Syntax Matters for Rhetorical Structure: The Case of Chiasmus

Marie Dubremetz
Uppsala University
Dept. of Linguistics and Philology
Uppsala, Sweden
marie.dubremetz@lingfil.uu.se

Joakim Nivre
Uppsala University
Dept. of Linguistics and Philology
Uppsala, Sweden
joakim.nivre@lingfil.uu.se

Abstract

The chiasmus is a rhetorical figure involving the repetition of a pair of words in reverse order, as in "**all** for **one**, **one** for **all**". Previous work on detecting chiasmus in running text has only considered superficial features like words and punctuation. In this paper, we explore the use of syntactic features as a means to improve the quality of chiasmus detection. Our results show that taking syntactic structure into account may increase average precision from about 40 to 65% on texts taken from European Parliament proceedings. To show the generality of the approach, we also evaluate it on literary text and observe a similar improvement and a slightly better overall result.

1 Introduction

There is a growing interest in applying computational techniques within the field of literature as evidenced by the growth of the digital humanities (Schreibman et al., 2008). This field has very specific demands. Unlike many technical fields, literature requires a serious treatment of non-literal language use and rhetorical figures. One of those figures is the antimetabole, or chiasmus of words, illustrated in Figure 1. It consists in the reuse of a pair of words in reverse order for a rhetorical purpose. It is called 'chiasmus' after the Greek letter χ because of the cross this letter symbolises (see Figure 1).

Identifying identical words is easy for a computer, but locating only repetitions that have a rhetorical purpose is not. Can a computer make this distinction? And if yes, which features should we model for that? This paper presents the first attempt to go beyond shallow surface features in order to detect chiasmus. We start from the shallow feature-based algorithm introduced by Dubremetz and Nivre (2015) and extend it with features based on syntactic structure. We train models on the annotated corpora already used in previous work and evaluate on a new corpus. Our results show that both positive and negative syntactic features can improve the quality of detection, improving average precision by almost 25% absolute compared to a baseline system using only shallow features. As a generalization test, we apply the model trained on political discourse to literary text (the Sherlock Holmes novels and short stories) and obtain an improvement of 17% average precision compared to the baseline.

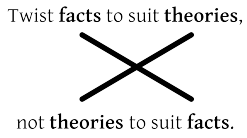

Figure 1: Schema of a chiasmus

2 Related Work

Despite a long tradition in rhetorics and linguistics, the terms *chiasmus* and *antimetabole* do not really have clear definitions. In the earliest times, Diderot and D'Alembert (1782) as well as Quintilian (Greene et al., 2012) give us very basic identification features. They talk about the degree of identity that can be accepted to consider two words as identical (strictly identical strings, lemmas or synonyms). On the other hand, Rabatel (2008) and

Proceedings of the Fifth Workshop on Computational Linguistics for Literature, NAACL-HLT 2016, pages 47–53,
San Diego, California, June 1, 2016.

Nordahl (1971) try to find subcategories of chiasmi on a deep semantic basis: for instance chiasmi expressing contrast (Rabatel, 2008). The notion of antimetabole is floating. Dictionaries of stylistics tend to quote the same prototypical chiasmi to illustrate examples, which is not helpful when trying to capture the linguistic variety of chiasmi. The purpose of the linguists is to define chiasmus compared to other figures (for instance chiasmus as opposed to paralellism). To the best of our knowledge there is no pure linguistic study that tries to distinguish between chiasmus and random repetition of words in a criss-cross manner. In non-computer assisted linguistics, as opposed to computational linguistics, rhetoric is taken for granted. Linguistics has to answer only one question: Which figure is instantiated by this piece of rhetoric? Computational linguistics now has to answer not only this question but also the question of whether a piece of text is a piece of rhetoric in the first place.

Gawryjolek (2009) was the first to tackle the automated detection of repetitive figures and of chiasmus in particular. Following the general definition of the figure, he proposed to extract every repetition of words that appear in a criss-cross pattern. Thanks to him, we know that this pattern is extremely frequent while true positive chiasmi are rare. To give an idea of the rarity, Dubremetz and Nivre (2015) give the example of *River War* by Winston Churchill, a book consisting of $150,000$ words, with $66,000$ examples of criss-cross patterns but only one true positive.[1] Hromada (2011) then proposed to add a feature constraint to the detection: he drastically reduced the number of false positives by requiring three pairs of words repeated in reverse order without any variation in the intervening material. Unfortunately, in the example of Churchill's book, this also removes the one true positive and the user ends up with a totally empty output. Finally, Dubremetz and Nivre (2015) built on the intuition of Hromada (2011) and added features to the detection of chiasmus, but in a different way. They observed that chiasmus, like metaphor (Dunn, 2013), is a graded phenomenon with prototypical examples and controversial/borderline cases such as Example 1.

(1) It is just as contrived to automatically allocate **Taiwan** to **China** as it was to allocate **China**'s territory to **Taiwan** in the past.

Thus, chiasmus detection should not be a binary classification task. Instead, Dubremetz and Nivre (2015) argue that a chiasmus detector should extract criss-cross patterns and rank them from prototypical chiasmi to less and less likely instances.

A serious methodological problem for the evaluation of chiasmus detection is the massive concentration of false positives (about $66,000$ of them for only one true positive in $150,000$ words). Such a needle in the haystack problem makes the constitution of an exhaustively annotated corpus extremely time consuming and repetitive to the extreme. This is analogous to the situation in web document retrieval, where the absolute recall of a system is usually not computable, and where recall is therefore measured only relative to the pool of documents retrieved by a set of systems (Clarke and Willett, 1997). The evaluation of Dubremetz and Nivre (2015) is based on the same principle: in a series of experiments their different "chiasmus retrieval engines" return different hits. They annotate manually the top two hundred of those hits and obtain a pool of relevant (and irrelevant) inversions, on which they can measure average precision to show that chiasmi can be ranked using a combination of shallow features like stopwords, conjunction detection, punctuation position, and similarity of n-gram context. The present work goes beyond the idea of Dubremetz and Nivre (2015). We believe that by using structural features defined in terms of part-of-speech tags and dependency structure, we can improve the average precision of chiasmus detection. Therefore, we will reproduce their algorithm and gradually add new features to check on a new corpus if there is any improvement.

3 Ranking Model and Feature Modeling

We reuse the linear model for prediction developed by Dubremetz and Nivre (2015), which allows the addition of any arbitrary features.

$$f(r) = \sum_{i=1}^{n} x_i \cdot w_i$$

[1] **Ambition** stirs **imagination** nearly as much as **imagination** excites **ambition**.

It is not a **beginning** of the **end**, but an **end** of the **beginning**.
W_a W_b W'_b W'_a

Figure 2: Schema of a chiasmus, W for word.

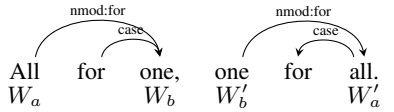

Figure 3: Schematic representation of chiasmus, W for word.

Here r is a string containing a pair of inverted words, x_i is the value of the ith feature, and w_i is the weight associated with this feature. Given two inversions r_1 and r_2, $f(r_1) > f(r_2)$ means that the inversion r_1 is more likely to be a chiasmus than r_2.

3.1 Part-of-Speech Tags

Part-of-speech tagging provides a coarse grammatical analysis of the text, which we can exploit to refine the detection of chiasmus. We model tag features as positive features. Words that are detected in a criss-cross pattern already share the same lemma (base form). As shown in Figure 2, we normally expect W_a to have the same tag as W'_a, and W_b the same tag as W'_b, unless they are ambiguous words that happen to share the same lemma. Unfortunately, this can be true in false positives too, above all in duplicates.[2] What seems more unique in Figure 2 is that all the main words of the prototypical chiasmus have the same tag, Noun in this case. In our tag-based model, we therefore add a weight of $+10$ for a binary feature that is true only if W_a, W_b, W'_b and W'_a all have the same tag.

3.2 Dependency Structures

To further exploit the syntactic structure of a chiasmus candidate, we add features defined over the dependency structure. Our hypothesis is that these features can be both negative and positive. The idea of using syntax as a positive feature is not hard to motivate. If chiasmus is the figure of symmetry (Morier, 1961, p.113), we should see that in the syntax. Symmetry means not only inversion, but also repetition. In Figure 3, we see that W_b has the same role as W'_a (both are the complement of a noun) in a perfectly symmetrical role switching.

It is perhaps harder to see that syntactic dependencies might also play a role as a negative feature, but we motivate this by the remark of Dupriez (2003, art. Antimetabole):

> Metabole consists in saying the same thing with other words, antimetabole saying something else with the same words.

Dupriez (2003) seem to say that for being prototypical the words sharing the same identity in the chiasmus should not be used to express the same thing. Indeed, in Footnote 1, what makes the quote so rhetorical is the fact that 'imagination' and 'ambition' are not repeated with the same role (subject versus verb complement). Therefore, we assume that if the same word is reused with the same syntactic role it is more likely to be a false positive. Example 2 is a false positive found in an earlier experiment: 'convention' is in both cases the direct object of a verb.

(2) We must call on Cameroon to respect this **convention** and find ways of excluding this **country** and any other **country** which violates the **conventions** which it has signed.

Our syntactic features are summarized in Table 1. These features simply count the number of incoming dependency types (labels) that are shared between two words. For example, in Figure 3: 'one' and 'all' share one dependency type (nmod:for).

4 Experiment

Classical machine learning methods cannot be applied as there is no big corpus of annotated chiasmi. The corpus produced by Dubremetz and Nivre (2015) contains about one thousand examples of false positives for only 31 true positives. Therefore, we decided to tune the weights manually, just like in the previous study(Dubremetz and Nivre, 2015). We use the corpus from Dubremetz and Nivre (2015) as training corpus (used both to create and tune the features) and a new corpus as final test corpus. All the data come from Europarl (Koehn, 2005). The training corpus consists of 4 million words. The test corpus is a distinct extract of 2 million words. To test the generality of the approach, we will then apply

[2]For example: "**All** for **one**, **one** for **all**" is a true positive instance, "**All for** one, one **for all**" is a duplicate.

Feature	Description	Weight
#sameDepW_b W'_a	Number of incoming dependency types shared by W_b and W'_a.	$+5$
#sameDepW_a W_b	Same but for W_a and W'_b	$+5$
#sameDepW_a W'_a	Same but for W_a and W'_a	-5
#sameDepW_a W'_a	Same but for W_b and W'_b	-5

Table 1: Dependency features used to rank chiasmus candidates

the trained model also to a corpus of literary text: the Sherlock Holmes stories.

4.1 Implementation

Our program takes as input a text that is lemmatized, tagged and parsed using the Stanford CoreNLP tools (Manning et al., 2014). It outputs a list of sentences containing chiasmi candidates. The system provides two types of information about each candidate: the score given by the combination of features and the main words selected. The score is used to rank the sentences as in a search engine: highly relevant criss-cross patterns at the top, less relevant ones at the bottom. Thanks to the main words selection, a human annotator can see which words the system considered to constitute the criss-cross pattern in the chiasmus and determine whether the candidate is a true positive, a false positive, or a duplicate of a true positive (that is, an instance covering the same text as a true positive but with the wrong words matched). In the evaluation, duplicates are considered as false positives.

Model	Average Precision	Compared to Baseline
Baseline	42.54	NA
Tag features	59.48	+14
Negative dependency features	40.36	-2.2
Pos dep features	62.40	+20
All dependency features	64.27	+22
All features	67.65	+25

Table 2: Average precision for chiasmus detection (test set).

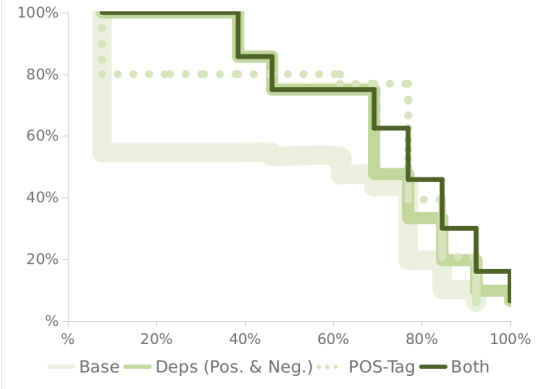

Figure 4: Interpolated precision-recall curve (test set).

4.2 Results and Analysis

To evaluate our features, we reproduce the experiment of Dubremetz and Nivre (2015) which uses only shallow features. Then we add our own features with the weights stated in Section 3. Following the idea of Clarke and Willett (1997, p.186), we annotate only the top 200 candidates in each experiment. We use two annotators for this task and base our evaluation only on the chiasmi that both annotators considered as true: we found 13 of them. We measured the inter-annotator agreement for the true/false classification task (counting duplicates as false) and obtained a kappa score of 0.69, which is usually considered as indicating good agreement.

Our table presents the average precision which is a standard measure in information retrieval (Croft et al., 2010). It averages over the precision scores at the ranks of the true positives.

In Table 2, we first of all see that tag features add 17% of average precision to the baseline, which shows that the simple idea of requiring tag identity for all words is a powerful way of eliminating false positives. When it comes to dependency features, negative features slightly damage the average precision when used alone (-2.2% compared to the baseline), while positive dependency features give nearly $+20\%$ average precision. However, negative features prove to be useful when combined with the positive features, and when combining both tag and dependency features, we improve by $+25\%$ compared to the baseline.

Combining tag and dependency features not only

improves average precision, but also improves recall compared to the baseline (as well as the system with only dependency features), because it retrieves the following chiasmus (originally ranked below 200):

(3) Do not imagine, however, that **legitimacy** in itself creates **democracy**. Rather, it is **democracy** which creates **legitimacy**.

As can be seen from the precision-recall curve in Figure 4, the combined system also has the most graceful degradation overall, even if it is surpassed by the pure dependency-based system in one region.

Our system definitely proves to be substantially better than the previous state of the art but it has its limits as well: first of all it needs a parsed input and parsing is time consuming. For 2 million words the Stanford CoreNLP takes days to give any output. Once parsed, our system needs 10 minutes per million words in order to output the result. Dependency features do not have the magic ability to get rid of all false positives (otherwise chiasmi like Example3 would be ranked 1 instead of 133 by dependency features). Moreover, syntactic features narrow the type of examples we get: some chiasmi are not based on perfect symmetry of roles and tags. For example:

(4) We must preach for **family values**, and **value families**.

Europarl is a convenient corpus for experimentation: it represents an almost endless source of clean text (more than 45 million words for just the English version), written in a consistent way. Literature is not as convenient: according to the Guiness Book of Records the longest novel ever written is about 1 million words long.[3] So far, our model has been trained on 4 million words and tested on 2 million words from the political discourse genre. We have successfully proven that a model tuned on one Europarl extract can generalise on another Europarl extract. Without any further tuning, can our detector find chiasmi in a different genre?

We chose to answer this by applying it to literary text. Our literature corpus is the complete anthology of Sherlock Holmes stories by Conan Doyle. We download the text file from the internet[4] and did not apply any kind of cleaning on it (thus, notes, chapter titles, and tables of content are still remaining). This gave us a corpus of about 650,000 words, to which we applied our baseline model and our final model. In Table 3, we see that the average precision is improved by +17% from the baseline to the final model. On a total of 8 chiasmi, the baseline finds 6 of within 200 candidates whereas our final model finds 7, which means that we improve not only precision but also recall. We can observe this performance on the recall-precision curve Figure 5.

Model	Average Precision	Diference
Baseline	53.00	NA
All features	70.35	+17

Table 3: Average precision for chiasmus detection (Sherlock Holmes set).

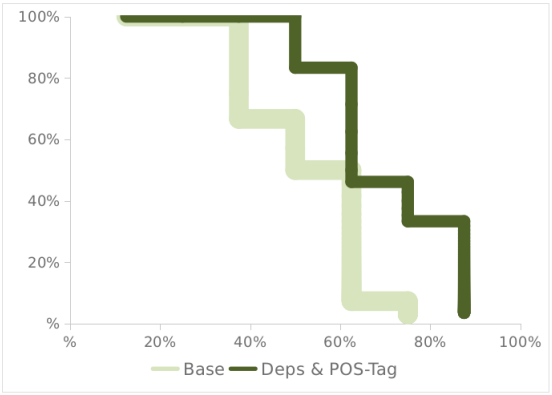

Figure 5: Interpolated precision-recall curve (literature set).

With so small numbers, we cannot be sure that the improvement is significant between the baseline and our system. However, the results show that running our model on a literary corpus can provide a significant help to the human user. Our algorithm with over 70% average precision managed to find 5 chiasmi within the top 10 candidates. This saves a considerable amount of human work, and we got this result without any special tuning or cleaning adapted to this genre.

5 Conclusion

The aim of this paper was to improve the performance of a chiasmus detector. The only existing system was based entirely on shallow features like words and punctuation. We have extended

[3] http://www.guinnessworldrecords.com

[4] https://sherlock-holm.es/stories/plain-text/cano.txt

that system with features capturing aspects of syntactic structure and discovered three effective features for chiasmus detection: tag features, positive dependency features and negative dependency features. Moreover, we have shown that the same model works well for literary text. An additional contribution of this paper is the annotation of two new corpora by two annotators. The first one is a Europarl corpus that includes 13 true positives on 466 instances. The second corpus is an anthology of Sherlock Holmes that includes 8 true positives on 399 instances.[5] By adding these to the corpus previously created by Dubremetz and Nivre (2015), we provide a data set that might be large enough to start exploring machine learning instead of tuning feature weights manually.

A Europarl Chiasmi

1. But if he were alive today, he would have said instead: "**East** is **West**, and **West** is **East**, and never the twain shall part."

2. I can therefore find no reason to differentiate between **Poland** and **Hungary** or between **Hungary** and **Poland**.

3. I should like to conclude by giving you some food for thought: Europe is good at converting **euros** into **research**, but often fails in converting **research** into **euros**, and that must change in future.

4. I think that Parliament is being held hostage to a few Stalinists, who always take a **strong** line with those who are **weak** and are **weak** in the face of those who are **strong**.

5. In turn, defence is constantly changing its boundaries in a world in which the perception of these is ever more blurred: nowadays, we cannot only consider the territorial defence of one State faced with a possible attack by another, but rather, as has been correctly said, we have **armies** that lack clear **enemies** and **enemies** that lack **armies**.

6. It is yet another example of the EU taking money from **poor** people in **rich** countries and giving it to **rich** people in **poor** countries.

7. Many of those areas have over the years turned from **land** into **sea** or from **sea** into **land**, with or without specific human intervention.

8. **Reason** without **passion** is sterile, **passion** without **reason** is heat.

9. We must avoid a situation where no answer is given because a society where **citizens** are afraid of their **institutions** - and perhaps more importantly **institutions** are afraid of their **citizens** - makes for a very weak democracy.

10. We want much greater enlargement, but without providing the corresponding funds and we invent lower and lower cohesion targets along the lines of "if the **mountain** won't come to **Mohammed**, then let's take **Mohammed** to the **mountain**".

11. What we now have to do, once we have consolidated the internal aspects of our project, is turn Europe into an international operator capable of comprehensive action with regard to the challenges facing the world, a world in which nations are too **big** to resolve their **small** problems and too **small** to resolve the **big** problems we are faced with on a global scale.

12. Women, men, workers, students, the unemployed, pacifists and ecologists will no longer be opposing the system but will be terrorists because - as Hegel, then an old man, wrongly said - 'the **real** is **rational** and the **rational real**' , and for our legislators nothing is more real than the present social and economic disorder and nothing is more irrational, and therefore terrorist, than the need to overthrow and eliminate it.

13. Do not imagine, however, that **legitimacy** in itself creates **democracy**. Rather, it is **democracy** which creates **legitimacy**.

B Sherlock Holmes Chiasmi

1. "After all, since we are to be on such terms, Mr. Altamont," said he, "I don't see why **I** should trust **you** any more than **you** trust **me**."

[5]The reader will find in appendix the list of all true positive chiasmi in both of our corpora.

2. “For years **I** have loved **her**. For years **she** has loved **me**.”

3. “I don’t think you need alarm yourself,” said I. “I have usually found that there was **method** in his **madness**.” “Some folks might say there was **madness** in his **method**,” muttered the Inspector.

4. “But the **Sikh** knows the **Englishman**, and the **Englishman** knows the **Sikh**.”

5. “He seems to have declared war on the **King**’s **English** as well as on the **English king**.”

6. “I can still remember your complete indifference as to whether the **sun** moved round the **earth** or the **earth** round the **sun**.”

7. “Insensibly one begins to twist **facts** to suit **theories**, instead of **theories** to suit **facts**.”

8. “He pays me well to **do** my **duty**, and my **duty** I’ll **do**.”

References

Sarah J. Clarke and Peter Willett. 1997. Estimating the recall performance of Web search engines. *Proceedings of Aslib*, 49(7):184–189.

Bruce Croft, Donald Metzler, and Trevor Strohman. 2010. *Search Engines: Information Retrieval in Practice: International Edition*, volume 54. Pearson Education.

Denis Diderot and Jean le Rond D’Alembert. 1782. *Encyclopédie méthodique: ou par ordre de matières, volume 66*. Panckoucke.

Marie Dubremetz and Joakim Nivre. 2015. Rhetorical Figure Detection: the Case of Chiasmus. In *Proceedings of the Fourth Workshop on Computational Linguistics for Literature*, pages 23–31, Denver, Colorado, USA, June. Association for Computational Linguistics.

Jonathan Dunn. 2013. What metaphor identification systems can tell us about metaphor-in-language. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 1–10, Atlanta, Georgia, June. Association for Computational Linguistics.

Bernard Dupriez. 2003. *Gradus, les procédés littéraires*. Union Générale d’Éditions 10/18.

Jakub J. Gawryjolek. 2009. *Automated Annotation and Visualization of Rhetorical Figures*. Master thesis, Universty of Waterloo.

Roland Greene, Stephen Cushman, Clare Cavanagh, Jahan Ramazani, and Paul Rouzer, editors. 2012. *The Princeton Encyclopedia of Poetry and Poetics: Fourth Edition*. Princeton University Press.

Daniel Devatman Hromada. 2011. Initial Experiments with Multilingual Extraction of Rhetoric Figures by means of PERL-compatible Regular Expressions. In *Proceedings of the Second Student Research Workshop associated with RANLP 2011*, pages 85–90, Hissar, Bulgaria.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *The Tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Henri Morier. 1961. *Dictionnaire de poétique et de rhétorique*. Presses Universitaires de France.

Helge Nordahl. 1971. Variantes chiasmiques. Essai de description formelle. *Revue Romane*, 6:219–232.

Alain Rabatel. 2008. Points de vue en confrontation dans les antimétaboles PLUS et MOINS. *Langue française*, 160(4):21–36.

Susan Schreibman, Ray Siemens, and John Unsworth. 2008. *A Companion to Digital Humanities*. John Wiley & Sons, April.