Can Sentiment Lexicon Help Identifying Parallelism?

A Machine Learning Approach to Identify Textual Parallel Structure (Inter-Sentence Level)

Seminar title: 02-25-2061-pj Projekt Korpus- und Computerlinguistik
Instructor: Dr. Sabine Bartsch **Semester:** WS 2019-20

TECHNISCHE UNIVERSITÄT DARMSTADT

Vorname, Nachname: Jingying, Wang

Matrikelnummer: 2797667

Studiengang: Linguistic and Literary Computing

Fach / Fächer: Sprach- und Literaturwissenschaft

Studiensemester: WiSe 2019/20

Date of submission: 31.03.2020

Table of contents

1 Introduction 1

2 Related Work 2

2.1 Semantic Representation in Parallelism Identification Systems 2

2.2 Rationale of Using VAD Sentiment Lexicon for Parallelism Identification 2

3 Dataset 3

3.1 Corpus Design and Pre-processing 3

3.2 Manual Annotation 4

3.2.1 Annotation Guidelines 4

3.2.2 Disagreement in Manual Annotation 6

4 Feature Engineering 7

4.1 Lexical Features 8

4.2 Syntactic Features 9

4.3 Semantic Features 10

4.3.1 Processing NRC-VAD-Lexicon 10

4.3.2 Semantic Representation and Parameter Selection 10

5 Results 13

5.1 Performance of the Classifier 14

5.2 Can Sentiment Lexicon Help Identifying Parallelism? 15

5.3 Error Analysis 17

5.3.1 False Positives 17

5.3.2 False Negatives 18

6 Conclusion and Discussion 18

Table of Figures / Abbildungsverzeichnis

Fig. 1: Relation Between the Threshold (for Arousal) and the Proportion of Parallel Sentence-pairs 12

Fig. 2: Relation Between the Threshold (for Dominance) and the Proportion of Parallel Sentence-pairs 12

Fig. 3: Relation Between abs. Difference and the Proportion of Parallel Sentence-pairs (for Valence) 13

Fig. 4: Relation Between abs. Difference and the Proportion of Parallel Sentence-pairs (for Dominance) 13

Tab. 1: Evaluation for class “T” (Best Results of all ML Algorithms) 15

Fig. 5: Most Informative Features (Naive Bayes Classifier) 15

Fig. 6: Emotional Dynamics of Arousal- and Dominance-Dimension (for 100 sentences) 16

Tab. 2: Evaluation for class “T” (Influence of Using Semantic Features) 16

1 Introduction

Parallelism, as an important rhetoric device and a special textual structure across different languages, has been studied for a long time and from different perspectives. In recent years, applying supervised/unsupervised machine learning methods to identify parallel structure automatically has been gaining momentum, especially in Chinese academic community. The value of using parallelism identifier for automatic text evaluation (穆婉青 et at., 2018: 145), sentiment analysis (Dai et at., 2018: 148) and automatic writing (熊李艳 et al., 2018: 1751) etc. has also been recognized. Previously, the author of this article has also conducted a project[1] on applying machine learning methods for identifying parallel structure, and the focus lies on the inner-sentence level parallelism. This project can be seen as a sister project of the previous one, but with the focus on inter-sentence level parallelism, i.e. parallel structure across sentences. The parallelism identifier developed in the previous study is based mainly on lexical and syntactic features and achieved a good result of 0.851 (F1-score). Nevertheless, semantic information of parallel structure is also important and worth further research. Therefore, apart from lexical and syntactic features, the semantic features, more specifically, sentiment features of parallelism and their relevance for classifier training is also analysed in this project. Another difference between these two projects is that while the definition of parallelism for the previous one is inclined to grammatical parallelism, this project takes more rhetorical aspects of parallelism into consideration.

The overall workflow of this project resembles the previous one which includes manual annotation (establishing gold standard), feature engineering, classifier training, performance evaluation as well as error analysis. In the following section, the semantic representation strategies in other similar work is discussed and the rationale of using sentiment lexicon for extracting semantic features is argued. How the dataset is established and the problems encountered in manual annotation process will be explained in section 3. Some insights about the difference and relationship between inner-sentence and inter-sentence parallelism are also drawn from section 3. How lexical, syntactic and semantic features are extracted and selected is demonstrated in section 4. In section 5, the performance of different identification systems and the influence of semantic features are evaluated. Error analysis is also performed in this section.

[1] https://github.com/JYWangGeneCosmo/linguistic-parallelism-identification

2 Related Work

2.1 Semantic Representation in Parallelism Identification Systems

In recent years, several parallelism identification systems have been proposed. As syntactic similarity is one of the most important features of parallelism and compared with semantic features, is relatively easier to be represented based on PoS-tags, most of the parallelism identification systems - rule-based or supervised/unsupervised machine learning based - have taken syntactic information of sentences into consideration. (Guégan and Hernandez, 2006; 梁社会 et al., 2013; Song et at., 2016; 穆婉青 et at., 2018; 熊李艳 et al., 2018). Some of these systems also make use of the semantic information with different approaches: Guégan and Hernandez (2006) have applied *WordNet* to normalize sentences as a pre-processing procedure, but the performance of this system is not quite satisfying[2]; Classification systems proposed by 穆婉青 et at. (2018) as well as Dai et at. (2018) are both based on neural network. The former is based on convolutional neural network (CNN) and has employed Word2Vec to get word embeddings as semantic representation. The latter is based on recurrent neural network (RNN) and used a bidirectional LSTM as an encoder to get semantic vector representation. These two systems have both achieved satisfying results[3]. The unsupervised machine learning methods does not require artificial designed features and can avoid adverse effects on improper selection of features. (Dai et at., 2018: 148). Although this kind of approach can overcome some disadvantages of manual feature extraction, it is still unknown to us human that which semantic information is taken account for making the classification decision. Word2Vec is also employed in Song et al. (2016), but unlike the 2 projects mentioned before, the word vectors are not used with neural network methods, instead, the cosine similarity between the vectors of two words are computed to derive features for machine learning methods. These previous studies suggest that semantic information is important for detecting parallelism, nevertheless, the concept of semantic information is quite general. In this project, the emotional/sentimental aspects of semantic information for parallelism identification is analysed from a more linguistic perspective.

2.2 Rationale of Using VAD Sentiment Lexicon for Parallelism Identification

There are different models for explaining the notion of emotion. Two influential models are basic emotions model (Plutchik, 1980; Russell & Barrett, 1999) and the VAD (valence(pleasure), arousal, and

[2] Best overall F-measure is 54%.
[3] The highest F1-scores achieved in these two projects are both above 90%.

dominance) model (Bradley & Lang, 1999; Bakker et al., 2014). The second model is more suitable for analysing the semantic features, more specifically, sentimental features of parallelism, as many studies investigating the rhetorical or functional effects of parallel structure have denoted the relationship between parallelism and the dominance/arousal dimensions of emotion in an explicit or implicit way. Some studies about the parallelism in political speeches have shown that this structure has persuasive properties and is employed to reinforce certain ideology or explicate important information (Al-Ameedi, 2017: 195; Almehmdawi, 2018: 280-283). This feature can also be found in parallel sentences from ancient Chinese books such as *Mencius* (Dai et at., 2018: 148). And According to (Bauman & Briggs, 2003: 113), the repetition with variation that characterizes parallelism imparts rhetorical emphasis and closure to the propositional content of the text. Therefore, the words used in the parallel structure can be influential and of high importance. This feature actually corresponds with the description of high dominance in many sentiment lexicons based on VAD model (Bradley & Lang, 1999: 2; Mohammad, 2018: 4) - High dominance is described with words such as influential, important or powerful etc. The arousal dimension is also related to parallelism to some extent. According to Almehmdawi (2018: 280), parallelism is often used to motivate the audiences in political speeches and an experimental study conducted by Menninghaus et al. (2017: 55) shows the parallel structure serves as intensifiers of emotional impact in poetry. It is clear that this *structure* usually contains high degree of arousal[4] and this high arousal might also be represented on the lexical level.

As the semantic information of words with respect to dominance and arousal is represented with fine-grained numerical values in VAD sentiment lexicons, this kind of lexicon is advantageous for deriving features of parallel sentences for classifier training with machine learning methods. In this project, the recently published NRC-VAD Lexicon[5] is employed, as it contains a large number of entries (up to 20,000 English words) and thanks to Best-Worst Scaling (BWS) method that addresses the limitations of traditional rating scales, the ratings are more reliable than those in existing lexicons (Mohammad, 2018: 1).

3 Dataset

3.1 Corpus Design and Pre-processing

Apart from the main goal of developing an inter-sentence level parallelism identifier and analysing the sentiment features of the parallel sentences, this project is also designed to get some insights about

[4] High arousal is described with words such as arousal, activeness, stimulation etc. in Bradley & Lang 1999: 2 and Mohammad, 2018: 4.

[5] https://saifmohammad.com/WebPages/nrc-vad.html

the difference between inner-sentence parallelism and inter-sentence parallelism. Therefore, the same three speeches from the American Inaugural Speech Corpus as for the previous project[6], which focuses on inner-sentence parallelism, are chosen together with the speech from 1993 (as expansion) to establish the training data.

In alignment with the previous project, similar pre-processing methods are applied, i.e. texts are PoS-tagged (Format: *WORD_TAG*) with Stanford POS Tagger (Version: 3.9.2 2018-10-16) and the segmentation is based on dot (full stop), exclamation mark, question mark, colon and semicolon (. ! ? : ;). It should be noted that since the focus of this project is inter-sentence parallelism, the dataset is structured in a different way - Every 2 successive sentences are combined together as a pair for manual annotation with the format: <sent1, sent2, tag>. In the following text, the first sentence in a sentence pair is referred to as sent1 and the following sentence is referred to as sent2.

3.2 Manual Annotation

As the two annotators of this project have also performed the annotation work for the previous parallelism project, the communication between them is relatively easier and more efficient this time. The workflow of this task follows the *MAMA* Cycle (Model Annotate Model Annotate) (Pustejovsky & Stubbs, 2012: 28): 1. The initial tagset and guidelines are established by the first annotator. 2. Both annotators annotate the first speech independently. 3. Both annotators discuss about the disagreement and remodel the guidelines together. 4. Repeat step 2 and 3 for the second speech. 5. Both annotators annotate the rest 2 speeches independently according to the final version of guidelines. The last 2 independently annotated speeches are used for evaluating the inter-annotator agreement based on Cohen's kappa statistic. The kappa score is approx. 0.81 which indicates a high level of agreement.

3.2.1 Annotation Guidelines

Compared with the annotation guidelines for inner-sentence parallelism, the guidelines for this project pay more attention to the rhetoric aspects of parallel structures.

Guidelines (simplified version)[7]:

I. **General type of Parallelism:** Sentence-pairs which have similar or identical grammatical structures (tolerance of length difference: 4 tokens) are tagged as "T" (Similar with inner-sentence parallelism), e.g. *sent1:* The Capital was abandoned. *sent2:* The enemy was advancing.
II. **Special types of parallelisms (with respect to rhetoric function)** are tagged as "**T**"**:**

[6] https://github.com/JYWangGeneCosmo/linguistic-parallelism-identification
[7] For full version see *Guidelines_inter_sent_parallelism.pdf*

1). Anaphora: Repetition of at least one word at the ***beginning*** of successive sentences/clauses.

e.g.: sent1: **From this day forward,** a new vision will govern our land.

sent2: **From this day forward,** it's going to be only America first. (2017, Donald J. Trump)

2). Epistrophe: Repetition of at least one word at the ***end*** of successive sentences/clauses.

e.g.: sent1: Our challenges **may be new.**

sent1: The instruments with which we meet them **may be new.** (2009, Barack Obama)

3). Anadiplosis: Rhetoric repetition of the words or phrase at the end of one sentence/clause, at the beginning of the next.

e.g.: sent1: We are not this story's author, who fills time and eternity with **his purpose**.

sent2: Yet, **his purpose** is achieved in **our duty**. (2001, George W. Bush)

Note:

- Words at the "Beginning" does not have to occur at the first position. Some particles or conjunctions like "yes", "and" etc. or appositives which does not disrupt the parallel structure as a whole, can be ignored. e.g. sent1: **This is your** celebration. sent2: And **this**, *the United States of America,* **is your** country.
- When only one word is repeated at the "Beginning": If this word is just a functional word (no rhetoric feature) and the sentence structure differs too much, then the sentence pair should not be considered as containing **Anaphora**.

III. Sentences with no parallel structure are tagged as "**F**".

3.1.2 Inter-sentence parallelism vs. Inner-sentence parallelism

There are some phenomena about the difference and the relation between these two types of parallelisms found during the annotation process:

1) Proportion of inter-sentence parallelism is lower than inner-sentence parallelism: In the previous sister project, 123 out of 380 (32.37%) sentences are annotated as containing parallelism. In comparison, the proportion of sentence-pairs with parallel structure is significantly lower. This is also the reason why the annotation corpus is expanded with one more speech - the sample number for "T" class is too small. After the expansion, altogether 101 out of 448 (22.5%) sentence pairs are annotated as "T".

2) Inter-sentence parallelism is constructed based more on lexical repetition and for rhetoric function: Most parallel sentence-pairs found in the corpus contain anaphora (repetition of words at the sentence-initial part), while the syntactic similarity between them is not prominent. On the other hand, the parallel structures found within a sentence are often grammatical parallelism.

3) Inter-sentence parallelism and inner-sentence parallelism interplay with each other: It is found that in some speeches with high quality, these two types of parallelisms are employed in a combined way.

Example 1: Inner-sentence parallel structure in two sentences are also parallel with each other:

sent1: We will restore science to its rightful place and wield technology's wonders ***to raise health care's quality*** and ***lower its cost***.

sent2: We will harness the sun and the winds and the soil ***to fuel our cars*** and ***run our factories***.

(Obama, 2009)

Example 2: Inter-sentence parallel structure is *partially* parallel with the inner-sentence parallelism:

sent1: We remain a young nation, but in the words of Scripture, **the time has come to set aside childish things**.

sent2: **The time has come *to reaffirm our enduring spirit*, *to choose our better history*, *to carry forward that precious gift***, that noble idea passed on from generation to generation … (Obama, 2009)

3.2.2 Disagreement in Manual Annotation

Although with the knowledge based on previous sister project, the manual annotation is done more efficiently this time, there are still some problems which are relevant for inter-sentence parallelisms and caused alteration in guidelines. The disagreement/annotation problems usually lie in the following 4 situations:

1) Defining Anaphora[8]

Although anaphora is usually defined as the deliberate repetition of the first part of successive sentences in order to achieve a rhetoric effect. This "deliberate repetition" and "rhetoric effect" can sometimes interpreted differently. Take "we will" this phrase as an example, although it is frequently used in political speeches to form a parallel structure so as to evoke the audience's emotions, it can also be used successively in 2 nonparallel sentences that are barely similar in their syntactic structure. e.g.:

sent1: **We will** begin to responsibly leave Iraq to its people and forge a hard-earned peace in Afghanistan. With old friends and former foes, **we will** work tirelessly to lessen the nuclear threat and roll back the specter of a warming planet.

sent2: **We will** not apologize for our way of life, ***nor* will we** waver in its defense. (Obama, 2009)

In this excerpt, "we will" is actually used to form parallelism, however, it concerns only inner-sentence parallel structure. These two sentences are not parallel with each other. Thus, this type of sentence pairs is not considered as containing inter-sentence parallelism.

2) Defining Anadiplosis

Problems in detecting anadiplosis is actually related to another fundamental question: How to define *word*? In the sentence pair shown below, although "drifted" and "drifting" are not the same on the surface, they have exactly the same lemma form. And the repetition pattern is close to anadiplosis.

[8] https://www.thefreedictionary.com/anaphora

However, considering that this situation is quite rare in the corpus, and using lemmatizer to treat this situation could result in many false positives, we decided not to assign label “T” to it.

sent1: Instead, we have **drifted**.

sent2: And that **drifting** has eroded our resources, fractured our economy, and shaken our confidence.

(Clinton, 1993)

3) The parallel structure lies partially in two sentences

At the beginning of the annotation process, we only identify the lexical repetition at the initial or final part of two sentences to detect anaphora, epistrophe and anadiplosis. However, in this way the parallel structure lies in between will be ignored, e.g.:

sent1: We remain a young nation, but in the words of Scripture, **the time has come to set aside childish things.**

sent2: **The time has come to reaffirm our enduring spirit,** to choose our better history, to carry forward that precious gift, that noble idea passed on from generation to generation: the God-given promise that all are equal, all are free, and all deserve a chance to pursue their full measure of happiness.

(Obama, 2009)

Although this parallelism concerns only part of each sentence, this parallel structure does exist across two sentences and therefore is considered as inter-sentence parallelism.

4). The last sentence can be seen as part of the parallelism block, but is not parallel with the previous sentence.

In the following example, the first two sentences are undoubtedly parallel with each other, and according to the usage of “semicolon”, the third sentence should also belong to the same block, but the third sentence no longer maintains the same structure and functions more like a further definition for the previously mentioned “citizens”. After discussion, we decided the treatment for this situation should be as follows: If the sentence structure differs too much and/or less than two words reoccurred in the following sentence, assign “F”.

(I ask you to be citizens:) **Citizens, not** spectators;

citizens, not subjects;

responsible **citizens** building communities of service and a nation of character.

(Bush, 2001)

4 Feature Engineering

4.1 Lexical Features

As is mentioned in section 3.1.2, lexical repetition is a prominent feature of inter-sentence parallelism. This feature can be extracted relatively easily, nevertheless, according to the previous study[9], the position in which this repetition occurs is crucial for avoiding false positive results. Thus, for anaphora, epistrophe and anadiplosis these 3 special types of parallelism which are characterized by lexical repetition, the repetition position is thoroughly analysed.

- For epistrophe and anadiplosis: These two types of parallelism both concern the words or phrase at the end of sent1. These words are repeated respectively at the end of sent2 (**epistrophe**) and at the beginning of sent2 (**anadiplosis**). Based on observation, the repeated phrases usually contain at least 2 and no more than 4 tokens and a sentence usually contains at least 4 tokens (including punctuation). Therefore, the sentence-initial/-final part, where the lexical repetition should be extracted, is defined as the first and the last 4 tokens of a sentence.

Feature 1a (binary): At least one bigram at the sentence-final part of sent1 is repeated at the sentence-initial or sentence-final part of sent2.

- **For anaphora:** the feature extraction for anaphora is relatively tricky, due to two main reasons:

1). This type of parallelism often contains particles, conjunctions or appositives that separate two parallel parts, e.g.:

> Sent1: We will make America safe again.
>
> Sent2: ***And, yes, together,*** we will make America great again.

2). Sent1 is partially parallel with sent2. This situation is usually due to the segmentation strategy (semicolon as sentence-closer) applied in the project, e.g.:

> We rededicate ourselves to the very idea of America, **an idea** born in revolution and renewed through two centuries of challenge**;**
>
> **an idea** tempered by the knowledge that, but for fate, we, the fortunate, and the unfortunate might have been each other; **an idea** ennobled by the faith that our Nation can summon from its myriad diversity the deepest measure of unity; **an idea**…

To deal with this problem, 2 *while* loops are employed. In each iteration, after checking bigram-repetition at the specific position, the word/phrases together with the comma at the beginning of a *sentence* will be cut off to form a new one and the new *sentence* will be used to continue searching the parallel structure. The pseudocode shows how this strategy is implemented (this strategy is applied to both sent1 and sent2):

```
num_of_repeated_Bigram = 0    # counter
while len(sent1) > 3:         # The sentence-initial/-final part should consist of at least 4 words
```

[9] https://github.com/JYWangGeneCosmo/linguistic-parallelism-identification (see p. 7 of the paper)

```
bi_initialS1 = get_Bigrams_at_the_beginning_of_sent1
bi_initialS2 = get_Bigrams_at_the_beginning_of_sent2
for bigram in bi_initialS1:
  if bigram is contained in bi_initialS2:
    num_of_repeated_Bigram += 1
if no more comma is contained in sent1 or repetition already found:
  break
# Tackle the situation when sent1 is partially parallel with sent2
else if sent1 contains comma:
 sent1 = the sub-sentence starts after the comma
   # go back to the beginning of the iteration
```

Feature 1b (binary): At least one bigram-repetition is recognized by the strategy above.

Since ***Feature 1a*** and ***Feature 1b*** are both lexical features, they are combined together as one.

4.2 Syntactic Features

The extraction of syntactic features can take advantage of the methods used for identifying inner-sentence parallelism, i.e. Normalization of PoS-tags (only the first two characters of the PoS-tag are used: NNS=>NN); use Longest Common Subsequence (LCS) algorithm to compute the similarity between 2 PoS-tag sequences. These methods are suitable for 2 successive sentences that are parallel as a whole (general type of parallelism), however, as mentioned in section 4.1, sent1 can be only partially parallel with sent2. Therefore, these methods need proper adaptation to inter-sentence parallelism.

Feature 2a(binary): PoS-tag sequence of sent1 is similar with PoS-tag sequence of sent2 (sent1 and sent2 are parallel as a whole).

It should be noted that, although the partial parallelism across two sentences resembles inner-sentence parallelism, merging 2 sentences as one and applying the same method for inner-sentence parallelism could result in extremely low performance. This is because in this project sentence that contains parallel structure, but is not parallel with the adjacent sentence, is not tagged as “T” and this extraction approach can not differentiate inner-sentence from inter-sentence parallel structure. It continuously compares the similarity of the PoS-tag sequences of two successive sub-sentences, so even when the similarity is recognized for the sub-sentences that are both belong to the same sentence, the result is still true. This can introduce many noises. Thus, only the last sub-sentence of sent1 is compared with the first sub-sentence of sent2.

Feature 2b(binary): The PoS-tag sequence of the last sub-sentence of sent1 is similar with that of the first sub-sentence of sent2 (sent1 is partially parallel with sent2).

Feature 2a and ***Feature 2b*** are combined together as the syntactic feature.

4.3 Semantic Features

4.3.1 Processing NRC-VAD-Lexicon

As mentioned in the previous section, NRC-VAD-Lexicon is used to extracted semantic features of sentences. NRC-VAD-Lexicon is a txt file in which words with their values for 3 emotional dimensions are stored in the format <Word Valence Arousal Dominance>. The scores range from 0 (lowest V/A/D) to 1 (highest V/A/D). To access the score for the corresponding word, the data in this file is pre-processed and restructured in a more manageable format.

It should be noticed that the format of the sentiment lexicon is in fact not strictly followed, which makes the token number of each row not always exactly 4. There are mainly two types of non-standard entries:

1) Additional information is added to suggest the sentiment of the context:

['extremely', ***'negative'***, '0.030', '0.786', '0.311']

['extremely', ***'positive'***, '0.993', '0.730', '0.839']

2) Compound words or phrases that usually used as a unit (collocates), are split into several separate tokens:

['business', 'man', '0.530', '0.598', '0.933']

['breaking', 'and', 'entering', '0.418', '0.729', ‘0.500’]

The treatment of the collocates is not always consistent, sometimes a phrase consists of several words are combined as one, e.g.: ['leavemealone', '0.240', '0.472', ‘0.402’]. Considering that the proportion of these non-standard entries are small and it is difficult to identify the related words. Therefore, they are not used for feature extraction. After the removal, altogether 19874 entries are restructured into three dictionaries (python datatype) respectively for valence dimension, arousal dimension and dominance dimension in which words are stored as keys.

4.3.2 Semantic Representation and Parameter Selection

In this project, the average arousal score and dominance score of a sentence are both computed for semantic representation based on the simple formula:$\frac{\sum ScoreOfallWords}{numOfWords}$. As the words in NRC-VAD-Lexicon are stored in their lemma form and in lowercase, the words in sentences need to be lemmatized and lowercased to access the corresponding A/D scores. For lemmatization task, an

advanced NLP tool *spaCy*[10] is employed. For lemmas that have no entry in the lexicon, their A/D scores are considered as 0.

As discussed in section 2.2, high arousal and high dominance could be informative features for identifying parallelism. These two semantic features are represented as follows:

Feature 3 (binary): The average arousal scores of two sentences are both higher than a certain threshold.

Feature 4 (binary): The average dominance scores of two sentences are both higher than a certain threshold.

In order to find the proper threshold for feature extraction without trying out a large number of decimal numbers with no orientation, the relationship between A/D scores and parallelism are analysed at first. For better visualization, this relationship is plotted with the python module *matplotlib.pyplot*[11]as follows. The X-axis stands for the threshold that increases linearly from 0 to 0.6. The Y-axis stands for the proportion of parallel sentence-pairs among all sentence-pairs under the same condition (the average A/D scores of two successive sentences are both higher than the threshold). As almost all sentences have an average A/D score of more than 0, the starting point of the proportion value should correspond with the proportion of parallel sentence-pairs within the whole dataset (22.5%). So the curves in Fig. 1 and Fig. 2 both start from slightly above 0.2. It is striking that for both arousal and dominance, overall speaking, the proportion increases rapidly as the threshold grows. When the threshold reaches a certain value (for arousal is 0.33 and for dominance is 0.39), the sentence-pairs found all contain parallel structure (the proportion is 100%). This corresponds with the analysis in section 2.2 and suggests that high dominance and high arousal should be suitable as semantic features for training a classifier. According to Fig. 1, the threshold for arousal should be at least 0.28, since it is the point where the proportion of parallel sentence-pairs is higher than that of the non-parallel sentence-pairs (higher than 50%). According to Fig. 2, the threshold for dominance should be at least 0.34. In addition, considering the sample number for sentence-pairs that have extremely high avg. A/D scores is not big enough[12], the thresholds for arousal and dominance should not exceed respectively 0.38 and 0.55.

Semantic similarity is also an important feature of parallel structure. And according to Mohammad (2018: 1), valence, arousal, and dominance (VAD) are the three primary dimensions of meaning. Thus, the overall semantic similarity might also be represented in these 3 dimensions. To represent the semantic similarity between two sentences, the average V/A/D scores are used as follows:

---

[10] https://spacy.io/models/en

[11] https://matplotlib.org/api/pyplot_api.html

[12] The proportion declined to 0 when the threshold reaches a certain value, because no two successive sentences in the dataset both have such high average A/D scores.

Fig. 1 Relation Between the Threshold (for Arousal) and the Proportion of Parallel Sentence-pairs

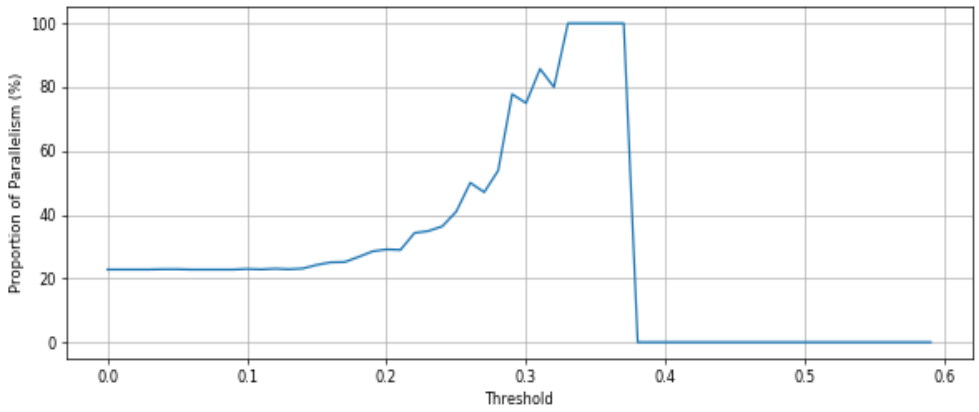
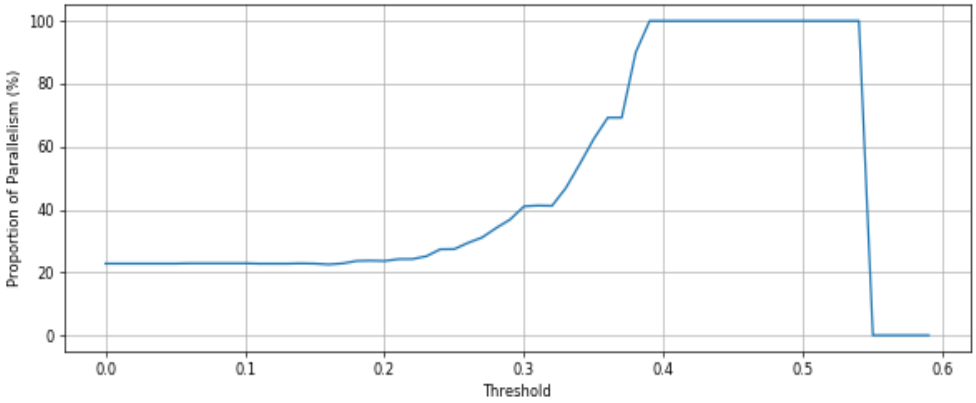


Fig. 2: Relation Between the Threshold (for Dominance) and the Proportion of Parallel Sentence-pairs



Feature 5 (binary): The absolute difference between average valence scores of two successive sentences is lower than a certain threshold.

Feature 6 (binary) and Feature 7 (binary): The extraction methods respectively for arousal similarity (Feature 6) and dominance similarity (Feature 7) are the same with Feature 5.

To select the best thresholds, similar visualization methods are applied. The relationships between the threshold for absolute Valence/Dominance difference and the proportion of parallel sentence-pairs are plotted as follows. The figure for arousal is similar with the figure for dominance and is therefore not shown here.[13] The X-axis stands for the absolute difference that increases linearly from 0 to 0.3. The Y-axis stands for the proportion of parallel sentence-pairs among all sentence-pairs under the same condition (the absolute difference between avg. V/A/D scores of two successive sentences is lower than the threshold). Since no two successive sentences have exactly the same avg. V/A/D scores,

[13] The figure is plotted in Project_Parallelism_CL_JingyingWang.ipynb

the curves in Fig. 3 and Fig. 4 both start from 0. As is shown in these two figures, unlike the threshold for high arousal/dominance, the thresholds for determining semantic similarity in V/A/D these three dimensions can barely influence the proportion of parallel sentence-pairs. For valence dimension, the proportion value fluctuates between 23% and 37%, for dominance and arousal dimensions, the value fluctuates slightly between 18% and 26%. However, no matter what threshold in is used for determining similarity, the proportion of non-parallel sentence-pairs is always much higher than that of parallel pairs, which suggests the similarity in these 3 dimensions is not prominent feature for inter-sentence parallelism. Thus, these 3 features with regard to semantic similarity are not chosen for training the classifier.

Fig. 3 Relation Between abs. Difference and the Proportion of Parallel Sentence-pairs (for Valence)

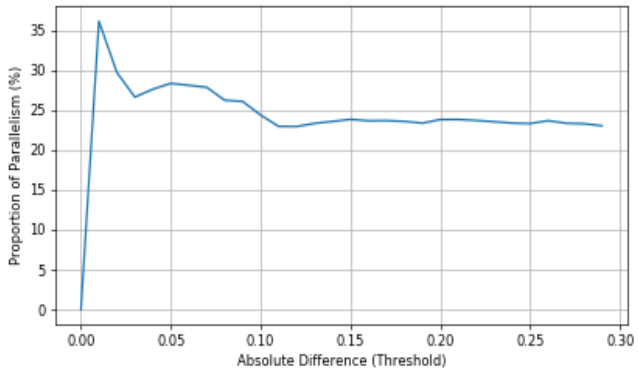
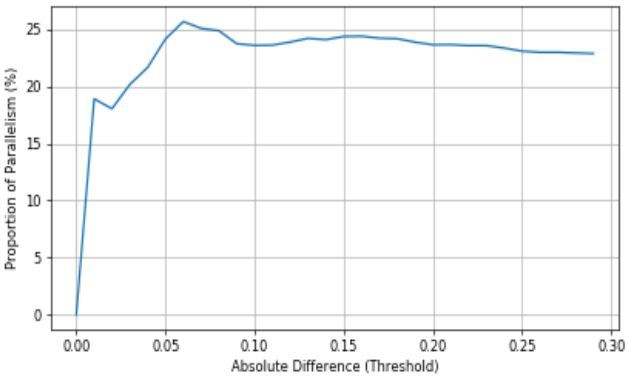


Fig. 4 Relation Between abs. Difference and the Proportion of Parallel Sentence-pairs (for Dominance)



5 Results

After the feature engineering process, the selected 4 features, namely lexical repetition in certain positions (Feature 1), syntactic similarity (Feature 2), high arousal (Feature 3) and high dominance (Feature 4) are used in different combinations with different machine learning algorithms. As the dataset is relatively small and “T” class is quite infrequent, cross-validation with 5 folds is performed for 3 standard metrics, namely precision, recall and F1-score, in order to tackle scarce data problem or selection bias and evaluate the performance of the classifier in a more reliable way. The baseline F1-score for “T” class is computed based on the assumption that all sentence-pairs are assigned with “T”, which is 0.368. Since the dataset is imbalanced (ratio between “F” and “T” is 4.4 : 1) and the focus of this project is the “T” class (parallelism), accuracy concerning both classes is not meaningful for evaluation and is therefore not computed.

5.1 Performance of the Classifier

Altogether 4 machine learning algorithms from 2 toolkits (*nltk* and *scikit-learn*) are employed. The feature combination that achieves the best performance is different for different algorithms. Final settings for different algorithms are as follows:

Naive Bayes Algorithm: The combination of all 4 features can achieve the best results. The threshold parameter for *Feature 2a* (sent1 and sent2 are similar as a whole) is 0.74 and the threshold for *Feature 2b* (sent1 is partially similar with sent2) is 0.79[14]. The former parameter is lower than the latter one, because it deals with two whole sentences and sentence is a larger unit than phrases/clauses, so the length difference is usually larger and leads to a relatively lower similarity. The thresholds used for determining high arousal and high dominance are respectively 0.28 and 0.37.

Algorithms from Scikit-learn: To find the best feature combination, the module *SelectKBest*[15] is employed together with *chi2*[16] module from *scikit-learn*. The *chi2* module performs chi-square test to filter out the features that are most likely to be independent of the labels and the results can be used by *SelectKBest* module to select the *k* best features. 2(*k*) most contributive features selected by this approach are the lexical feature and the syntactic feature. For all three machine learning algorithms (K-nearest neighbor, Logistic regression and Linear Linear Support Vector) from scikit-learn, the combination of these 2 can achieve the best performance. The influence of using semantic features will be demonstrated in section 5.2.

The cross-validated precision, recall and F1-scores for different classifiers with the settings mentioned above are shown in the following table. According Tab. 1, the F1-scores of all classifiers are above 87%, which suggest good prediction performance and are significantly higher than the baseline. KNN classifier and LinearSVC classifier have exactly the same performance. The results of them are also the best among all classifiers with respect to precision and F1-score. The recall of Naive Bayes Classifier is the highest and its precision and F1-score are also close to the best results. Therefore, to detect as many parallel sentence-pairs as possible, Naive Bayes system should be the best choice. It is worth noting that, in the previous project for inner-sentence parallelism identification, the Naive Bayes system also has the best recall.[17] In comparison, although the precision of the logistic regression classifier is extremely close to the highest result, its recall is considerably lower than other systems.

---

[14] This parameter is taken from the sister project for inner-sentence parallelism identification.

[15] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html

[16] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.chi2.html#sklearn.feature_selection.chi2

[17] https://github.com/JYWangGeneCosmo/linguistic-parallelism-identification (see p. 8 of the paper)

Tab. 1 Evaluation for class "T" (Best Results of all ML Algorithms)

Algorithms	P	R	F
NaiveBayesClassifier (nltk)	0.8876	**0.893**	0.8862
KNeighborsClassifier (sklearn)	**0.8978**	0.8877	**0.8902**
LogisticRegression (sklearn)	0.8967	0.8655	0.877
LinearSVC (sklearn)	**0.8978**	0.8877	**0.8902**

5.2 Can Sentiment Lexicon Help Identifying Parallelism?

Different algorithm behaves differently when using semantic features (*Feature 3 & 4*).

Naive Bayes Classifier: High dominance and high arousal are both the most informative features (see Fig. 5) and they rank even higher than syntactic similarity. These two features appear to have equal contribution to the classifier, which is probably because these two values are correlated with each other to some extent. According to Fig. 6, although dominance scores are usually higher than arousal scores (which explains why the threshold for high dominance is higher than that for high arousal), the dynamics of these two scores are quite similar throughout the text. It is noteworthy that sometimes the all words in a sentence do not have entries in the sentiment lexicon, and result in an extreme score of 0 (see Fig. 6). which might affect the performance of the classifier.

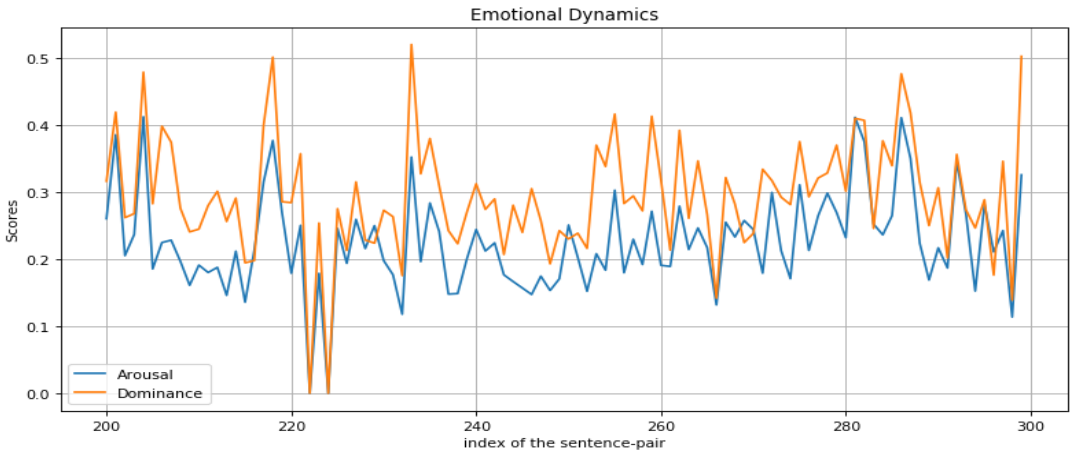
Fig. 5 Most Informative Features (Naive Bayes Classifier)

```
    simiLexical = True              T : F      =     26.6 : 1.0
       highDomi = True              T : F      =      6.5 : 1.0
      highArous = True              T : F      =      6.5 : 1.0
    simiLexical = False             F : T      =      5.0 : 1.0
   simiSyntactic = False            F : T      =      1.2 : 1.0
```

Syntactic feature is not as contributive as it for inner-sentence parallelism identification task. The reasons could be: Inner-sentence parallelism is often concerned with grammatical parallelism and is characterized with syntactic similarity, while the inter-sentence parallelism is usually in its rhetoric sense and featured with lexical repetition. As in the annotation guidelines no limit of length difference is set for sentence-pairs that have rhetoric parallelism, the similarity between PoS-tag sequences of these two sentences can be extremely low and thus become a less informative feature.

Classifiers from Scikit-learn: On the other hand, semantic features have adverse effect on KNN classifier and logistic regression system. According to Tab. 2, if the two semantic features (high A/D) are fed into the KNN classifier, its performance will decrease with respect to all three measures, especially recall. Under this situation, LR classifier have the same results with KNN. The decrease can be identified in recall and F1-score, but the impact of using semantic features is relatively small for this system. For Linear SVC classifier, no influence is found.

Fig. 6 Emotional Dynamics of Arousal- and Dominance-Dimension (for 100 sentences)



Tab. 2 Evaluation for class "T" (Influence of Using Semantic Features)

Algorithms	P	R	F
KNeighborsClassifier (sklearn)	0.8971(-0.007)	0.8618(-0.0259)	0.8747(-0.0155)
LogisticRegression (sklearn)	0.8971(+0.0004)	0.8618(-0.0037)	0.8747(-0.0023)
LinearSVC (sklearn)	0.8978	0.8877	0.8902

***High A/D vs. V/A/D similarity*:** As the module *SelectKBest* is able to show which features are relevant/irrelevant for classification in an intuitive way, it is decided to carry out a small test to check whether features 5, 6 and 7 (similarity in V/A/D scores), which are weeded out in the previous feature engineering section, are indeed irrelevant for classifier training. Basic idea of the test is to increase the parameter *k* by *n* and see which *n* features can be selected apart from the k (2) best features (lexical and syntactic features). The threshold for determining V/A/D similarity is decided by analysing Fig. 3 and Fig. 4 shown in section 4.3.2. The value of absolute difference that produces the highest proportion of parallelism is selected as threshold. The thresholds for extracting valence similarity, arousal similarity and dominance similarity are respectively 0.1, 0.6 and 0.6.

It is found that when the *n* is 1, high dominance is additionally selected. High arousal can also be selected when *k* is enlarged by 2. This proves that the similarity in V/A/D scores is less relevant than the other two semantic features for classifying parallelism which corresponds to the analysis results in section 4.3.2. In addition, the performance of KNN and LR becomes even worse when adding these 3 features. This test demonstrates from a different perspective that the previous feature engineering process is meaningful.

5.3 Error Analysis

Error analysis is not only important for understanding the classifier's behaviour and fine-tuning the feature extraction methods, but also advantageous for finding tricky data in the corpus. Typical false positive and false negative problems found after the final iteration are shown below.

5.3.1 False Positives

(1) Falsely extracted lexical features

As mentioned in section 3.1.2, some frequently occurred phrases can be repeated in two successive sentences just by coincidence. In the example below, the repetition of "in our" can be treated as the feature for anadiplosis. As lexical feature is the most contributive feature for all classification systems train in this project, this sentence pair is falsely tagged as "T". This problem is difficult to tackle, since the rhetoric feature of the lexical repetition is sometimes also unclear for human and sometimes depends on the context.

sent1: Church and charity, synagogue and mosque lend our communities their humanity, and they will have an honored place in our plans and **in our** laws.

sent2: Many **in our** country do not know the pain of poverty. But we can listen to those who do.

(Bush, 2001)

2) Inner-sentence parallelism can also have high arousal/dominance scores

The feature high arousal/dominance should not be limited to inter-sentence parallelism. It is found that some sentences with inner-sentence parallelism also have high A/D scores. And if the following sentence also happens to have high A/D scores this kind of sentence pair is likely to tagged as "T". In the following excerpt, sent1 contains inner-sentence parallelism and in the meantime, sent2 is part of an inter-sentence parallel structure. And their A/D scores are indeed both higher than the thresholds. (Note: the errors in this section are produced by NaiveBayesClassifier which uses high A/D scores as informative features).

sent1: Homes have been lost, **jobs shed, businesses shuttered**.

sent2: **Our** health care **is too costly.**

sent3: **Our** schools **fail too many.** (Obama, 2009)

This problem suggests that treating inner-sentence and inter-sentence parallelism separately is sometimes inappropriate.

There are also some false positive errors show the innovative feature of natural language generation and are worth further analysis. The sentence-pair below, for instance, could be considered as an on-going antithetic parallelism.

sent1: **We will seek** friendship and good will with the nations of the world, but we do so with the understanding that it is the right of all nations to put their own interests first.

sent2: **We do not seek** to impose our way of life on anyone, but rather to let it shine as an example—we will shine—for everyone to follow. (Trump, 2017)

5.3.2 False Negatives

1) Syntactic similarity is not extracted

Since the length difference tolerance is increased to 4 tokens for evaluating syntactic similarity between sentences, the similarity between PoS-tag sequences of two parallel sentences that have very different lengths can be low e.g.:

sent1: Technology_NNP is_VBZ almost_RB magical_JJ ._.

sent2: And_CC ambition_NN for_IN a_DT better_JJR life_NN is_VBZ now_RB universal_JJ ._.

(Clinton, 1993)

2) Parallelism or Repetition?

As the sentence-initial/final part is decided to contain at least 4 tokens (including punctuation), and the lexical repetition feature is extracted from the sentence-initial/final part, therefore the repetition of "America first" in the following sentence-pair is not recognized. This brings up a question: Should we treat this structure as simple repetition?

sent1: From this this day forward, it's going to be only **America first**.

sent2: **America first**. (Trump, 2017)

3) Repeated bigram is separated by appositives

As the annotation guideline specifies that the appositives that does not disrupt the effect of parallelism can be ignored when identifying rhetoric parallelisms, sometimes the repeated bigrams are separated by the appositive. In the following example, "you have" is separated by "my fellow Americans" and is therefore not considered as repeated by the feature extraction method.

sent1: And **you have** changed the face of Congress, the Presidency, and the political process itself.

Sent2: Yes, **you**, my fellow Americans, **have** forced the spring. (Clinton, 1993)

6 Conclusion and Discussion

In this project, semantic features of parallel sentences are represented with the help of NRC-VAD sentiment lexicon. Apart from manual analysis, different tools such as the visualization tool *matplotlib* and *SelectKBest* module are employed for feature engineering. All classifiers developed in this project have achieved good results. Best F1-score achieved is 89.02%. Compared with the classifier for inner-sentence parallelism, the syntactic similarity extracted based on PoS-tag sequence is not quite contributive for identifying inter-sentence parallelism, probably because inter-sentence parallelism is more characterized by lexical repetition. It is found that high arousal and high dominance are used as informative features in Naive Bayes algorithm, however, they can have adverse effect on KNN and

logistic regression classification algorithms. Based on the adverse effect of using similarity in V/A/D scores as features in different classifier, the overall semantic similarity of parallel sentences perceived by human seems not to be represented in V/A/D these 3 dimensions.

As the feature of high arousal/dominance score might not be limited to inter-sentence parallelism, these two features could also be used for identifying inner-sentence parallelism. The influence of using these sentimental features for identifying parallel structures within a sentence is worth further investigation. This project also demonstrates the difference and the relationship between inner- and inter-sentence parallelism. For studies need automatic parallelism identification, the combination of both classifiers should be advantageous.

References

Al-Ameedi, R.T., & Mukhef, R.N. (2017). Aspects of Political Language and Parallelism. Journal of Education and Practice.

Bakker, Iris & Van der Voordt, Theo & Boon, Jan & Vink, Peter. (2014). Pleasure, Arousal, Dominance: Mehrabian and Russell revisited. Current Psychology. 33. 405-421. 10.1007/s12144-014-9219-4.

Bauman, R., & Briggs, C. (2003). Voices of Modernity: Language Ideologies and the Politics of Inequality (Studies in the Social and Cultural Foundations of Language). Cambridge: Cambridge University Press. doi:10.1017/CBO9780511486647

Bradley, M.M., & Lang, P.J. (1999). Affective norms for English words (ANEW): Instruction manual and affective ratings. Technical report, The Center for Research in Psychophysiology, University of Florida.

梁社会,陈小荷,刘浏.先秦汉语排比句自动识别研究—以《孟子》《论语》中的排比句自动识别为例 (Study on automatic identification of parallel sentences in Pre-Qin Chinese — with automatic identification of parallel sentences in Mencius and the Analects of Confucius.).计算机工程与应用,2013,(19):222-226. DOI:10.3778/j.issn.1002-8331.1303-0471.

Mohammad, Saif. (2018). Obtaining Reliable Human Ratings of Valence, Arousal, and Dominance for 20,000 English Words. 174-184. 10.18653/v1/P18-1017.

穆婉青,廖健,王素格. 融合 CNN 和结构相似度计算的排比句识别及应用(A Combination of CNN and Structure Similarity for Parallelism Recognition)[J]. 中文信息学报，2018, 32(2): 139-146. http://jcip.cipsc.org.cn/CN/abstract/abstract2525.shtml

Plutchik, R. (1980). A general psychoevolutionary theory of emotion. In R. Plutchik & H. Kellerman (Eds.), Emotion: Theory, research, and experience: Vol. 1. Theories of emotion (pp. 3-33). New York: Academic.

Pustejovsky, James & Stubbs, Amber (2012). Natural Language Annotation for Machine Learning. A Guide to Corpus-Building for Applications. O'Reilly.

Russell, James & Barrett, Lisa. (1999). Core affect, prototypical emotional episodes, and other things called emotion: Dissecting the elephant. Journal of personality and social psychology. 76. 805-19. 10.1037//0022-3514.76.5.805.

Saleema AbdulZahra Almehmdawi (2018). Parallelism in One of Hillary Clinton's Speeches: A Critical Discourse Analysis. DOI: 10.18018/URUK/018-11/272-286

Song, W., Liu, T., Fu, R., Liu, L., Wang, H., & Liu, T. 2016. Learning to Identify Sentence Parallelism in Student Essays. COLING.

Winfried Menninghaus, Valentin Wagner, Eugen Wassiliwizky, Thomas Jacobsen, Christine A. Knoop. 2017. The emotional and aesthetic powers of parallelistic diction, Poetics, Volume 63, Pages 47-59, ISSN 0304-422X, https://doi.org/10.1016/j.poetic.2016.12.001.

熊李艳,林晓乔,钟茂生.面向自动写作的中文排比句抽取方法(Automatic writing based on Chinese parallelism sentence extraction method). 计算机应用研究, 2018, 35(06):1751-1755.

Y. Dai, W. Song, X. Liu, L. Liu and X. Zhao, “Recognition of Parallelism Sentence Based on Recurrent Neural Network,” 2018. IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, pp. 148-151. doi: 10.1109/ICSESS.2018.8663734