Syntax Matters for Rhetorical Structure: The Case of Chiasmus

Marie Dubremetz, Joakim Nivre

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

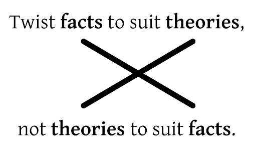

Figure 1: Schema of a chiasmus

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?

2. Experiments

The repetition of words is an extremely common phenomenon. Defining a figure of speech by just the position of word repetitions is not enough (Gawryjolek, 2009; Dubremetz, 2013). To become a real rhetorical device, the repetition of words must be "a use of language that creates a literary effect".[1] This element of the definition requires us to distinguish between the false positives, or accidental inversions of words, and the (true) chiasmi, that is, when the inversion of words explicitly provokes a figure of speech. Sentence (2) is an example of false positive (here with 'the' and 'application'). It contrasts with Sentence (1) which is a true positive.

(1) Chuck **Norris** does not fear **death**, **death** fears Chuck **Norris**.

(2) My government respects the **application** of **the** European directive and **the application** of the 35-hour law.

We want rank the chiasmi by how likely they are to create a literary effect. Through a linear model applied on the Europarl corpus, we tune features and weights in order to give a score to chiasmi. This score allows to rank chiasmi. At the end, if the features are well selected and tuned we must observe that the computer systematically ranks higher chiasmus like 1 and ranks lower the non rhetorical cases like 2.

This paper presents the first attempt to go beyond shallow surface features in order to detect rhetorical 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.

We use the corpus from Dubremetz and Nivre (2015) as our training corpus (used to learn weights for a fixed set of features) and a new corpus as our final test corpus. The training corpus consists of four million words from the Europarl corpus, containing about two million instances of criss-cross patterns.

3. Results

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

[1] Definition of 'rhetorical device' given by Princeton wordnet: https://wordnet.princeton.edu/

$$\underbrace{\text{In prehistoric times}}_{C_{\text{Left}}}\ \underbrace{\textbf{women}}_{W_a}\ \underbrace{\text{resembled}}_{C_{ab}}\ \underbrace{\textbf{men}}_{W_b}\ \underbrace{\text{, and}}_{C_{bb}}\ \underbrace{\textbf{men}}_{W_b'}\ \underbrace{\text{resembled}}_{C_{ba}}\ \underbrace{\textbf{women}}_{W_a'}.$$

Figure 2: Schematic representation of chiasmus, C stands for context, W for word.

Feature	**Description**	**Weight**
sameTag	True if W_a W_b W_b' W_a' words have same PoS-Tag.	10
#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_b W_b'	Same but for W_b and W_b'	-5

Table 1: Dependency features used to rank chiasmus candidates

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).

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

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

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.

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. 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

4. 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 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. 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.

References

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. Association for Computational Linguistics.

Marie Dubremetz and Joakim Nivre. 2016. Syntax Matters for Rhetorical Structure: The Case of Chiasmus. In *Proceedings of the Fifth Workshop on Computational Linguistics for Literature*, pages pages 47–53, San Diego, California,USA. Association for Computational Linguistics.

Marie Dubremetz. 2013. Vers une identification automatique du chiasme de mots. In *Actes de la 15e Rencontres des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RECITAL'2013)*, pages 150–163, Les Sables d'Olonne, France.

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

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