CHAPTER 5

Finding Claims

Unlike many of the standard tasks in NLP, argumentation mining is not a single unified process, but a constellation of subtasks, which are of different prominence depending on the goals of the underlying target application. For a (hypothetical) example, in order to obtain the gist of a Twitter conversation, it can be sufficient to extract claims and purported evidence material. By contrast, in order to run a deep analysis of argumentation strategies in left-wing vs. right-wing newspapers, the editorials ideally need to be mapped to a complete constellation of statements, their argumentative relationships, and the reasoning patterns underlying those relationships.

Therefore, similar to the surveys of Peldszus and Stede [2013] and Lippi and Torroni [2016a], we break the overall problem into a set of individual subtasks. In this chapter, we address the core task of finding the claim statements made by the writer, i.e., the central component of an argument. The next chapters will be concerned with finding additional components (viz., supporting and objecting statements), organizing them in a structured representation, and assessing some 'deeper' aspects of the argumentation.

Depending on the application scenario, before finding argument components it can be necessary to first filter a set of input texts as to whether they might contain an argument at all or not. Thus, in Section 5.1, we consider the task of classifying a text as argumentative or non-argumentative. This step applies both to full documents and to portions of text: many texts contain both argumentative and other material, which then can be separated.

Another potential preparatory step concerns the delimitation of potential minimal units of argument analysis, i.e., those units that may become argument components. In practice, this is not often operationalized as a separate task, but nonetheless we briefly address it in Section 5.2.

Then, in Section 5.3, we turn to the central task of identifying *claims* in a text: what does the writer argue *for*? Many implemented approaches combine this step with also finding the other core component of an argument—the supportive statements—but since for some purposes, just finding claims can be the most relevant problem, we discuss the two steps separately.

Throughout this and the following chapters, we mention both work involving human annotation (albeit to a smaller extent), and automatic processing, including relevant datasets, methods that have been applied, and the performance results that were achieved. Along the way we will see that there are several parallels to well-known NLP problems, such as opinion mining, semantic relation extraction, stance classification, and discourse parsing. Accordingly, many computational techniques can be borrowed from those areas.

5.1 CLASSIFYING TEXT AS ARGUMENTATIVE VS. NON-ARGUMENTATIVE

The question "Argumentative or not?" can be asked for a complete text document or for particular parts of a text. As a special case of the latter, portions of dialogic interaction can be deemed argumentatively-relevant. We discuss these three situations in turn.

5.1.1 DOCUMENT LEVEL

When dealing with 'traditional' kinds of documents, the task can be couched as one of *genre classification*. In linguistics, a genre is conceived as a class of texts that

- serve the same purpose,
- often have a typical structure in terms of 'zones' playing specific roles for the common purpose (cf. Section 2.6.2), and
- can have characteristic distributions of linguistic features.

Examples of fairly 'prototypical' genres are recipes, weather reports, and persuasive essays. In each case, we expect certain pieces of information, linearized in a suitable or conventionalized way.

Newspaper (and other print) text In NLP, automatic genre classification has been applied, for instance, to newspapers, where opinionated text (editorials, letters to the editor) is to be distinguished from non-opinionated news. Even though not every opinionated text is necessarily argumentative, the ability to distinguish categories like these is of relevance to argumentation mining, too.

The early work of Karlgren and Cutting [1994] and Kessler et al. [1997] on the Brown Corpus aimed at distinguishing various categories from each other, with one setting being 'editorial' vs. 'reportage'. These researchers experimented with part-of-speech (PoS) tags, document and sentence length, type/token ratio, punctuation symbols, and frequencies of specific words that the authors considered relevant: 'therefore', 'I', 'me', 'it', 'that', and 'which'. Kessler et al. reported classification accuracies of 83% and 61% for reports and editorials, respectively.

Later work replaced linguistic analysis by bag-of-words [Freund et al., 2006] or bag-of-character-n-grams [Sharoff et al., 2010] models. Such knowledge-free approaches yielded very good results in the domains of the training data, but, as shown by Petrenz and Webber [2011], this idea is extremely vulnerable to shifts in topics and domains. Working with the *New York Times Annotated Corpus (NYTAC)*,[1] Petrenz and Webber reimplemented earlier approaches, built subsets of articles covering different topics, and demonstrated that the early work using linguistic features is more robust when the topics of the articles change.

[1] https://catalog.ldc.upenn.edu/ldc2008t19 (accessed May 28, 2018)

In the same spirit, Krüger et al. [2017] used texts with different topics, and they also varied the newspaper and the time of origin of the texts, by working with the Brown, NYTAC, and Wall Street Journal corpora. For distinguishing editorials and letters to the editor from news texts, they experimented with a large linguistic feature set and showed that it generally outperforms PoS and bag-of-lemma approaches. Among the most predictive features were first- and second-person pronouns, negation suffixes, and certain word classes such as sentiment words and communication verbs. The results differ quite a bit among the various settings they considered. We mention here just one: When classifying 'opinion' vs. 'news', trained on the mixed-topic WSJ corpus and tested on NYTAC 'medicine' articles, the F-score for the best model was 0.87.

Web text To assist a user who is interested in different viewpoints on a controversial topic, Roitman et al. [2016] devised a document retrieval method for Wikipedia articles. First, this system finds articles that address the desired topic, by means of current information retrieval techniques. In a follow-up step, the document set is reranked according to its argumentative potential. To this end, the authors employ a manually-built list of words that can point to controversy, including 'justify', 'deny', and others.

Although Wikipedia text is relatively orderly, the majority of web text to be exploited for argumentation mining is generated by users and thus can pose well-known challenges for automatic processing. Habernal and Gurevych [2017] compiled a corpus of such texts, designed to cover a set of predefined topics within the domain of education. Their work includes an annotation study where contributions to online forums and user comments had to be judged as either persuasive or non-persuasive.[2] For a document to be considered persuasive, it also has to be on-topic, so that in effect a second classification dimension is mixed into the task. Three annotators labeled 990 documents and achieved a Fleiss κ of 0.59. This step lead to a gold-standard set of 524 persuasive on-topic documents. As a common source of disagreement, the authors note the presence of implicit claims or stances, thus involving interpretation on the side of the annotators. Habernal and Gurevych then built an automatic SVM classifier using lexical n-gram baseline features and obtained a macro-F score of 0.69. In a follow-up experiment, they used a rich feature set, but the results turned out to be worse than the baseline.

5.1.2 SUB-DOCUMENT LEVEL

In principle, the attribute of being 'argumentative' can also be applied to paragraphs or other stretches of text, but to our knowledge this has not been explicitly addressed in the research yet. Instead, the task is commonly tackled on the level of sentences or clauses.

[2]While in general not every argument aims at persuasion, within the genre considered here, 'argumentative' and 'persuasive' can be considered as referring to the same class of texts.

Sentences The majority of research on fine-grained argumentativity classification is concerned with finding sentences in monologue text, but the task has been framed in a variety of slightly different ways.

We begin with some pioneering work in argumentation mining by Moens et al. [2007]. They worked with texts from AraucariaDB [Reed et al., 2008a] and with court decisions from the European Court of Human Rights (ECHR), and as their first analysis step classified sentences as argumentative or not. (Some examples of argumentative sentences from the ECHR corpus are shown later in Figure 5.1 on page 65.) As features they used lexical (token n-grams, adverbs, verbs, modals, argumentative markers) and syntactic (punctuation, depth of parse tree, number of subclauses) features as well as some text statistics, such as length of sentence and position of sentence in the text. With multinomial naive Bayes and maximum entropy models, they achieved accuracies of 73% and 80%, respectively, for the two data sets.

Florou et al. [2013] worked with Greek texts concerning public policy-making, to identify sentences expressing a position toward a proposal. The authors specifically proposed that "future and conditional tenses and moods often indicate conjectures and hypotheses which are commonly used in argumentation techniques such as illustration, justification, rebuttal", and they studied these features in tandem with a manually compiled list of connectives signaling justification, explanation, deduction, rebuttal, and conditionals. With a C4.5 decision tree classifier, their best result is an F-score of 0.76 for a subset of the morphosyntactic and lexical features.

Among the first researchers applying the task to social media, Goudas et al. [2014] classified Greek sentences from blogs and other web sites as "containing an argument or not". They used a rich feature set including certain PoS distributions, position, and length features, manually selected cue words, the number of entities mentioned in previous sentences, as well as token n-grams. With a logistic regression classifier, they obtained an accuracy of 77%. Turning to Twitter, Dusmanu et al. [2017] used a feature set including n-grams, Twitter-specific tokens like emoticons, dependency triples, and sentiment information for classifying tweets as argumentative or not, and they report an F1-score of 0.78, obtained by a linear regression classifier.[3]

Al-Khatib et al. [2016a] suggest a distant supervision approach to classifying argumentative units: they automatically labeled text from idebate.org as argumentative or not, depending on which of the pre-defined components in the forum the text belongs to.[4] On this data, they train a classifier that uses n-grams, constituency syntax production rules, as well as various morphosyntactic features. They evaluated the model on the same type of data (obtaining an accuracy of 0.92), as well as on two other corpora: the persuasive essay corpus of Stab and Gurevych [2014a] and the web text corpus of Habernal and Gurevych [2017] (both mentioned in our list in Section 4.2.2). On those corpora, the accuracy is 0.67 and 0.88, respectively. The authors then also report results on cross-domain train/test experiments.

[3]The DART data set [Bosc et al., 2016a] contains 4,000 Twitter posts that annotators identified as non-/argumentative (defined as "(not) containing an opinion"). In addition, around 1,900 argumentative tweets were grouped by semantic similarity (topics) and then, within groups, pairwise linked via the relations support (446 tweets), attack (122), and unknown (1,323).
[4]The resulting corpus is available for research: `https://www.webis.de/data` (accessed May 28, 2018).

Student essays are another very popular genre for argumentation mining, where the ultimate goal is to evaluate the quality of the essay and hence of the argumentation therein. Following this direction, Song et al. [2014] posed the non-/argumentative distinction in a more specific way: they aimed to classify whether a sentence addresses a critical question of an argumentation scheme (in the sense of Walton, see Section 3.3) and can thus be considered argumentatively relevant. Features were n-grams of the sentence and the two surrounding ones, sentence position and length, PoS tag presence, and lexical overlap between the sentence and the writing prompt underlying the essay. The performance of their logistic regression model was measured by Cohen κ and was in the range of up to 0.5 in various train/test settings involving the same or different prompts.

While the approaches discussed thus far performed binary classification (argumentative or not), it is also possible to solve the problem as a byproduct of running a multi-class classifier that detects the specific argumentative role of a sentence, and also allows for a 'none' value. An example for this technique is the system by Stab and Gurevych [2014b], which is also trained and tested on student essays. Using a very rich set of features (structural, lexical, syntactic, and specific lexical cues) they experimented with several classification techniques, leading to a best result of 77.3% accuracy (macro-F 0.73) obtained by an SVM. The F-score for the class 'none' was 0.88. It turned out that the text-statistics and structural features are most helpful, which points to the role of conventionalized writing patterns in the genre of student essays. More details on this classifier will be provided below on page 70.

Sub-sentences Presupposing that sentences are the proper unit for making the non-/argumentative distinction is for many use cases a simplification. As, for example, Palau and Moens [2009] or Lawrence et al. [2014] pointed out, a complex sentence may very well contain more than one argument component (i.e., a premise and a conclusion), and furthermore it may contain both argumentative and non-argumentative material, which ought to be kept separate when a fine-grained analysis is being targeted. For automatic analysis this is a complication, though, and hence it is not uncommon (see, e.g., Song et al. [2014]) to let human annotators mark arbitrary text spans as argumentative, but then for the automatic approach to reduce the fine-grained annotation to sentence-level.

Lawrence et al. [2014] argue that the initial step of non/-argument classification should be to separate the text into a sequence of individual 'propositions', by which the authors mean a sequence of words that is not necessarily delimited by punctuation or on syntactic grounds.[5] They tokenize the text, compute a feature set for each token, and train two classifiers for identifying tokens starting and ending a proposition, respectively. Features are the word, its PoS tag and length, and the two words preceding and following the word in question. The results were not very encouraging, though: on a cleaned version of a book chapter, this approach leads to an accuracy of 20% of correctly identified propositions (with exact matching), and the figure even

[5] Hence, like in the other work reported here, the authors are *not* aiming at any semantic analysis when speaking of 'propositions'.

decreases (surprisingly) as further chapters are added as training data. Lawrence et al. do not make the decision on the argumentativeness of propositions directly; rather it occurs as a side effect of classifying argument components (to be discussed in the next chapters). Any 'leftover' material from that step is considered non-argumentative.

5.1.3 SUMMARY

The performance figures quoted above are generally not bad, but when considering the features that are being used for classification, we largely find 'the usual suspects', including bag-of-words, PoS tag distributions, and positional and length information. This indicates that most approaches are geared toward separating a domain-specific document set in two parts, rather than to specifically capturing the linguistic signals of argumentativeness. Among the exceptions is the idea of manually compiling lists of words indicating controversy [Roitman et al., 2016], and of employing larger sets of linguistically motivated features. The latter approach, however, led to mixed results: it served the purpose of separating editorials from news in Krüger et al. [2017] but did not perform well on web text in Habernal and Gurevych [2017]. At any rate, the standard features appear to capture more general aspects of subjectivity and opinion, while not necessarily those of argumentation in particular. Hence, it is not clear yet whether a relatively domain-independent separation of argumentative and non-argumentative text spans can in fact be achieved using standard surface-based features.

5.2 SEGMENTING TEXT INTO ARGUMENTATIVE UNITS

The task of segmentation amounts to partitioning an argumentative text (or portion of text) into units that can later on be identified as playing a certain role—or none—in the argumentation. This is difficult, though, and from the perspective of NLP practitioners, it is tempting to effectively circumvent this segmentation step by using sentences as the 'default' unit. This strategy has two obvious consequences.

- Argumentative units that are smaller than sentences cannot be processed:

  (5.1) [Although the candidate has a few good ideas,] [you should not vote for her!]

- Argumentative units that consist of more than a single sentence are difficult to process:

  (5.2) [We need to tear the building down.]$_1$ [It is contaminated with asbestos.]$_2$ [In every single corner.]$_3$ [From the first to the last floor!]$_4$

The first problem is illustrated above in Example 5.1. The second problem can be particularly acute when working with user-generated texts, where the 'sentence' can be hard to define, and material between punctuation marks can be very short. In Example 5.2 (taken from Peldszus and Stede [2013]), statement 1 is the claim made by the speaker, and the evidence is provided by the sequence 2–4; there is no point in assigning an argumentative role to each

sentence individually. Hence, a sentence-oriented system needs a way to distinguish between a sequence of sentences that each provide different evidence (and which play the same role for the argument), and a sequence of sentences that collectively form a single argumentative unit, as in Example 5.2.

So, in principle, the minimal units of argumentation may span multiple full sentences, or be shorter than a sentence, and thus we can characterize an *argumentative discourse unit (ADU)* as

> a span of text that plays a single role for the argument being analyzed, and is demarcated by neighboring text spans that play a different role, or none at all.

Human annotation Taking our definition seriously would mean to assign the question of demarcating units to human annotators, and then try to reproduce it by automatic means. A few researchers took the first step and produced datasets where annotators have been asked to delimit the ADUs, without setting any restrictions on pre-defined unit candidates. For example, Wacholder et al. [2014], in their annotation of 'callout' and 'target' in online interactions (see Section 4.2), evaluated cases where annotators disagreed only by few tokens, and devised a strategy for deriving a gold standard from their annotations. Also, in the web text corpus of Habernal and Gurevych [2017] (see Section 5.3.5), boundaries were decided freely by annotators (yet in their automatic approach, the authors re-simplified the task to working with sentences only).

Computation Among the few approaches that actually tackle the segmentation problem (instead of working with sentences as default units), some employ a syntax parser to supply clauses, which then serve as candidates for minimal units. For instance, the early work of Palau and Moens [2009] on legal text used syntactic clauses as units to be classified as argumentative or not. More recently, Persing and Ng [2016] used manually-devised rules for filtering parse trees and obtaining argumentative units for analyzing student essays. They worked with the corpus compiled by Stab and Gurevych [2014a] (see Section 4.2.2), which has been annotated on the level of clauses. Persing and Ng extracted ADU candidates from a parse tree, and in their evaluation, they accounted for boundary mismatches with two separate measures: one for exact matching, and one for partial matching, where success is defined as 50% of the tokens being in agreement. Their overall F-score for argument component identification is 0.57 with partial match and 0.47 for exact match, where the difference indicates the difficulties with exact ADU boundary detection. However, on the same corpus, Eger et al. [2017] perform a comparative study of various approaches, where the best one (an LSTM-based dependency parser) achieves an F1-score of 0.91 for finding the exact boundaries of argument components. The authors also provide an informative error analysis, and they point out that their result even beats the human agreement that was determined by Habernal and Gurevych [2017] as 0.89.

A recent thorough study of the segmentation problem was done by Ajjour et al. [2017]. These authors experimented with different feature sets as well as different machine learning models, and were also interested in cross-domain performance. For this reason they worked with

three different available datasets (student essays, news editorials, and web discourse). In terms of features, it turned out that for in-domain analysis, bag-of-words features are most helpful (whereas embeddings did generally not help), while structural features (whether the token is at beginning/middle/end of a sentence/clause/phrase) proved most robust when training on one domain and testing on the other. Regarding machine learning approaches, a bidirectional LSTM achieved the best results in most cases, regardless of the domain or the features.

An important observation, however, was that the cross-domain scenario suffered from generally bad performance. The authors take this as an indication that the notion of ADU is not quite the same across different corpora, and hence across different annotation guidelines. Evidence for this hypothesis is the high variance that Aijour et al. found in the size of ADUs, ranging from clause-like segments in the newspaper data to the frequent occurrence of multiple sentences in a Wikipedia data set by Aharoni et al. [2014] (see Section 5.3.4).

5.3 IDENTIFYING CLAIMS

Detecting claims is the first of two indispensable tasks for any argumentation mining system (the second one being the detection of evidence or premises), and accordingly, a lot of work has been undertaken here. However, the reader should also be reminded that in many instances of argumentation, there is no explicit claim being formulated at all; instead the reader needs to derive it from what is said. In fact, Habernal and Gurevych [2017] report that in their social media data, almost half of all claims are only implicit. The ensuing problem of actually *inferring* claims is obviously very difficult and has not received much attention. Instead, the research has focused on the simpler task of only identifying explicit claims.

We can broadly distinguish two families of methods that have so far been applied to claim detection.

Classification Given a minimal unit of analysis (in practice this is almost always a sentence, as described in the previous section), it can be classified in different ways.

- Binary classification: claim or no claim.
- Binary classification: claim/premise. When the overall goal of the system is restricted to finding claims and premises, thus adopting a coarse-grained definition of argument, then the classifier can directly distinguish between these two types of unit.
- Multi-class classification: When more types of argument components are being considered, they may be identified by a single classifier. Also, as pointed out earlier, one class can be 'none', so that the decision whether the unit is argumentative at all is also included here.

Sequence labeling Some approaches tackle the identification of argument components (claims and possibly more) as an IOB labeling problem.[6] Thus, words are tagged as B-premise, B-claim, I-premise, I-claim, or O. Again, this can subsume the detection of non-argumentative material. Also, when claim detection is the *only* task performed by a system, it is trivially identical to the task of non-/argumentative classification. We will discuss this type of work in the present section (rather than above in 5.1), because the definitions used for 'claim' in this work are more specific than those generally adopted for 'argumentative'.

We organize the following summary by the genres being addressed, as these imply certain differences as to what exactly a 'claim' is.

5.3.1 LEGAL DOCUMENTS

{ [**SUPPORT:** The Court recalls that the rule of exhaustion of domestic remedies referred to in Article x of the Convention art. x obliges those seeking to bring their case against the State before an international judicial or arbitral organ to use first the remedies provided by the national legal system.
CONCLUSION: Consequently, States are dispensed from answering before an international body for their acts before they have had an opportunity to put matters right through their own legal systems.]

[**SUPPORT:** The Court considers that, even if it were accepted that the applicant made no complaint to the public prosecutor of ill-treatment in police custody, the injuries he had sustained must have been clearly visible during their meeting.
AGAINST: However, the prosecutor chose to make no enquiry as to the nature, extent and cause of these injuries, despite the fact that in Turkish law he was under a duty to investigate see paragraph above.
SUPPORT: It must be recalled that this omission on the part of the prosecutor took place after Mr Aksoy had been detained in police custody for at least fourteen days without access to legal or medical assistance or support.
SUPPORT: During this time he had sustained severe injuries requiring hospital treatment see paragraph above.
CONCLUSION: These circumstances alone would have given him cause to feel vulnerable, powerless and apprehensive of the representatives of the State.]
CONCLUSION: The Court therefore concludes that there existed special circumstances which absolved the applicant from his obligation to exhaust domestic remedies. }

Figure 5.1: Annotated text from an ECHR decision based on Palau and Moens [2009].

A prominent example for argumentation in legal texts is a court justifying its ruling in the decision document. In Figure 5.1, we reproduce an annotated example from Palau and Moens [2009], who worked on decisions of the European Court of Human Rights (ECHR). The bracketing indicates the hierarchical structure (to some extent): the first two paragraphs represent sub-arguments that both are given in support of the overall conclusion in the third paragraph. Thus, 'conclusion' in their terminology corresponds to what we call a 'claim'. For the central conclusions, the surface form "The court concludes that X" or some paraphrase can be regarded as typical; but notice that the interim conclusions can be descriptions of various kinds, in fairly general form.

[6] IOB stands for inside-outside-beginning. This method represents labeled "target" text spans by tagging the tokens of a text with one of the three letters: B = token is the beginning of a span; I = token is within a span ('inside'); O = token is not part of a span ('out'). As an example, consider finding all named entities (NE) in a text. B indicates that NE begins, I indicates that an NE continues: *We-O met-O in-O the-O New-B York-I City-I subway-O.*

Before the task of 'argumentation mining' was identified and labeled as such, legal case documents were analyzed from an information extraction point of view, in an attempt to distinguish the different functions of sentences, very similar to the 'argumentative zoning' idea of Teufel and Moens [2002], which we mentioned in Section 2.6.2. For example, Hachey and Grover [2006] annotated 'rhetorical roles' in judgments of the UK House of Lords, the goal being that of automatic summarization. Seven roles were distinguished, among them 'fact' (recounting the events triggering the legal proceedings), 'background' (citation of source of law material), and 'framing', which represents parts of the judge's argumentation. Using features similar to those of Teufel and Moens (cue phrases, location, entities, sentence length, quotations, thematic words), the authors obtained a micro-averaged F-score of 0.6 with an SVM classifier using all features, but, interestingly, a decision tree classifier using only location features achieved a considerably better score of 0.65.

Palau and Moens [2009] went a step further and distinguished the argumentative roles shown in Figure 5.1. Their SVM model for premise/conclusion classification takes as input sentences that have already been predicted to be argumentative (see our description in Section 5.1). The resulting F-scores for premise and conclusion are 0.68 and 0.74, respectively. Among their features are syntactic ones (subject type, main verb tense), domain-specific cues, token counts, position of sentence, and a contextual feature with the prediction for the previous and the following segment.

More recently, Rooney et al. [2012] proposed an SVM sequence kernel classifier as an alternative to the type of feature engineering done by Palau and Moens. The kernel compares subsequences of sentences, where a word is tagged with its root form and PoS label. These authors worked with the Araucaria DB dataset [Reed et al., 2008a], which contains arguments not only from legal documents but also from newspapers, advertising, and several other genres. The task is to label sentences as containing a premise (1299 instances in the annotated corpus), containing a claim (304 instances), being part of both a premise and a claim (161), or none (1686). Using 10-fold cross-validation, Rooney et al. report an overall accuracy of 0.65, which those authors consider as promising on the grounds of dispensing with sophisticated features. For claims, the result is particularly low, though (around 0.3).

In general, mining legal texts is a very difficult task. For an overview of the limitations of current IR systems and a sketch of necessary steps for automated argument retrieval, see Ashley and Walker [2013a].

5.3.2 INSTRUCTIONAL TEXT

Instructions, as they can be found in user manuals, often include advice and warnings; sometimes, these are backed up by an explanation designed to increase the reader's motivation to actually obey them. Here are some examples from Saint-Dizier [2012].

(5.3) Never put this cloth into the sun; otherwise it will shrink dramatically.

(5.4) It is essential that you switch off electricity before starting any operation. Electricity shocks are a major source of injuries and death.

As such, the structure and function correspond exactly to that of an argument, where the claim is of the specific type 'instruction' to do or not do something. Based on a corpus study covering French and English text, Saint-Dizier concluded that such constructions have a highly regular linguistic form, making it relatively easy to identify them. In his approach, a set of manually-constructed rules serves to identify both the 'claim' and the supporting statements. The rules exploit the linear order of the statements, and a set of common lexical patterns. These patterns may contain expressions on different levels of abstraction such as word form, word set, or part of speech. The rule language is processed by a dedicated text processing platform developed by the author, called <*TextCoop*>. Saint-Dizier reports recognition accuracies of 88%/91% (claim/support) for warnings and 79%/84% for advice.

5.3.3 STUDENT ESSAYS

Essays written by students in response to a given prompt are a target of NLP primarily for the application of automatically scoring them, or for supporting the human grader or peer-reviewer by automatically adding helpful markup (and thereby encouraging the grader to provide qualitative feedback). A significant portion of essays in education are persuasive, which Burstein and Marcu [2003, p. 457] define as requiring "the writer to state an opinion on a particular topic, and to support the stated opinion to convince the reader that the perspective is valid and well-supported."

We can broadly distinguish two lines of work here, which use slightly different labeling schemes.

Thesis and conclusion Especially the American tradition of essay writing encourages students to make sure that their texts contain a 'thesis' and a 'conclusion'. According to Falakmasir et al. [2014, p. 254], a thesis "communicates the author's position and opinion about the essay prompt; it anchors the framework of the essay, serving as a hook for tying the reasons and evidence presented and anticipates critiques and counterarguments", whereas the conclusion "reiterates the main idea and summarizes the entire argument in an essay. It may contain new information, such as self-reflections on the writer's position."

Finding these two elements is made difficult by two features of student essays: as opposed to other genres like the legal texts discussed above, or scientific articles, student essays have little internal structure in terms of headings and subheadings. Furthermore, topics vary widely, and the students' theses are often substantiated by personal experience rather than by cited sources or authorities. Thus, the role of genre-specific wording can also be expected to be less helpful than for other texts. However, we will see that positional features can play an important role for the task.

"You can't always do what you want to do!," my mother said. She scolded me for doing what I thought was best for me. It is very difficult to do something that I do not want to do. **<Thesis>**But now that I am mature enough to take responsibility for my actions, I understand that many times in our lives we have to do what we should do. However, making important decisions, like determining your goal for the future, should be something that you want to do and enjoy doing.**<Thesis>**

I've seen many successful people who are doctors, artists, teachers, designers, etc. In my opinion they were considered successful people because they were able to find what they enjoy doing and worked hard for it. It is easy to determine that he/she is successful, not becaue it's what others think, but because he/she have succeed in what he/she wanted to do.

In Korea, where I grew up, many parents seem to push their children into being doctors, lawyers, engineer, etc. Parents believe that their kids should become what they believe is right for them, but most kids have their own choice and often doesn't choose the same career as their parent's. I've seen a doctor who wasn't happy at all with her job because she thought that becoming doctor is what she should do. That person later had to switch her job to what she really wanted to do since she was a little girl, which was teaching.

<Conclusion> Parents might know what's best for their own children in daily base; but deciding a long term goal for them should be one's own decision of what he/she likes to do and want to do. **<Conclusion>**

Figure 5.2: Annotated essay from Burstein and Marcu [2003, p. 457], used with permission.

Figure 5.2 shows an essay annotated with thesis and conclusion, taken from the early work of Burstein and Marcu [2003]. Their annotation rules state that both components can be one or more sentences, but no sub-sentences. For their decision-tree classifier, the authors used various positional features of sentences and paragraphs, syntactic clause types, and a set of manually-defined cue words. One group contained connectives (e.g., *first, in summary, in conclusion,* etc.), the other included modals and other lexical elements such as *agree*. Furthermore, the output of an early, cue-based discourse parser [Marcu, 2000] was mapped to two sentence features: the nuclearity status and the coherence relation the sentence takes part in (according to Rhetorical Structure Theory [Mann and Thompson, 1988]). Feature utility tests were not reported. A crucial question for an approach as lexically-based as this one is how to transfer to a new content domain, i.e., a new essay prompt. When evaluating on a prompt that was not present in the training set, the approach reached an average F-score of 0.54 for thesis and 0.8 for conclusion segments.

Falakmasir et al. [2014] tackled the same problem and worked with 432 essays (responding to 8 different prompts) for training and test, with the prompts in the 2 sets being disjoint. Human annotators identified sentences that were candidate thesis or conclusion statements, and rated their quality on a scale (1: vague or incomplete; 2: simple but acceptable; 3: sophisticated). The

central goal of this work was feature engineering, and the authors used an iterative process to find the most predictive features: starting with 42 basic features (position, syntax, cues, rhetorical status) inspired by the Burstein/Marcu work, they employed several feature selection algorithms and experimented with various combinations. Finally, they converged on the following.

- **Positional** features: sentence number in the paragraph, paragraph number in text, and type of paragraph (first, body, and last paragraph). A positional baseline predicts all sentences in the first paragraph as a thesis and all sentences within the last paragraph as a conclusion.
- Various features based on the **syntactic/semantic analysis and dependency parsing** of the sentence. Prepositional and gerund phrases turned out as highly predictive of thesis and conclusion sentences, as did the number of adjectives and adverbs within the sentence.
- A set of **frequent words** (e.g., *although, even though, because, due to, led to, caused*).
- **Essay-level** features: number of keywords among the most frequent words of the essay, number of words overlapping with the assignment prompt, and a sentence importance score based on RST (similar to the feature used by Burstein and Marcu [2003], mentioned above).

A three-way classification experiment (thesis, conclusion, other) was performed, where thesis/conclusion sentences whose quality had been rated 1 were shifted to the 'other' category. Among three tested methods a decision tree classifier produced the best F-measures (0.83 for thesis and 0.59 for conclusion on the test set).

In a follow-up study on the same dataset, Jabbari et al. [2016] focused on finding thesis statements, and noted that the central challenge in the previous work was the skewed distribution of non-/thesis sentences, which had been tackled by means of complex feature optimization. As an alternative, they now proposed to instead balance the training data distribution. With random under- and oversampling as baselines, their experiment centered on the SMOTE approach for generating additional 'synthetic' examples [Chawla et al., 2002]. The idea of that technique is to produce new instances not on the text but on the feature vector level, by interpolating between existing minority instance vectors. The classification problem was somewhat simplified by considering only sentences in the first paragraph of the text (where theses commonly show up). Using an SVM, for finding theses the authors achieved a micro-F-score of 0.9, which compares favorably to the earlier work as well as to the two baselines (0.84 with undersampling; 0.87 with oversampling).

Major claim, claim, and premise A second line of research was started with the Persuasive Essay Corpus [Stab and Gurevych, 2014a], which we introduced in Section 4.2.2. According to the annotation scheme, an essay can have one *major claim*, which expresses the author's stance with respect to the topic, and which is usually found in the first paragraph. (It appears to correspond to the *thesis* statement in the scheme explained above.) Two examples [Stab and Gurevych, 2014a, p. 1503] are as follows.

(5.5) I believe that **we should attach more importance to cooperation during education.**

(5.6) From my viewpoint, **people should perceive the value of museums in enhancing their own knowledge.**

Then, the paragraphs between the introduction and the conclusion usually contain arguments related to the major claim. These consist of *claims* and *premises*, which (in contrast to the work discussed above) are annotated not as complete sentences but as continuous token sequences, cf. the bold-faced portion of the examples above. Altogether, the corpus has 90 major claims (one per text) and 429 claims.

The detection of argument components is implemented as a four-way classifier (major claim, claim, premise, non-argumentative) presented by Stab and Gurevych [2014b], which was mentioned in Section 5.1.2. Features can be grouped as follows.

- **Structural:** location/length/punctuation.
- **Lexical:** n-grams, verbs, adverbs, modals.
- **Syntactic:** parse tree depth, production rules, verb tense.
- **Cues:** connectives from the Penn Discourse Treebank (PDTB) corpus [Prasad et al., 2008], first-person references.
- **Attributes of preceding and following sentence:** number of tokens and of punctuation, number of sub-clauses, presence of modal verbs.

Among various classifiers, an SVM performed best. The F-scores are: major claim 0.63, claim 0.54, premise 0.83, non-argumentative 0.88. While premises can be identified quite reliably also when using just a single group of features (F between 0.65 for syntactic and 0.78 for structural features), the corresponding values for claims and major claims are much lower, the maximum being 0.48 (major claim) and 0.42 (claim) for structural features. In other words, claim detection profits more from feature combination than premise detection does.

In later work on the same corpus, Nguyen and Litman [2015] describe an alternative approach that centers on lexical/cue features. Their idea is to separate the argumentative vocabulary from that indicating the domain. To do this, they use LDA topic modeling [Blei et al., 2003] for learning domain and argumentation vocabulary from a separate corpus of 6794 essays (i.e., excluding those from the Stab/Gurevych evaluation corpus), starting with sets of seed words. For argumentation, the set consists of 10 words. In Figure 5.3, we reproduce three of the top-ranked argumentation (Topic 1) and domain (Topic 2, 3) word stems produced by their algorithm. Altogether, 263 argument words are generated.

For the new classification model, Nguyen and Litman reimplemented the features of Stab/Gurevych, excluding the n-grams and the production rules. Instead, they added the newly harvested argument word unigrams, dependency pairs (as a replacement for skipped bigrams),

Topic 1 *reason exampl support agre think becaus disagre statement opinion believe therefor idea conclus*
Topic 2 *citi live big hous place area small apart town build communiti factori urban*
Topic 3 *children parent school educ teach kid adult grow childhood behavior taught*

Figure 5.3: LDA-topics (word stems) generated for student essays [Nguyen and Litman, 2015, p. 25], used with permission. Topic 1 shows argument words; Topic 2 and 3 are domain words.

and numbers of argument and domain words. Due to the removal of n-grams, the model has only 1/5 of the size of the Stab/Gurevych reimplementation. The authors show that the new model outperforms the reimplementation (and the results that had been reported by Stab/Gurevych), especially when only top-performing features are used.

5.3.4 WIKIPEDIA

A different perspective on claim detection is taken by Aharoni et al. [2014] and Levy et al. [2014]. These researchers work on the IBM *Debater* project (cf. Sct. 1.3), whose overall goal is a system that searches web pages for pro and con arguments on a given controversial topic. Thus, the claim detection task is dependent on a predefined *topic*, which Levy et al. (p. 1489) define as "a short phrase that frames the discussion"; and a context-dependent claim is "a general, concise statement that directly supports or contests the given topic." The text source is Wikipedia pages, where (in contrast to other web text) a certain quality of language and argumentation can be presumed.

A dataset was constructed on the basis of 32 debate motions from *Debatabase*.[7] Then, 326 Wikipedia articles were identified as relevant to these topics, and therein, annotators labeled 976 topic-related claims, achieving a rather moderate agreement of κ=0.39. Figure 5.4 shows examples for annotation decisions. (Notice that, in other terminology, this 'topic' would itself qualify as a 'claim', as it represents a thesis or demand by the author.) A 'V' indicates that a sentence was regarded as a topic-dependent claim, while 'X'-marked sentences were not.

In general, claims need not be complete sentences but can be smaller units. The automatic claim detection system thus needs to find, given a Wikipedia page and a topic, sentences and sub-sentences that qualify as a relevant claim. To achieve this, Levy et al. built a pipeline of three modules.

[7] `http://www.idebate.org/Debatabase` (accessed May 28, 2018). Upon request, the authors are making this dataset available for research.

Topic: The sale of violent video games to minors should be banned		
S1	Violent video games can increase children's aggression	V
S2	Video game addiction is excessive or compulsive use of computer and video games that interferes with daily life	X
S3	Many TV programmers argue that their shows just mirror the violence that goes on in the real world	X
S4	Violent video games should not be sold to children	X
S5	Video game publishers unethically train children in the use of weapons	V

Figure 5.4: Topic and associated claim candidates from Levy et al. [2014, p. 1490], used with permission.

The first module is in charge of identifying sentences that contain a claim. A logistic regression classifier uses features exploiting topic relevance (cosine similarity between the topic phrase and subjects in the sentence; and between the topic and a WordNet-expanded version of the sentence) and various topic-independent features (including morphosyntax, subjectivity, sentiment). It passes the top scoring 200 sentences to the next component.[8]

For each sentence found, the second module generates the 10 best candidate sub-sentences using a maximum likelihood model that primarily considers the tokens at the beginning and the end of the sequence. Then a logistic regression classifier selects the most probable claim, again looking at boundary tokens, along with a few other features.

The third module ranks the identified claims found for all the sentences, using another logistic regression classifier. It considers the scores produced by the previous components and also re-computes some of the features that have been used in the preceding modules.

In subsequent work, Shnarch et al. [2017] added a pattern matching approach called GRASP, which integrates information from different layers of analysis to the previous system. GRASP identifies the most discriminative features for the classification task, and composes patterns out of these. The authors note that these patterns are open to human inspection, hence allowing for qualitative error analyses. After adding this approach, significant improvements in performance on claim identification are reported.

Working on a slightly bigger version of the IBM dataset, Lippi and Torroni [2015] addressed only the first step, i.e., sentence classification, and suggested a rather different approach. They started from the observation that argumentative sentences are often characterized by common 'rhetorical structures' and operationalized that intuition as the similarity between syntactic (constituency) parse trees. Their goal thus is to account for argumentative language irrespective of the domain, and to do so on the basis of syntactic patterns. They obtained the parses from

[8] In this dataset, the input to step 1 consists of 1,500 Wikipedia sentences on average.

the Stanford CoreNLP suite, and trained an SVM classifier that exploits a partial tree kernel to compute similarities. The authors conclude that they can obtain results that are competitive to those of Levy et al. [2014], even without considering topic similarity. When adding a single context feature (cosine similarity between the candidate sentence and the topic, both represented as bag of words), the F-measure increases by a further 1.2 points.

In addition, Lippi and Torroni tested their tree-similarity approach on the essay dataset by Stab and Gurevych [2014a] (mentioned in the previous subsection). Considering the union of their categories 'claim' and 'major claim' as the target class, Lippi and Torroni obtained an F-score of 0.71, which compares favorably to the results of Stab and Gurevych (whose problem was a more difficult multi-class task, though).

Finally, we mention that the IBM dataset was used by Laha and Raykar [2016] for experiments with RNNs and CNNs for detecting both claims and evidence (see next section). Their goal was to establish the first deep learning baselines for these tasks, and they report comparisons of various architecture variants. For finding claims, however, the results were not as good as those obtained by Levy et al. [2014].

5.3.5 SOCIAL MEDIA AND USER-GENERATED WEB TEXT

For many practical applications, the various kinds of social media (or other user-generated text) are a highly relevant source of opinions and arguments to be mined. Some early work addressed public comments on proposed government regulations. Specifically, Kwon et al. [2007] studied a corpus of user comments on proposed U.S. Environmental Protection Agency legislation. Here, the claim detection task benefits from all documents addressing the same domain (the legislation in question). The authors exploited this by computing a feature representing the lexical overlap between sentences and the text topic 'public policy', in addition to various standard features (n-grams, frequencies of positive and negative words, position in text, and head verb token sequence). In the experiments, a boosting algorithm beat an SVM and achieved an agreement with a human annotator of κ=0.55. In a follow-up step, the authors then computed the polarity (stance) of the claim by means of a seed word approach that identifies positive and negative words.

In later work, Rosenthal and McKeown [2012] sought to detect claims in LiveJournal weblogs and in Wikipedia discussion pages. They define claims as "assertions by a speaker who is attempting to convince others that his opinion is true" (p. 30), and more specifically attend to the class of claims expressing opinionated beliefs, which are defined as "personal view[s] that others can disagree with" (p. 30). Figure 5.5 shows examples, which demonstrate that especially the LiveJournal text can be very informal. Not surprisingly, it is not trivial to demarcate this class of claims precisely; when two annotators labeled 2,000 sentences as claims (without considering the context) in each corpus, they achieved a Cohen κ of 0.50 for 663 LiveJournal sentences, and 0.56 for 997 Wikipedia sentences.

LiveJournal	1	oh yeah, you mentioned the race … that is so un-thanksgivingish !
	2	A good photographer can do awesome work with a polaroid or ' phonecam .
	3	hugs I feel ike I have completely lost track of a lot of stuff lately .
Wikipedia	4	The goal is to make Wikipedia as good as possible and, more specifically , this article as good as possible .
	5	This was part of his childhood , and should be mentioned in the article .
	6	If the book is POV or the writer has only a slender grasp of relevant issues , material can be wrong .

Figure 5.5: Examples of claims from two online corpora based on [Rosenthal and McKeown, 2012, p. 31].

The focus of the work was on measuring the utility of sentiment and so-called *comitted belief* features. For the latter, the authors adopted the system by Prabhakaran et al. [2010], which tags individual words for the underlying type of belief: committed (*I know*), non-committed (*I may*), and not applicable (*I wish*). Rosenthal and McKeown report various experiments on in-domain and cross-domain classification, including feature impact measurements. In brief, they found that sentiment features are more useful in LiveJournal, while committed belief features are more predictive for Wikipedia discussions.

On a corpus of 204 different social media and news documents (16,000 sentences) in Greek, Goudas et al. [2014] employed IOB sequence labeling for combined claim and premise detection. The features were the words in the sentence, domain-specific named entities, manually compiled cue words, and verbs and adverbs found representative for claim and premise sentences by a TF-IDF computation. Their CRF model achieved a precision of 0.62 and recall of 0.32, resulting in an F-measure of 0.42. In follow-up work on news text, Sardianos et al. [2015] sought to reduce the role of domain-dependent gazetteer lists and report on experiments with word embeddings generated by word2vec [Mikolov et al., 2013].

Recently, Habernal and Gurevych [2017] extended the IOB tagging approach to a set of argument components that they derived from the Toulmin scheme (see Section 3.4.1): backing, claim, premise, rebuttal, refutation. They used the Web Discourse corpus (see Section 4.2.1), whose annotation covers the aforementioned components, plus 'appeal to emotion' (representing the pathos dimension of argumentation); see the example in Figure 5.6. We will come back to this work in subsequent sections, and here just focus on the claim identification. The trained human annotators achieved an agreement of 0.59 on claims, measured by Krippendorf's unitized α.

The 11-class IOB tagging is applied to complete sentences as instances to be classified; the authors argue that a token-level annotation is too fine-grained for a machine learner. Consequently, the problem is simplified to deciding whether a sentence hosts an argument component or not. For sequence labeling, Habernal and Gurevych used SVM^{hmm}. The rich feature set is split into five groups:

Doc#163 (comment, homeschooling) Thank you for bringing this tragedy to light. [*backing:* I am a Christian, an educator, a student and a parent and I have seen too many children like the Powells. As an admissions officer, we had applicants whose "record keeping" consisted of sending boxes full of paper for our office to review as part of the application.] [*premise:* If their students did get an interview, which was rare, they didn't have the social skills to survive the first round.]¶
[*premise:* I personlly am acquainted with four families who are home schooling their large families. All four have no intention of book-schooling their daughters past age 13 as they need to learn :homemaking skills". One of the girls, who has not been taught for two years, could be Josh Powell's twin. She is intelligent and desperate to learn, but her parents won't allow it.] [*app-to-emot:* It is heartbreaking.¶
That the Commonwealth of Virginia has such a rich tradition of the education of young people and allows this travesty is shameful.] All of us, no matter our religious beliefs, need to pray that the law changes before more smart children are left behind.

Figure 5.6: Sample annotation from the corpus of Habernal and Gurevych [2017, p. 169], used with permission.

- **Lexical:** word n-grams;
- **Structure and syntax:** initial and final tokens, relative positions, POS n-grams, dependency tree depth, constituency tree production rules, number of subclauses;
- **Topic and sentiment:** LDA topics and five sentiment categories (-2, …, 2);
- **Semantics and discourse:** semantic role labels, various coreference chain attributes, type of discourse relation, presence of connectives, attributions; and
- **Embeddings:** sums of word embedding vectors.

All features are considered not only for the current sentence but also for the four preceding and four subsequent sentences. The overall best results are achieved with the full feature set, yielding a macro-F score of 0.25, in comparison to 0.6 obtained by human annotators. The classes Claim-B and Claim-I are among the top-performing classes, reaching 0.27 when omitting the lexical and structure/syntax features. As these numbers show, the problem tackled here is much harder than those we discussed before.

5.3.6 SUMMARY: FINDING CLAIMS OF DIFFERENT KINDS

The claim, undoubtedly, is at the heart of the argument—it states what the author wants us to believe. Beyond this fairly general characterization, however, different authors provide somewhat different characterizations, and the corpus examples we have seen make it clear that there is a

relatively broad range of statements treated as claims in different text types and genres. Often, but not always, claims are indicated by linguistic surface features, which some approaches try to isolate as topic-independent 'claim shells' that signal the claim. Also, in some genres, there seem to be relatively strong position-in-document tendencies that can be exploited, as for example in student essays.

Recently, the variety in claim realizations prompted Daxenberger et al. [2017] to run a systematic claim identification experiment across six different datasets.[9] They mapped the existing annotations to the sentence level, thus making a simplifying assumption, which, however, is shared by most of the previous work, as we have pointed out above. Daxenberger et al. wanted to measure the influence of different groups of features (structure, lexical, syntax, discourse, embedding)[10] and learning approaches; they ran three deep learning schemes, and a regularized logistic regression classifier.

For in-domain classification, they obtained an average macro-F score of 0.67, ranging from 0.6 (Wikipedia TalkPages) to 0.8 (arg. microtexts). Lexical (unigram), embedding, and syntax features were most predictive, whereas structural features did not help in most cases. Discourse features were useful only on the microtext corpus. The average performance of the best neural network and the logistic regression classifier was virtually the same.

With cross-domain training and testing, performance drops were found throughout, most pronounced for the datasets that performed best in the in-domain setting. The best feature-based approach outperformed the deep learning approach in most scenarios. However, the neural networks benefit when trained on all datasets but one and testing on the remaining one, yielding the best results on average. Still, the best NN approach did not show benefits over training on good (single) source training datasets.

After studying the six corpora further, Daxenberger et al. concluded that it is difficult to predict cross-domain performance from lexical similarity of the datasets. Their overall conclusion is that "the essence of a claim is not much more than a few lexical clues." This points to the important role of context: depending on genre and topic, a claim may on the surface look like any other declarative sentence, and only in the particular context assume the role of an opinion that the speaker wants to convince the hearers of.

Finally, we have to point out that the notion of claim will be given an additional facet of complication later in Chapter 7 when we discuss structures of complex arguments: a statement in a text may be supported by another statement (which makes the first one a claim) and at the same time it may in turn support yet another statement and thus play multiple roles in the argumentation. But before turning to such recursive structures, we first look more closely at the 'support' relation.

[9]Most of them have been introduced in Section 4.2.2 or will be mentioned in the next chapter: AraucariaDB [Reed et al., 2008a], essays from Stab and Gurevych [2014b], web discourse from Habernal and Gurevych [2017], online comments and Wikipedia talk pages from Biran and Rambow [2011], and argumentative microtexts from Peldszus and Stede [2016b].

[10]They also experimented with sentiment dictionaries (as in, e.g., Rosenthal and McKeown [2012]) but found these to be of very little value.

CHAPTER 6

Finding Supporting and Objecting Statements

Detecting claims, the topic of the previous chapter, is obviously a core task of argumentation mining, but in order to find a complete argument, there is at least one more component to be identified: the statement(s) that the author introduced to support the claim. In the argumentation literature, this is often called the 'premises' of the argument; in other places it is 'evidence' or 'justification'. In this chapter, we use these terms interchangeably with 'support'.

Besides claims and evidence, it may be important to also look for counter-arguments, in case the text also mentions the 'opposing view', and the application seeks to identify it. Here is a simple example.

(6.1) You should buy that camera, because it has a brand-new excellent sensor. OK, it is quite expensive. But it's worth the money!

Our speaker, in giving her recommendation, imagines an 'opponent' pointing out the high price of the product. After quoting that potential counterargument, she refutes it by emphasizing the good value one gets for the money here. Thus, the argument components to be detected are as follows.

- **Claim:** You should buy that camera.
- **Support:** That camera has a brand-new excellent sensor.
- **Objection:** That camera is quite expensive.
- **Support:** That camera is worth the money.

Notice that when 'digging deeper', we might want to not just enumerate the components but also record in what way the speaker wants them to relate to each other (here especially the link between the objection and the second support)—but that is the topic of the next chapter. For now, in the following sections, we look at the tasks of finding support (Section 6.1) and objections (Section 6.2). Furthermore, a potentially helpful related NLP task is *stance detection*, which determines whether a text communicates a positive or negative attitude toward a given topic. We examine it briefly in Section 6.3.