin, Lisa B. and Catherine E. Snow (2020). 'Arguing for Teachers and for Friends: Eighth-graders' Sensitivity to Argumentation Features When Judging and Revising Persuasive Essays'. In: *Discourse Processes* 57.10, pp. 823–843. DOI: 10.1080/0163853X.2020.1803032.

Huang, Zhiheng, Wei Xu, and Kai Yu (2015). 'Bidirectional LSTM-CRF Models for Sequence Tagging'. In: *CoRR* abs/1508.01991. arXiv: 1508.01991.

Huddleston, Rodney and Geoffrey K. Pullum (2002). *The Cambridge Grammar of the English Language*. Cambridge University Press.

Iida, Ryu and Takenobu Tokunaga (2014). 'Building a Corpus of Manually Revised Texts from Discourse Perspective'. In: *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland: European Language Resources Association (ELRA), pp. 936–941. URL: http://www.lrec-conf.org/proceedings/lrec2014/pdf/155_Paper.pdf.

Invanic, Roz (2004). 'Discourses of Writing and Learning to Write'. In: *Language and Education* 18, pp. 220–245. DOI: 10.1080/09500780408666877.

Ishikawa, Shinichiro (2013). 'The ICNALE and Sophisticated Contrastive Interlanguage Analysis of Asian Learners of English'. In: *Learner Corpus Studies in Asia and the World* 1, pp. 91–118.

Ishikawa, Shinichiro (2018). 'The ICNALE Edited Essays: A dataset for analysis of L2 English Learner Essays Based on a New Integrative Viewpoint'. In: *English Corpus Linguistics* 25, pp. 1–14.

Jacobs, Holly L. et al. (1981). *Testing ESL composition: a practical approach*. Rowley, Massachusetts: Newbury House.

Janier, Mathilde, John Lawrence, and Chris Reed (2014). 'OVA+: an Argument Analysis Interface'. In: *International conference on computational models of argument*, pp. 463–464. DOI: 10.3233/978-1-61499-436-7-463.

Johns, Ann M. (1986). 'The ESL Student and The Revision Process: Some Insights from Scheme Theory'. In: *Journal of Basic Writing* 5.2, pp. 70 –80.

Kaplan, Dain, Ryu Iida, and Takenobu Tokunaga (2010). 'Annotation Process Management Revisited'. In: *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. Valletta, Malta: European Language Resources Association (ELRA). URL: http://www.lrec-conf.org/proceedings/lrec2010/pdf/129_Paper.pdf.

Kaplan, Robert B. (1966). 'Cultural Thought Patterns in Inter-cultural Education'. In: *Language Learning* 16.1-2, pp. 1–20. DOI: 10.1111/j.1467-1770.1966.tb00804.x.

Kendall, Alex, Yarin Gal, and Roberto Cipolla (2018). 'Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics'. In: *Proceedings of the International Conference on Learning Representations (ICLR)*. URL: https://arxiv.org/abs/1705.07115.

Kendall, Maurice George (1938). 'A New Measure of Rank Correlation'. In: *Biometrika*, pp. 81–93. DOI: 10.2307/2332226.

Kingma, Diederik P. and Jimmy Lei Ba (2015). 'Adam: a method for stochastic optimization'. In: *Proceedings of International Conference on Learning Representations (ICLR)*. URL: https://arxiv.org/abs/1412.6980.

Kiperwasser, Eliyahu and Yoav Goldberg (2016). 'Simple and Accurate Dependency Parsing Using Bidirectional LSTM Feature Representations'. In: *Transactions of the Association for Computational Linguistics* 4, pp. 313–327. DOI: 10.1162/tacl_a_00101. URL: https://www.aclweb.org/anthology/Q16-1023.

Kirschner, Christian, Judith Eckle-Kohler, and Iryna Gurevych (2015). 'Linking the Thoughts: Analysis of Argumentation Structures in Scientific Publications'. In: *Proceedings of the 2nd Workshop on Argumentation Mining*. Denver, CO: Association for Computational Linguistics, pp. 1–11. DOI: 10.3115/v1/W15-0501. URL: https://www.aclweb.org/anthology/W15-0501.

Koppel, Moshe, Jonathan Schler, and Kfir Zigdon (2005). 'Automatically Determining an Anonymous Author's Native Language'. In: *Intelligence and Security Informatics*. Ed. by Paul Kantor et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 209–217.

Kuribayashi, Tatsuki et al. (2019). 'An Empirical Study of Span Representations in Argumentation Structure Parsing'. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 4691–4698. DOI: 10.18653/v1/P19-1464. URL: https://www.aclweb.org/anthology/P19-1464.

Lan, Zhenzhong et al. (2020). 'ALBERT: A Lite BERT for Self-supervised Learning of Language Representations'. In: *Proceedings of the International Conference on Learning Representations*. URL: https://openreview.net/pdf?id=H1eA7AEtvS.

Lapata, Mirella (2003). 'Probabilistic Text Structuring: Experiments with Sentence Ordering'. In: *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. Sapporo, Japan: Association for Computational Linguistics, pp. 545–552. DOI: 10.3115/1075096.1075165. URL: https://www.aclweb.org/anthology/P03-1069.

— (2006). 'Automatic Evaluation of Information Ordering: Kendall's Tau'. In: *Computational Linguistics* 32.4, pp. 471–484. DOI: 10.1162/coli.2006.32.4.471. URL: https://www.aclweb.org/anthology/J06-4002.

Lauscher, Anne et al. (2018). 'Investigating the Role of Argumentation in the Rhetorical Analysis of Scientific Publications with Neural Multi-Task Learning Models'. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, pp. 3326–3338. DOI: 10.18653/v1/D18-1370. URL: https://www.aclweb.org/anthology/D18-1370.

Leffa, Vilson J. and Rua Felix Da Cunha (1998). 'Clause processing in complex sentences'. In: *Proceedings of the First International Conference on Language Resources and Evaluation (LREC)*, pp. 937–943.

Li, Jiwei and Dan Jurafsky (2017). 'Neural Net Models of Open-domain Discourse Coherence'. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 198–209. DOI: 10.18653/v1/D17-1019. URL: https://www.aclweb.org/anthology/D17-1019.

Lin, Ziheng, Hwee Tou Ng, and Min-Yen Kan (2011). 'Automatically Evaluating Text Coherence Using Discourse Relations'. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon, USA: Association for Computational Linguistics, pp. 997–1006. URL: https://www.aclweb.org/anthology/P11-1100.

Lippi, Marco and Paolo Torroni (2016). 'Argumentation Mining: State of the Art and Emerging Trends'. In: *ACM Trans. Internet Technol.* 16.2. ISSN: 1533-5399. DOI: 10.1145/2850417.

Little, Alexa and Stephen Tratz (2016). 'EasyTree: A Graphical Tool for Dependency Tree Annotation'. In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. Portorož, Slovenia: European Language Resources Association (ELRA), pp. 2343–2347. URL: https://www.aclweb.org/anthology/L16-1371.

Liu, Lu (2005). 'Rhetorical education through writing instruction across cultures: A comparative analysis of select online instructional materials on argumentative writing'. In: *Journal of Second Language Writing* 14.1, pp. 1–18. ISSN: 1060-3743.

Liu, Pei et al. (2020). 'A Survey of Text Data Augmentation'. In: *2020 International Conference on Computer Communication and Network Security (CCNS)*, pp. 191–195. DOI: 10.1109/CCNS50731.2020.00049.

Logeswaran, L., H. Lee, and Dragomir R. Radev (2018). 'Sentence Ordering and Coherence Modeling using Recurrent Neural Networks'. In: *AAAI*. URL: https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/download/17011/16079.

Louis, Annie and Ani Nenkova (2012). 'A Coherence Model Based on Syntactic Patterns'. In: *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Jeju Island, Korea: Association for Computational Linguistics, pp. 1157–1168. URL: https://www.aclweb.org/anthology/D12-1106.

Mann, William C. and Sandra A. Thompson (1988). 'Rhetorical Structure Theory: Toward a Functional Theory of Text Organization'. In: *Text* 8.3, pp. 243–281.

Matsumura, K. and K. Sakamoto (2021). 'A Structure Analysis of Japanese EFL Students' Argumentative Paragraph Writing with a Tool for Annotation Discourse Relation'. In: *The Bulletin of the Writing Reserch Group, JACET Kansai Chapter* 14.

McNemar, Quinn (1947). 'Note on the sampling error of the difference between correlated proportions or percentages'. In: *Psychometrika* 12.2, pp. 153–157.

Miwa, Makoto and Mohit Bansal (2016). 'End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures'. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 1105–1116. DOI: 10.18653/v1/P16-1105. URL: https://www.aclweb.org/anthology/P16-1105.

Mochales, Raquel and Marie Francine Moens (2001). 'Argumentation Mining'. In: *Artificial Intelligence and Law* 19.1, pp. 1–22.

Morio, Gaku et al. (2020). 'Towards Better Non-Tree Argument Mining: Proposition-Level Biaffine Parsing with Task-Specific Parameterization'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 3259–3266. DOI: 10.18653/v1/2020.acl-main.298. URL: https://www.aclweb.org/anthology/2020.acl-main.298.

Nguyen, Huy and Diane Litman (2016). 'Context-aware Argumentative Relation Mining'. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 1127–1137. DOI: 10.18653/v1/P16-1107. URL: https://www.aclweb.org/anthology/P16-1107.

Noreen, Eric W. (1989). *Computer Intensive Methods for Testing Hypotheses: An Introduction*. New York: Wiley.

O'Donnell, Michael (June 2000). 'RSTTool 2.4 - A markup Tool for Rhetorical Structure Theory'. In: *INLG'2000 Proceedings of the First International Conference on Natural Language Generation*. Mitzpe Ramon, Israel: Association for Computational Linguistics, pp. 253–256. DOI: 10.3115/1118253.1118290.

Okazaki, Naoaki, Yutaka Matsuo, and Mitsuru Ishizuka (2004). 'Improving Chronological Sentence Ordering by Precedence Relation'. In: *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*. Geneva, Switzerland: COLING, pp. 750–756. URL: https://www.aclweb.org/anthology/C04-1108.

Park, Joonsuk and Claire Cardie (2018). 'A Corpus of eRulemaking User Comments for Measuring Evaluability of Arguments'. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA). URL: https://www.aclweb.org/anthology/L18-1257.

Paszke, Adam et al. (2019). 'PyTorch: An Imperative Style, High-Performance Deep Learning Library'. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Peldszus, Andreas and Manfred Stede (2015). 'Joint prediction in MST-style discourse parsing for argumentation mining'. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 938–948. DOI: 10.18653/v1/D15-1110. URL: https://www.aclweb.org/anthology/D15-1110.

— (2016). 'An annotated corpus of argumentative microtexts'. In: *Argumentation and Reasoned Action - Proceedings of the 1st European Conference on Argumentation, Lisbon, 2015*.

Persing, Isaac, Alan Davis, and Vincent Ng (2010). 'Modeling Organization in Student Essays'. In: *Proceedings of the 2010 Conference on Empirical Methods in Natural*

Language Processing. Cambridge, MA: Association for Computational Linguistics, pp. 229–239. URL: https://www.aclweb.org/anthology/D10-1023/.

Persing, Isaac and Vincent Ng (2015). 'Modeling Argument Strength in Student Essays'. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Beijing, China: Association for Computational Linguistics, pp. 543–552. DOI: 10.3115/v1/P15-1053. URL: https://www.aclweb.org/anthology/P15-1053.

Potash, Peter, Alexey Romanov, and Anna Rumshisky (2017). 'Here's My Point: Joint Pointer Architecture for Argument Mining'. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 1364–1373. DOI: 10.18653/v1/D17-1143. URL: https://www.aclweb.org/anthology/D17-1143.

Prabhumoye, Shrimai, Ruslan Salakhutdinov, and Alan W Black (2020). 'Topological Sort for Sentence Ordering'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 2783–2792. DOI: 10.18653/v1/2020.acl-main.248. URL: https://www.aclweb.org/anthology/2020.acl-main.248.

Prasad, Rashmi et al. (2008). 'The Penn Discourse TreeBank 2.0.' In: *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*. Marrakech, Morocco: European Language Resources Association (ELRA). URL: http://www.lrec-conf.org/proceedings/lrec2008/pdf/754_paper.pdf.

Putra, Jan Wira Gotama, Simone Teufel, and Takenobu Tokunaga (2019). 'An Argument Annotation Scheme for the Repair of Student Essays by Sentence Reordering'. In: *Proceedings of Annual Meeting of the Association for Natural Language Processing Japan*, pp. 546–549. URL: https://www.anlp.jp/proceedings/annual_meeting/2019/pdf_dir/P3-9.pdf.

Putra, Jan Wira Gotama and Takenobu Tokunaga (2017). 'Evaluating text coherence based on semantic similarity graph'. In: *Proceedings of TextGraphs-11: the Workshop on Graph-based Methods for Natural Language Processing*. Vancouver, Canada: Association for Computational Linguistics, pp. 76–85. DOI: 10.18653/v1/W17-2410. URL: https://www.aclweb.org/anthology/W17-2410.

Rabinovich, Ella et al. (2016). 'On the Similarities Between Native, Non-native and Translated Texts'. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 1870–1881. DOI: 10.18653/v1/P16-1176. URL: https://www.aclweb.org/anthology/P16-1176.

Reed, Chris and Simon Wells (2007). 'Dialogical argument as an interface to complex debates'. In: *IEEE Intelligent Systems* 22.6, pp. 60–65. DOI: 10.1109/MIS.2007.106.

Reed, Chris et al. (2008). 'Language Resources for Studying Argument'. In: *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*. URL: https://www.aclweb.org/anthology/L08-1553/.

Reimers, Nils and Iryna Gurevych (2017). 'Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging'. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 338–348. DOI: 10.18653/v1/D17-1035. URL: https://www.aclweb.org/anthology/D17-1035.

— (2019). 'Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks'. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language*

Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3982–3992. DOI: 10.18653/v1/D19-1410. URL: https://www.aclweb.org/anthology/D19-1410.

Rush, Alexander M., Sumit Chopra, and Jason Weston (2015). 'A Neural Attention Model for Abstractive Sentence Summarization'. In: *Proceedings of the EMNLP*. Lisbon, Portugal: Association for Computational Linguistics, pp. 379–389. URL: http://aclweb.org/anthology/D15-1044.

Sanh, Victor et al. (2019). 'DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter'. In: *Proceedings of 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing, NeurIPS*. URL: https://arxiv.org/abs/1910.01108.

Schulz, Claudia et al. (June 2018). 'Multi-Task Learning for Argumentation Mining in Low-Resource Settings'. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 35–41. DOI: 10.18653/v1/N18-2006. URL: https://www.aclweb.org/anthology/N18-2006.

Sennrich, Rico, Barry Haddow, and Alexandra Birch (Aug. 2016). 'Improving Neural Machine Translation Models with Monolingual Data'. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 86–96. DOI: 10.18653/v1/P16-1009. URL: https://www.aclweb.org/anthology/P16-1009.

Shermis, Mark, Jill Burstein, and Claudia Leacock (2006). 'Applications of Computers in Assessment and Analysis of Writing'. In: *Handbook of writing research*. Ed. by Jill Filzgerald Charles A. MacArthur Steve Graham. New York: Guilford Publications.

Silva, Tony (1993). 'Toward an Understanding of the Distinct Nature of L2 Writing: The ESL Research and Its Implications'. In: *TESOL Quarterly* 27.4, pp. 657–677. URL: http://www.jstor.org/stable/3587400.

Skeppstedt, Maria, Andreas Peldszus, and Manfred Stede (2018). 'More or less controlled elicitation of argumentative text: Enlarging a microtext corpus via crowdsourcing'. In: *Proceedings of the 5th Workshop on Argument Mining*. Brussels, Belgium: Association for Computational Linguistics, pp. 155–163. DOI: 10.18653/v1/W18-5218. URL: https://www.aclweb.org/anthology/W18-5218.

Song, Wei et al. (2020). 'Discourse Self-Attention for Discourse Element Identification in Argumentative Student Essays'. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 2820–2830. DOI: 10.18653/v1/2020.emnlp-main.225. URL: https://www.aclweb.org/anthology/2020.emnlp-main.225.

Sonntag, Jonathan and Manfred Stede (2014). 'GraPAT: a Tool for Graph Annotations'. In: *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland: European Language Resources Association (ELRA), pp. 4147–4151. URL: http://www.lrec-conf.org/proceedings/lrec2014/pdf/824_Paper.pdf.

Stab, Christian and Iryna Gurevych (2014). 'Annotating Argument Components and Relations in Persuasive Essays'. In: *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. Dublin, Ireland: Dublin City University and Association for Computational Linguistics, pp. 1501–1510. URL: https://www.aclweb.org/anthology/C14-1142.

— (2017). 'Parsing Argumentation Structures in Persuasive Essays'. In: *Computational Linguistics* 43.3, pp. 619–659. DOI: 10.1162/COLI_a_00295. URL: https://www.aclweb.org/anthology/J17-3005.

Stenetorp, Pontus et al. (2012). 'brat: a Web-based Tool for NLP-Assisted Text Annotation'. In: *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*. Avignon, France: Association for Computational Linguistics, pp. 102–107. URL: https://www.aclweb.org/anthology/E12-2021.

Strobl, Carola et al. (2019). 'Digital support for academic writing: A review of technologies and pedagogies'. In: *Computers & Education* 131, pp. 33–48. ISSN: 0360-1315. DOI: https://doi.org/10.1016/j.compedu.2018.12.005. URL: https://www.sciencedirect.com/science/article/pii/S036013151830318X.

Suleiman, Mahmoud F. (2000). 'The Process and Product of Writing: Implications for Elementary School Teachers'. In: ERIC.

Sun, Yuechuan, Sijing Wu, and Ian Spence (2015). 'The Commingled Division of Visual Attention'. In: *PLOS ONE* 10.6, pp. 1–18. DOI: 10.1371/journal.pone.0130611.

Tarjan, Robert Endre (1976). 'Edge-disjoint spanning trees and depth-first search'. English (US). In: *Acta Informatica* 6.2, pp. 171–185. ISSN: 0001-5903. DOI: 10.1007/BF00268499.

Teufel, Simone and Marc Moens (2002). 'Articles Summarizing Scientific Articles: Experiments with Relevance and Rhetorical Status'. In: *Computational Linguistics* 28.4, pp. 409–445. DOI: 10.1162/089120102762671936. URL: https://www.aclweb.org/anthology/J02-4002.

Todd, Richard Watson, Patteera Thienpermpool, and Sonthida Keyuravong (2004). 'Measuring the coherence of writing using topic-based analysis'. In: *Assessing Writing* 9.2, pp. 85 –104. ISSN: 1075-2935.

Toledo-Ronen, Orith et al. (Nov. 2020). 'Multilingual Argument Mining: Datasets and Analysis'. In: *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics, pp. 303–317. DOI: 10.18653/v1/2020.findings-emnlp.29. URL: https://www.aclweb.org/anthology/2020.findings-emnlp.29.

Vaswani, Ashish et al. (2017). 'Attention Is All You Need'. In: *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS)*. Long Beach, CA, USA. URL: http://arxiv.org/abs/1706.03762.

Vinyals, Oriol, Meire Fortunato, and Navdeep Jaitly (2015). 'Pointer Networks'. In: *Advances in Neural Information Processing Systems*. Ed. by C. Cortes et al. Vol. 28. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2015/file/29921001f2f04bd3baee84a12e98098f-Paper.pdf.

Wachsmuth, Henning, Khalid Al-Khatib, and Benno Stein (2016). 'Using Argument Mining to Assess the Argumentation Quality of Essays'. In: *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan: The COLING 2016 Organizing Committee, pp. 1680–1691. URL: https://www.aclweb.org/anthology/C16-1158.

Wachsmuth, Henning et al. (2018). 'Argumentation Synthesis following Rhetorical Strategies'. In: *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, pp. 3753–3765. URL: https://www.aclweb.org/anthology/C18-1318.

Wang, Wenhui and Baobao Chang (2016). 'Graph-based Dependency Parsing with Bidirectional LSTM'. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 2306–2315. DOI: 10.18653/v1/P16-1218. URL: https://www.aclweb.org/anthology/P16-1218.

Webber, Bonnie, Markus Egg, and Valia Kordoni (2012). 'Discourse Structure and Language Technology'. In: *Natural Language Engineering* 18.4, pp. 437–490.

Webber, Bonnie and Aravind Joshi (2012). 'Discourse Structure and Computation: Past, Present and Future'. In: *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*. Jeju Island, Korea: Association for Computational Linguistics, pp. 42–54. URL: https://www.aclweb.org/anthology/W12-3205.

Wolf, Florian and Edward Gibson (2005). 'Representing Discourse Coherence: A Corpus-Based Study'. In: *Computational Linguistics* 31.2, pp. 249–287. DOI: 10.1162/0891201054223977. URL: https://www.aclweb.org/anthology/J05-2005.

Yamada, Hiroaki, Simone Teufel, and Takenobu Tokunaga (2019). 'Building a corpus of legal argumentation in Japanese judgement documents: towards structure-based summarisation'. In: *Artificial Intelligence and Law* 27.2, pp. 141–170. DOI: 10.1007/s10506-019-09242-3.

Yanase, Toshihiko et al. (2015). 'Learning Sentence Ordering for Opinion Generation of Debate'. In: *Proceedings of the 2nd Workshop on Argumentation Mining*. Denver, CO: Association for Computational Linguistics, pp. 94–103. DOI: 10.3115/v1/W15-0512. URL: https://www.aclweb.org/anthology/W15-0512.

Ye, Yuxiao and Simone Teufel (2021). 'End-to-End Argument Mining as Biaffine Dependency Parsing'. In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Online: Association for Computational Linguistics, pp. 669–678. URL: https://www.aclweb.org/anthology/2021.eacl-main.55.

Yimam, Seid Muhie et al. (2013). 'WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations'. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Sofia, Bulgaria: Association for Computational Linguistics, pp. 1–6. URL: https://www.aclweb.org/anthology/P13-4001.

Yin, Yongjing et al. (2019). 'Graph-based Neural Sentence Ordering'. In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 5387–5393. DOI: 10.24963/ijcai.2019/748.

Yuan, Zheng and Ted Briscoe (2016). 'Grammatical error correction using neural machine translation'. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, pp. 380–386. DOI: 10.18653/v1/N16-1042. URL: https://www.aclweb.org/anthology/N16-1042.

Zhang, Fan and Diane Litman (2015). 'Annotation and Classification of Argumentative Writing Revisions'. In: *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 133–143. DOI: 10.3115/v1/W15-0616. URL: https://www.aclweb.org/anthology/W15-0616.

Zhang, Fan et al. (2017). 'A Corpus of Annotated Revisions for Studying Argumentative Writing'. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 1568–1578. DOI: 10.18653/v1/P17-1144. URL: https://www.aclweb.org/anthology/P17-1144.