Meiosis and litotes in *The Catcher in the Rye* by Jerome David Salinger: text mining

Marta Karp[a], Nataliia Kunanets[a,b] and Yuliia Kucher[a]

[a] *Lviv Polytechnic National University 12 Bandera street, Lviv, 79013, Ukraine*
[b] *Ivan Franko National University of Lviv, Universutetska Street 1, Lviv, 79000, Ukraine*

Abstract

This paper deals with the different methods, particularly statistical analysis and text mining, which help in stylistic research. The examination of the lexical and semantic features of meiosis and litotes in the novel *The Catcher in the Rye* by Jerome David Salinger is presented as an example. The examination in question has been carried out with the help of the programming language R. To have a good-quality research, the specific features of litotes and meiosis have been explored thoughtfully. Therefore, the broad range of possible scientific views has been described, and, subsequently, we have made a general assumption of typical linguistic patterns of meiosis and litotes. Using the obtained insights, it is possible to apply different tools of text mining in stylistic research. The present paper outlines in detail the creation of concordances, word frequencies and sentiment analysis. To reach our goal, we have used the programming language R and the R packages which are distributed by members of the community. In the scope of concordances, the concept of Key Word in Context has been discussed as well, and the advantages of using concordances in stylistic research have been introduced. The possible implementation of statistical analysis in the research of litotes has been proposed and discussed. Within the framework of sentiment analysis, we have focused on the negation, and how it affects the opinion orientation. Thus, the present paper also aims to validate the importance of litotes in sentiment analysis, as litotes are directly linked to the effects of negation. The results of each stage of the research have been provided and meticulously discussed.

Keywords1

Meiosis, litotes, natural language processing, text mining, sentiment analysis

1. Introduction

The present research consists of several parts, namely the statistical analysis, the basic methods and instruments of text mining and the semantic-stylistic analysis. Therefore, the work includes both technical and philological ways of working with texts, particularly with the novel *The Catcher in the Rye* by Jerome David Salinger. A significant part of the present paper is dedicated to the more linguistic aspects of the data under the study, which are lexical and semantic features of meiosis and litotes. More than that, we have explored in which way the statistical analysis is helpful during these kinds of research, and how the litotes impact the machine processing of the texts.

2. Theoretical framework

The present paper mainly deals with the ways in which we have applied statistical approaches and methods of text mining for stylistic investigations. We are focusing on such stylistic devices as

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: martakarp26@gmail.com (M. Karp); nek.lviv@gmail.com (N. Kunanets); yuliia.s.kucher@gmail.com (Yu. Kucher)
ORCID: 0000-0002-7332-7739 (M. Karp); 0000-0003-3007-2462 (N. Kunanets); 0000-0002-0728-6617 (Yu. Kucher)

meiosis and litotes in particular. That is why first and foremost it is important to discuss the theoretical aspects of those devices. Both meiosis and litotes are the types of understatement, however, there are several significant distinctions between them. Therefore, we suggest a brief overview to better understand their specific features. So, meiosis stands for the figure of speech, which refers to the object in a special way, therefore deliberately reducing its significance. Commonly, they are functioning in the sentence as modifiers, and, subsequently, they are such parts of speech, as adjectives or attributive nouns. On other hand, litotes expresses the meaning of the understatement by denying some qualities to claim the opposite. So, litotes takes the form of "the negation of something", and on the syntactic level it generally embraces the next pattern: "not/no/none + something"; however, it is not limited to one. To clarify all the specific features of meiosis and litotes, we suggest to look further into the ideas of different scholars, who profoundly have worked with these stylistic devices. Hence, James Jasinski proposes such a definition of meiosis: "Meiosis is a statement that depicts something important in terms that lessen or belittle it". Besides, he states that: "Any verbal effort to make an event, an idea, or a person less significant is a form of meiosis" [9, p. 550]. Another researcher, Brett Zimmerman, who is interested in Edgar Allan Poe's style, offers the following description of the meiosis: "a lessening, sometimes belittling thing or person, possibly with a degrading epithet, or with the substitution of a word." [26, p. 43]. Very often the stylistic meaning of meiosis is reduced only to the ironic effect of the understatement, and we observe it in the definition of Bernard Marie Dupriez: "A figure which uses ironic understatement to represent something as in some way less than it is: a form of ironic emphasis." [3, p. 273]. A very comprehensive definition is given by Aida Besancon Spencer in her stylistic study of the Apostle Paul's style of written communication. She writes the following: "Quintillian says that meiosis may refer to a style as well as a figure. 1. When applied to a style, meiosis indicates "meagreness and inadequacy of expression," characterizing an 'obscure style rather than one which lacks ornament.'" 2. Meiosis and litotes are often not distinguished. Meiosis is a deliberately employed understatement, presenting something as less than it really is. It is belittle, often through a change of meaning of one word, as in using a degrading epithet [20, p. 195].

Among the most prominent scholars, who worked with litotes, we draw special attention to Otto Jespersen, Laurence R. Horn, Dwight Bolinger and Ton van der Wouden. We have picked exactly their works, as we believe that their views and ideas are perfectly suited to our research. Jespersen in his work *Negation in English and Other Languages*, describes the cases of double negation and states the following: "It seems to be a universal rule in all languages that two negatives make an affirmative, if both are special negatives attached to the same word; [...] But it should be noted that the double negative always modifies the idea, for the result of the whole expression is somewhat different from the simple idea expressed positively." [10, p. 63]. Ton van der Wouden provides us with a rather comprehensive analysis of the nature of litotes, where he, inter alia, refers to the Horn's analysis of litotes. Therefore, Wouden describes, that Horn's analysis of litotes has two parts: a semantic one and a pragmatic one. The semantic one embraces the ideas that two negations make an affirmative. The pragmatic one has been explained in the following interpretation of one of Grice's maxims, and is called *Division of Pragmatic Labour*, meaning: "The use of a longer, marked expression in lieu of a shorter expression involving less effort on the part of the speaker tends to signal that the speaker was not in a position to employ the simpler version felicitously." [24, p. 122-123]. Horn, in his paper *Duplex negatio affirmat...: The Economy of Double Negation* writes that "The expectation that two negatives SHOULD cancel out is a linguistic reflex of the logical Law of Double Negation (LDN), $\sim(\sim\alpha) \equiv \alpha$" [8, p. 80]. However, as he elaborates further on this matter, he suggests that there are many nuances. Thus, he cites Bolinger and his thoughts on litotic negation: "... the denial of the negative leaves the entire positive range open to whatever degree is appropriate. The litotes, in fact, call attention to this gradient – the hearer is invited to consider the degree to which the facts point." [2, p. 116]. The latter sentence is indeed overlapping with the *Division of Pragmatic Labour*, implying that litotes is stylistically less natural and more complex, and that the speaker usually uses the double negative (or the negative of the contrary) in lieu of simple positive description is quite deliberate. So, moving forward on Wouden's description, we have seen that he describes the so-called "grey zone" between one world and its negation. For example, in the next sentence: *It's not too bad when the sun's out, but the sun only comes out when it feels like coming out* [17, p. 202].

Here we see such litotes as *not too bad* and it is quite common in the English language. Taking Wouden's discourse as a basis, we have derived the following: the opposition to "bad" is "good", however, these words are not contradictory, but contrary or gradable antonyms, meaning that they do not refer to absolute qualities, therefore, they are not strict oppositions. The contrary antonyms are pictured as a scale with these two words as polar members, and what is the most important, they admit possibilities between them. In our case, the more direct word "good" is deliberately omitted in lieu of negating the opposite word "bad". Thus, the specific range between two meanings of "good" and "bad" is created, where "not bad" tends to be closer to "good", but, and it is the main point there, does not exactly substitutes it. It is pictured in the following scheme:

too bad	somewhat bad	somewhat good	too good
		not too bad	

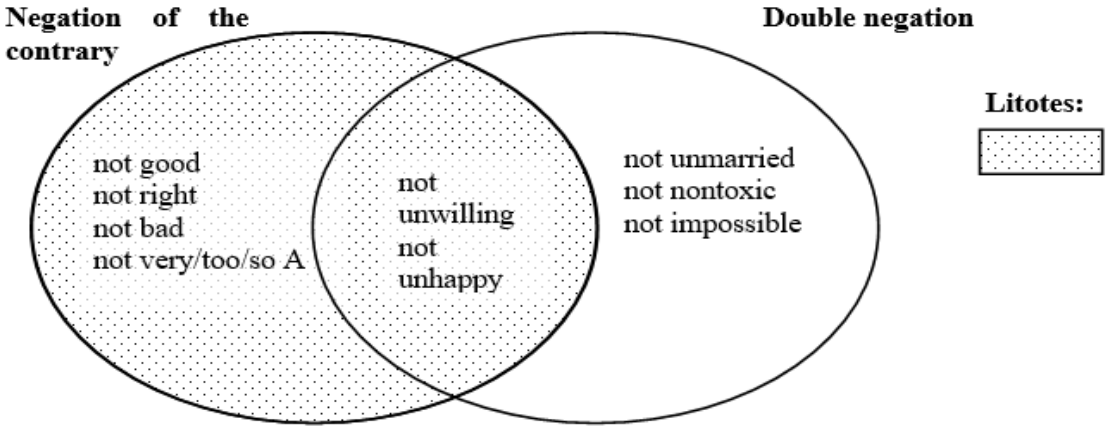
So, we have observed an area between two extremes *too bad* and *too good*, where neither of them applies. As Bolinger writes: "When intensifiers are present, the litotes tends to deny one end of a polarity to imply an encroachment on the other end. [...] The negative passes conceptually from the intensifier to the intensified, with the intensifier weakened to *rather*" [2, p. 116]. Thus, *not too bad* corresponds with *rather good*. However, Bolinger also mentions, that there is a discrepancy between the litotes with an intensifier, and without one. He calls the litotes without an intensifier a contradictory one – it expresses that the entire opposite range is open. For instance, if we are to examine the litotes *not unwilling*, the results are presented in the following scheme:

willing	···	unwilling
all that is not unwilling		

where *not unwilling* embraces any degree of willingness. On the other hand, Wouden presents a slightly different approach to the analysis of litotes. Drawing upon his notions, we have inspected litotes *not unwilling* in the following way:

1.	willing	...	unwilling
2.		not willing	
3.	not unwilling		
4.		not unwilling	

Thus, according to the Wouden, in the first row, we have seen two extremes, *willing* and *unwilling*, and the space between them, and it is exactly the grey zone, that has been mentioned before. It is the zone, where neither *willing*, nor *unwilling* is not applicable. We also have to keep in mind the fact, that there is no definite boundary between the boundary and the grey zone. The second row pictures what Wouden calls *logical denotation* of *not willing*. Logically, the negation of *willing* covers the space on the scale that remains. Similarly, with the third row, logical denotation of *not unwilling* is the rest of the scale, but *unwilling*. However, as we have seen in the fourth row, the pragmatic denotation of *not unwilling* opens the only possible range that is between the two extremes. Thus, we have made the next interim conclusion: *not unwilling* refers only to the middle part of the scale, so it is not direct opposition to the *unwilling*, but a litotes which carries the effect of understatement. Moving forward to the next important details about the features of litotes, both Horn and Wouden claim the following: "Litotetic constructions with nongradable predicates are ungrammatical or unfelicitous, equivalent to the straightforward expression, or figurative by necessity." [24, p. 123]. Horn, in turn, proposes, that "[...] litotes, so defined, does not require single, let alone, double negation." [8]. He also proposes that not all examples of double negations are obligatory litotes. We pay attention to this idea, as it has become clear that we are not entitled to assume that every pattern of double negation has the stylistic effect of litotes. So, Horn's point of view can be pictured in the following diagram:



Another important matter with the negation of the contrary is that it is utterly context, and sometimes even intonationally, dependent. We have already mentioned, that *not bad* is a quite popular example of litotes. However, if we take a look at the following extract, we see that the function of the phrase is not litotic:

He didn't know what the hell I was talking about, so all he said was "Oh" and took me up. Not bad, boy. It's funny. All you have to do is say something nobody understands and they'll do practically anything you want them to [17, p. 205].

As we have determined from the context, *not bad* does not act as underestimation. Its functions are exactly opposite – it has the meaning of overestimation, which can be rendered as *very good*. Stern in Horn [7, p. 356] writes the next statement about litotes: "*Not bad,* taken literally, leaves a large latitude, from *indifferent* to *excellent,* and may mean [sic] either, depending on the intonation used and the circumstances". Bolinger [2, p. 116], in turn, states that "A familiar example is the interjection *Not bad,* usually written with an exclamation point to indicate the intonation of surprise that suggests *Very good!*, but without to indicate the terminal fall-rise that damns with faint praise."

Statistical data and quantitative analysis are of great importance in nowadays linguistical research. They help the scientists to get clear and objective results, and it is a high priority in the realm of science. Thus, we are also working with the methods that are parts of the general scope of statistics for linguistics. We want our methods to be agile, veritable, and precise, so we believe that the best tool that satisfies all requirements, is programming language *R*. *R* is one of the most popular tools among statisticians and data miners, on the one level with such well-known languages as Python and MATLAB.

The workflow of analysing data consists of several steps:

- Accessing the data (we store information for analysis in different places, so inevitably we need to get the data into our application)
- Cleaning the data (the data may be kept in such formats, that is not appropriate for the analysis; also, some parts may be missing or/and miscoded, so we should take care of it)
- Annotating the data (make the relations between the pieces of data and what they represent)
- Summarising the data (statistical characterisation of the data)
- Visualization the data (the possibility to create different graphs, plots, histograms, charts etc.)
- Modelling the data (applying the mathematical and statistical rules to the set of data to identify relationships between its parts and accordingly make hypotheses for further investigations)
- Preparing the results (presenting the outcomes in the publication-quality and user-friendly formats) [11, p. 24]

The list above is the most general way to think about the data analysis. The *R*'s capability allows us to accomplish all of these steps, so this is our starting point from the statistical point of view. Another definition, which has been used often in the present paper is *Text Mining*. This term embraces the large scale of different concepts and practices; however, it is important to describe it generally to

ensure the clear comprehending of the further explanations, and to avoid ambiguity. Therefore, as Ian H. Witten states in the paper concerning text mining [25], this term is described as attempts to glean meaningful information from natural language text. The term *text mining* is confused with data mining, although they are quite different notions. The latter one is all about searching for patterns in data, and therefore its main goal is to retrieve the comprehensible information from the large amount of input data. What is also important to mention, is that the input data in data mining is implicit, meaning that raw data make no or little sense for human comprehension, and is only analysed with the help of different automatic techniques of data mining. The main problem of text mining is different. In the text mining, the input data is almost perfectly clear for human, as it is texts written in natural language. However, for the machine processing, the input data which has the form of the texts, might be even more complicated, than data from different databases. Working with text mining, you are indispensably to encounter the notion of Natural Language Processing, or NLP. To eliminate the term above, we are citing the definition, introduced by Elizabeth D. Liddy: "Natural Language Processing is a theoretically motivated range of computational techniques for analysing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications." [12]. At the present time, there are many various applications of NLP, for example, machine translations, automatic summarization, sentiment analysis, conversational agents (chatbots), and so on. In the present paper, we decide to use the following methods and techniques: word frequency, collocation, concordance and sentiment analysis.

One of the ways, in which we have explored the possibilities for text mining (or text analysis), is building the concordances. The concordance is a set of the searched words (or collocations), within the context, and they are utterly useful for the investigation of meiosis and litotes, especially in fiction. The concordance is best perceived in the form of KWIC (Key Word in Context), but the ways of presenting the outlines are not limited to KWIC. As Stefanowitsch [21, pp. 50-51] has rightly stated, KWIC concordance means displaying the hits for the query in the proximate context, so there are a particular number of words or characters to the left and the right of it. He also mentions that concordance provides an overview of the typical usage of word forms (or sets) that is of our interest. Hence, concordances in the form of KWIC have been presented in the paper.

A vitally important step in text analysis is *preprocessing*. This process is all about data cleaning, and it is crucial to complete this step in a correct way because with "dirty" data it is very hard to undertake proper analysis. In the realm of text data cleaning means several actions:

- Removing the punctuation, such as period, exclamations points, question marks, commas, semicolons, colons, hyphens etc.
- Changing all the words to the lower case
- Removing the numbers
- Removing the stop words

Concerning the last point in the list above, it might be necessary to elaborate on the concept of stop word. Traditionally, in the realm of text mining, the stop words are the words that bear no semantic value, and therefore have no or little impact on the overall text perception. Stop words are usually articles, prepositions, conjunctions and pronouns. For example, such words as *a*, *the* and *it* provide rather scarce information and low semantic significance. These stop words just confuse the researcher and may cause unnecessary problems, meaning that they are redundant, so it is advisable to get rid of them [6]. However, it is crucial to keep in mind, that the removal of stop words not always is considerate to be the best practice. There are cases where the removal of stop words can corrupt the process and therefore the results of the investigation. The case, which directly concerns the present research, is outlined as follows: the removal of stop words causes changes in the meaning of the undertaken text. For example, the removal of words with the meaning of negations, such as *not*, *no*, *non*, transforms the meaning of the sentence into exactly the opposite one. After the cleaning, in the sentence *"It is not a good thing"*, the only meaningful words for computer programme are *good* and *thing*, and it is analysed as something with positive connotations, however, we understand, that the original sentence has rather negative connotation.

Another important matter of the present paper is sentiment analysis. As Bing Liu [13] states in his paper *Sentiment Analysis and Subjectivity*: "Sentiment analysis or opinion mining is the

computational study of opinions, sentiments and emotions expressed in text". So, this specific type of analysis is all about the processing of people's opinions. The development of the Internet has resulted in the establishing of a new unique space, where people have become able to express their opinions in various ways (e.g., reviews, comments, blogs). For the last years, sentiment analysis has come a long way of development, and we distinguish the set of the most common features, that used in research and practice:

- Terms and their frequency (generally well-known technique in text analysis that is used in sentiment classification as well)
- Part of speech tags (the prevailing part of speech that is responsible for indicating the subjective opinions is an adjective)
- Opinion words and phrases (opinion words are the words with positive or negative connotations; for instance, the word *beautiful* has positive evaluation, while the word *rubbish* expresses clearly negative sentiment)
- Syntactic dependency (the word positions in sentences also influences the way that sentence is perceived)
- Negation (negations often change the opinion orientation, however, there are cases when the negative particles do not mean negation, as, for example, in pattern *not only ... but also*)

The main concern in the present study is a litotes and meiosis, thus, we have inspected the sentiment analysis through the prism of these stylistic devices. What is interesting here, is the methods, with the help of which, sentiment analysis deals with double negation and litotes. Professor Liu discusses the effects of negation in opinion mining, and outlines the next effects of negation:

- Negation Neg → Positive
- Negation Pos → Negative

These insights are quite logical, however, as we have mentioned before, the litotes is the understatement that is resulted from the double negation or the negation of the contrary. The negative particle in litotic construction does not state the direct opposite, but a weaker meaning. We will discuss the litotic effect in the sentiment analysis further in Section 3.3.

3. Practical analysis

The present study is dedicated to two major problems: how statistical data and data mining help during stylistic research, and how stylistic devices, especially litotes, impact the sentimental analysis. The main tool of our investigation has been *R*, which is a language and environment for statistical computing and graphics. We have decided to use this particular technology infrastructure, as it allows us to implicate various data analytic techniques. *R* provides a range of different instruments, starting from quite easy ones, that do not require special knowledge, and ending with advanced powerful tools. More than that, *R* have a very useful feature called packages. To elucidate this term, it is necessary to refer to Robert Kabacoff [11, p. 54]: "Packages are collections of *R* functions, data, and compiled code in a well-defined format." In other words, packages are shareable bundles of code. Thus, there are a huge community of other *R* users, who have already dealt with the same problems that may occur during our work, so with the help of packages it is possible to benefit from their contributions, and significantly optimise your workflow. This feature is what makes *R* such a popular tool for undertaking different investigation. More than saving your time, it also enhances the process of the research in a way, that makes the investigation more agile. What we mean by that, is that the investigation is divided into several steps, and it enables us to manage every step, hence, you gain more control over what you do. So, we believe that *R* is a perfect and balanced tool for such linguistic investigations, like ours.

3.1. Concordance

The building of the *concordance* is an important part of linguistic analysis. It enables rapidly and efficiently process a vast amount of data. Usually, the results are displayed in the form of keyword-in-context displays or KWICs. This form of KWIC implies the searched word (or phrase) is shown in

context, meaning that there is a couple of words before and after the searched word or phrase. For our analysis, we will use the package *quanteda* [1]. This package is used for managing and analysing text, and it also contains a very useful function *kwic*. It is quite easy to use, nevertheless, it is agile and provide with sufficient number of ways, in which the output concordance is presented. As our study is about meiosis and litotes, as well as their semantical and lexical features, the concordances give us insights that are quite important. Meiosis is a stylistic device with the functions of lessening the significance of something, thus, we have made an assumption that such words as *little, small* and *tiny* could be used. Therefore, to build a concordance, firstly we need to access the data. It has been done with the help of *tm* package [4]. Overall, *tm* package is a great framework for text mining applications, which also contains several useful functions, that we are going to apply later. The main structure for managing documents in *tm* is a *Corpus*, representing a collection of text documents. *Tm* also contains a function with the same name. As expected, the *Corpus* function creates a corpus. The first argument to *Corpus* is in which way we want to create the corpus. In our case, we have read PDF file from the directory with the help of the built-in function *DirSource*. Using the second argument, *readerControl*, we tell *Corpus* which reader to use to read the text from the PDF files. That has been *readPDF*, also a *tm* built-in function. The *readerControl* argument requires a list of control parameters, one of which is *reader*. After these manipulations, what we have got is a corpus, that in our case consists only of one document. Subsequently, we have used function *kwic*, that allows us to easily extract concordances. The *kwic* function takes the text (*x*) and the search pattern (*pattern*) as its main arguments but it also has some additional arguments, that helps to make the concordance more flexible. The simplest way of creating concordance with the searched word *tiny* is pictured in Figure 1.

kwic(as.character(corp[[1]]), pattern = "tiny")

	docname	from	to	pre	keyword	post	pattern
1	text26	507	507	personal habits . 7 A	tiny	bit of light came through	tiny
2	text33	220	220	serious . I have this	tiny	little tumor on the brain	tiny
3	text33	265	265	. And it's a very	tiny	one . They can take	tiny
4	text38	463	463	when she was a very	tiny	little kid . When she	tiny
5	text38	472	472	When she was a very	tiny	little kid , I and	tiny
6	text38	653	653	, and not such a	tiny	little kid any more ,	tiny
7	text47	309	309	It was one of those	tiny	little tables that if the	tiny
8	text47	421	421	kidding . At this other	tiny	table , right to my	tiny
9	text52	258	258	said . She had a	tiny	little wheeny-whiny voice . You	tiny
10	text67	175	175	watched a couple of very	tiny	kids on a seesaw .	tiny
11	text95	307	307	all when she was a	tiny	little kid . She's a	tiny
12	text114	401	401	. When she was a	tiny	little kid , and Allie	tiny

Figure 1: Concordance with word *tiny*

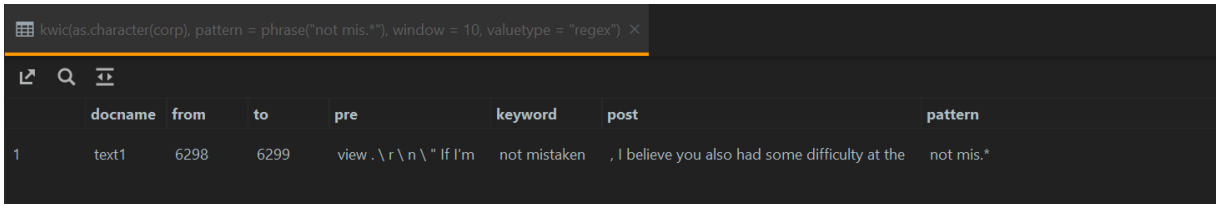
With the help of the concordance, we have distinguished such example of meiosis, as, for instance, *tiny little tumor*. The whole extract is the following:

"It isn't very serious. I have this tiny little tumor on the brain."

"Oh, no!" She put her hand up to her mouth and all. "Oh, I'll be all right and everything! It's right near the outside. And it's a very tiny one. They can take it out in about two minutes." [17, p. 75]

The next sentences after the *tiny little tumor* even enhance the effect of the meiosis. Thus, in such a way, we can investigate other occurrences of meiosis as well.

As for litotes, we have created concordances for them too. The most common patterns for litotes are double negation and negation of the contrary. We will use the same *kwic* function, however, at this step we have added extra arguments: *window* and *valuetype*. The former one specifies how many words/elements are shown to the left and right of the keyword. By default, there are 5 words by each side of the searched term, and for a better understanding of the context, we have expanded it to 10 words. The *valuetype* argument is responsible for the type of pattern matching: "glob" for "glob"-style wildcard expressions; "regex" for regular expressions; or "fixed" for exact matching. For example, we can create a concordance for the pattern "not + word with negative prefix", as in Figure 2.


kwic(as.character(corp), pattern = phrase("not mis.*"), window = 10, valuetype = "regex")

	docname	from	to	pre	keyword	post	pattern
1	text1	6298	6299	view . \ r \ n \ " If I'm	not mistaken	, I believe you also had some difficulty at the	not mis.*

Figure 2: Concordance with litotic pattern

We can investigate litotes *not bad* in the context accordingly. It is also important to mention that there can be other words, intensifiers in particular, between *not* and *bad*, thus, we can use regular expressions to achieve our goal (Figure 3).


kwic(as.character(corp), pattern = phrase("not [a-z] bad"), window = 10, valuetype = "regex")

	docname	from	to	pre	keyword	post	pattern
1	text1	12197	12199	brush your teeth . \ " Stradlater's all right He's	not too bad	, \ " I said . \ " You don't	not [a-z] bad
2	text1	25902	25904	got it . My mother gets very hysterical . She's	not too bad	after she gets \ r \ nsomething thoroughly digested ,	not [a-z] bad
3	text1	27577	27579	. \ r \ n \ " Pencey ? It's	not too bad	. It's not paradise or anything , but it's as	not [a-z] bad
4	text1	59663	59665	, if you want to know the truth . They're	not as bad	as \ r \ nmovies , but they're certainly nothing	not [a-z] bad
5	text1	79773	79775	nknown him , you'd know what I mean . It's	not too bad	when the sun's out , but the sun only \	not [a-z] bad
6	text1	88257	88259	If somebody at least listens , it's \ r \	nnot too bad	. \ r \ n \ " Daddy's going to	not [a-z] bad
7	text1	102439	102441	\ r \ n I was still sweating , but	not so bad	any more . I went over to the stairs and	not [a-z] bad

Figure 3: Concordance with litotes *not bad*

As we see, there are examples of litotes with intensifiers between the negative particle and the main word. Such cases are important to the same extent as the classical examples of litotes.

To sum up, concordances can provide a very convenient way of investigating the cases of litotes and meiosis in the context. From the philological point of view, it allows us to see and investigate the examples of the rhetorical devices. The concordances demonstrate the convenient way of analysing meiosis and litotes in the scope of the novel, as we have operated not only with the numbers of found examples but also immediately have seen the context and have regulated it by choosing the quantity of words or symbols before and after the searched keyword. This approach is quite convenient because it is almost impossible to undertake the appropriate analysis of the stylistic devices without

the context. Regarding the application of this method to the novel *The Catcher in the Rye*, some results have been outlined as follows: we have observed some examples of the meiosis and litotes in the text, however they are not numerous. Nevertheless, they still significantly contribute to the stylistic perception of the novel. It is also worth mentioning, that due to the specific recognisable syntactic pattern of litotes, they are more easily detectable via the text mining tools. More than that, these obtained concordances will serve as a basis for further research.

3.2. Word frequencies

Just like with the task of creating concordance, first of all, we have extracted a text in a format that is suitable for *R*. Therefore, we need a package that allows us to work with PDF format in *R*. Such package exists and it is called *pdftools* [15]. The *pdftools* function for extracting text is *pdf_text*. As result, we have a vector that contains the text of the PDF file. The length of a vector corresponds to the number of pages in the PDF file. In our original PDF file, we have 116 pages, hence, the length of our vector is 116.

With the help of such packages as *tidytext* [18], *tibble* [14], *dplyr* [23] and *ggplot2* [22], it is possible to undertake an assessment of words and n-grams frequencies, and, finally, to build plots with the results. Thus, recalling the steps of data analysis flow, we have transformed our raw data into tidy text format. Getting your text into tidy text format is believed to be a fundamental requirement to perform text mining and counting word frequencies. Therefore, we have changed the already transformed text of PDF format into *tibble* (*data frame*). A *data frame* is a two-dimensional array, where different columns contain different modes of data (numeric, character, and so on). There are differences between *tibbles* and *data frames*, but they are insignificant in the current study. After that, we have used the *unnest_token* function from *tidytext* that, roughly speaking, performs the *tokenization*. *Tokenization*, in NLP, is a process of transforming the human-readable text into machine-readable data. There are various ways of *tokenization*, the most common one is splitting the text into separate words. In the present study, we have used this method, as well as breaking the text into n-grams. *N-grams* are described as a consecutive sequence of words, where a bi-gram is a pair of two words, a tri-gram is a group of three words and so on. Moreover, the *unnest_token* function removes all punctuation and converts words to lower case. One more significant step is removing stop words. It is a common recommendation for creating tidy text formats, however, the negative words like *not*, *no*, *none* are considered stop words, but they are vital for our analysis, therefore we have skipped this step. Finally, using a function *filter*, we have counted and compared the frequency of occurrence of word combination *not bad*, which includes the features of litotes and word *good* with a direct meaning. Similarly, like in the case with concordances, it is crucial to remember about intensifiers, that occur between *not* and *bad*. We have visualised the results with the help of the *ggplot2* package and its functions (Figure 4).

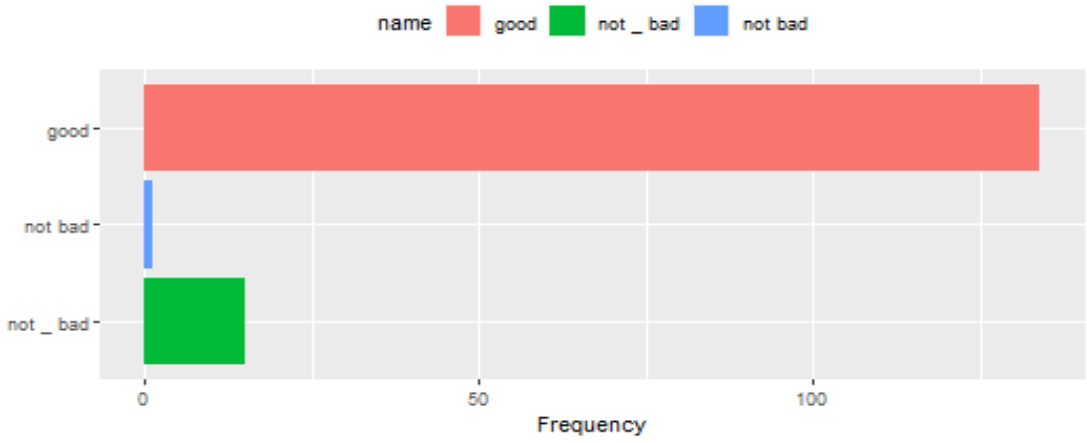


Figure 4: Frequencies of the word combination *not bad* and *good*

The plot has shown us the significant difference between frequencies of word combination *not bad*, word combination *not bad* with the intensifier, and *good*. Despite the existence of several litotic cases, it is clear that the non-litotic is prevalent for this case.

The same comparison can be made for the opposite – word combination *not good*, word combination *not good* with the intensifier, and *bad*.

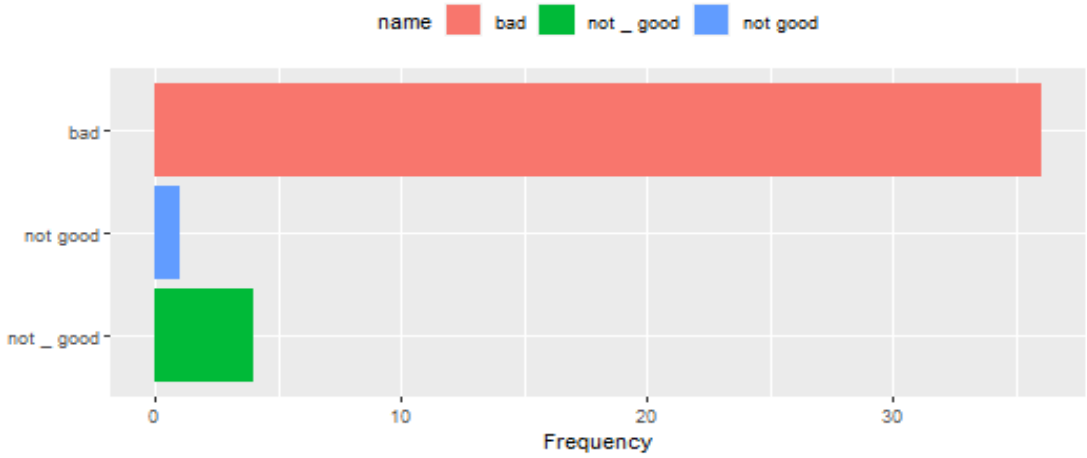


Figure 5: Frequencies of the word combination *not good* and *bad*

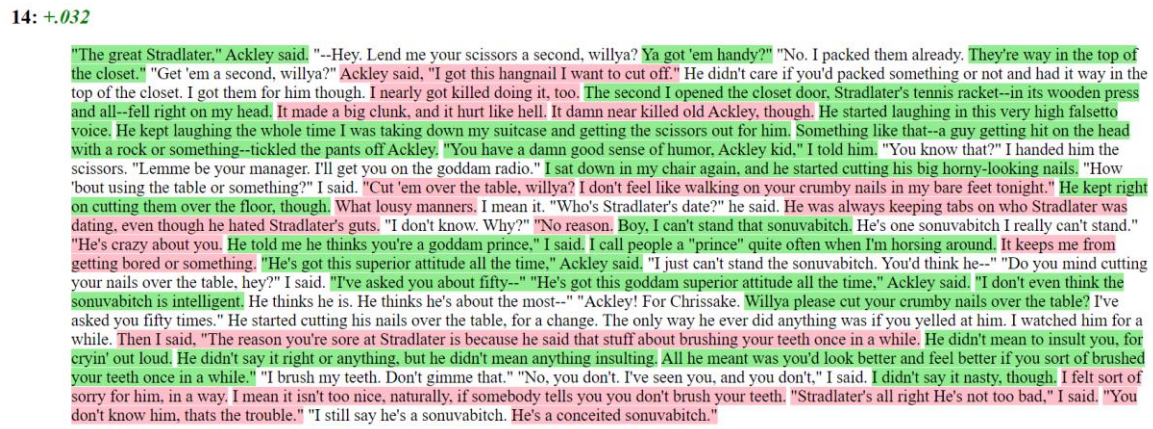
Different stylistic devices are an important part of the story. They enrich the language and attach a unique style to the text. As for litotes, they are especially important, as they lessen the significance of the statement in a special way, therefore provoke the reader to contemplate the situation. However, after analysing the concordances and word frequencies for this stylistic device, we conclude that the author has decided to use more direct forms instead of litotic constructions. Such peculiarity of usage of this type of understatement is one of the many others features of Salinger's idiostyle.

3.3. Sentiment analysis

There are several approaches to *sentiment analysis*. The first one is based on unigrams (or single words) and the second one is based on sentences. In other words, the difference is in units of tokenization – the former is done at the word level, while the latter is done at the syntactical level. The sentiment analysis that is undertaken with the help of single word tokenization is fruitful, however, there are cases when it can distort the facts. For instance, if we take the *"afinn"* lexicon from the *tidytext* package, which assigns words with a score that is in a range between – 5 and 5, with negative scores indicating negative sentiment and positive scores indicating positive sentiment. We have tried to undertake the analysis of sentence with litotic construction, using the *"afinn"* lexicon (Figure 5): *I mean it isn't too nice, naturally, if somebody tells you you don't brush your teeth* [17, p. 32].

Figure 6: Results of "*afinn*" sentiment analysis

After the analysis, we have got that the sentence has only one word which corresponds with *"afinn"* lexicon, and its evaluation is 3 points. It does not properly reflect the sentiment of the sentence. For such incidents, we have used the tools which undertake the tokenization on the level beyond the single word. One of such tools is *sentiment* [16]. It has been used for the calculation of the text polarity sentiment at the sentence level. Thus, after we get the text in an appropriate format, we have used the built-in functions *get_sentences*, *sentiment_by* and *highlight*. The first one is used for breaking the text into sentences. The second one actually performs the sentiment analysis. Finally, the last one allows to highlight positive/negative sentences as an HTML document (positive = green; negative = pink). The result is presented in Figure 6 (only one part of a whole text).


14: +.032

"The great Stradlater," Ackley said. "--Hey. Lend me your scissors a second, willya? Ya got 'em handy?" "No. I packed them already. They're way in the top of the closet." "Get 'em a second, willya?" Ackley said, "I got this hangnail I want to cut off." He didn't care if you'd packed something or not and had it way in the top of the closet. I got them for him though. I nearly got killed doing it, too. The second I opened the closet door, Stradlater's tennis racket--in its wooden press and all--fell right on my head. It made a big clunk, and it hurt like hell. It damn near killed old Ackley, though. He started laughing in this very high falsetto voice. He kept laughing the whole time I was taking down my suitcase and getting the scissors out for him. Something like that--a guy getting hit on the head with a rock or something--tickled the pants off Ackley. "You have a damn good sense of humor, Ackley kid," I told him. "You know that?" I handed him the scissors. "Lemme be your manager. I'll get you on the goddam radio." I sat down in my chair again, and he started cutting his big horny-looking nails. "How 'bout using the table or something?" I said. "Cut 'em over the table, willya? I don't feel like walking on your crumby nails in my bare feet tonight." He kept right on cutting them over the floor, though. What lousy manners. I mean it. "Who's Stradlater's date?" he said. He was always keeping tabs on who Stradlater was dating, even though he hated Stradlater's guts. "I don't know. Why?" "No reason. Boy, I can't stand that sonuvabitch. He's one sonuvabitch I really can't stand." "He's crazy about you. He told me he thinks you're a goddam prince," I said. I call people a "prince" quite often when I'm horsing around. It keeps me from getting bored or something. "He's got this superior attitude all the time," Ackley said. "I just can't stand the sonuvabitch. You'd think he--" "Do you mind cutting your nails over the table, hey?" I said. "I've asked you about fifty--" "He's got this goddam superior attitude all the time," Ackley said. "I don't even think the sonuvabitch is intelligent. He thinks he is. He thinks he's about the most--" "Ackley! For Chrissake. Willya please cut your crumby nails over the table? I've asked you fifty times." He started cutting his nails over the table, for a change. The only way he ever did anything was if you yelled at him. I watched him for a while. Then I said, "The reason you're sore at Stradlater is because he said that stuff about brushing your teeth once in a while. He didn't mean to insult you, for cryin' out loud. He didn't say it right or anything, but he didn't mean anything insulting. All he meant was you'd look better and feel better if you sort of brushed your teeth once in a while." "I brush my teeth. Don't gimme that." "No, you don't. I've seen you, and you don't," I said. I didn't say it nasty, though. I felt sort of sorry for him, in a way. I mean it isn't too nice, naturally, if somebody tells you you don't brush your teeth. "Stradlater's all right He's not too bad," I said. "You don't know him, thats the trouble." "I still say he's a sonuvabitch. He's a conceited sonuvabitch."

Figure 7: Results of the sentiment analysis

The sentence that we have discussed before has a negative evaluation. Thus, we have inferred that the analysis is correct. However, the litotes in the following sentence: *"He's not too bad," I said. "You don't know him, that's the trouble."* [17, p. 32] is assessed as negative, while from the theoretical background, section 2, we have outlined that *not too bad* is corresponds with *rather good*.

These outcomes show that some tools for sentiment analysis are better if we want to take into account the stylistic effect of the litotes. However, the litotic impact is considerably context-dependent, so, even if we are trying to analyse them in the scope of not a single word, but a whole sentence, the results still are not ideal.

4. Conclusions

In this paper, we have thoroughly examined some ways of implementing text mining and statistical methods into stylistic research. As a basis, we have chosen such stylistic devices as meiosis and litotes in *The Catcher in the Rye* by Jerome David Salinger.

The whole research has consisted of several constituent stages. Firstly, we have highlighted the important notions concerning the lexical and semantical characteristics of litotes and meiosis. We have also outlined the most general syntactical patterns of litotic expressions and examined the concepts of the negation of the opposite and double negation at this stage. The main principles of analysing data have been introduced as well. In the scope of *preprocessing*, we have examined the stop words and their impact on the quality of the results. We have also drawn our attention to the notions of sentiment analysis, and the role negation plays during the performance of such opinion assessment analysis. At the second stage of the research, we have undertaken the practical analysis of namely the creation of the *concordances*, the *word frequencies* and *sentiment analysis*. The programming language *R* and the *R* packages have been used as a tool for examination. The significant part of this stage has been the correct and demonstrable visualisation of the outcomes.

The results have shown us that we can successfully use the *R* tools to get valuable stylistic insights. The *concordances* are a great way of investigating the typical examples of meiosis and litotes in context. The *word frequencies* of litotic structures and their non-litotic correspondences, and their correlations have been visualized and thus provide us with the overall understanding of the

functioning of litotic expressions in the text. Finally, we have investigated different *sentiment analysis*' approaches, and conclude how each of them treats the litotes.

The present paper potentially leads to other researches in this field. Our work suggests that more investigations can be undertaken in the scope of text mining and stylistics. One of the possible directions can be related to the problem of the significance of litotes and meiosis in the text of different genres. For example, the correlation of occurrences of these stylistic devices throughout the range of various texts can be investigated.

5. References

[1] K. Benoit, K. Watanabe, H. Wang, P. Nulty, A. Obeng, S. Müller, A. Matsuo, quanteda: An R package for the quantitative analysis of textual data, Journal of Open Source Software 3(30) (2018). URL: https://quanteda.io. doi: 10.21105/joss.00774.

[2] D. Bolinger, Degree Words, Mouton, The Netherlands, 1972.

[3] B.M. Dupriez, A Dictionary of Literary Devices: Gradus, A-Z, University of Toronto Press, Toronto and Buffalo, 1991.

[4] I. Feinerer, K. Hornik, tm: Text Mining Package. R package version 0.7-8, 2020. URL: https://CRAN.R-project.org/package=tm.

[5] I. Feinerer, K. Hornik, D. Meyer, Text Mining Infrastructure in R, Journal of Statistical Software, 25(5) (2008) 1–54. doi: 10.18637/jss.v025.i05.

[6] E. Haddia, Xiaohui Liua, Yong Shib, The Role of Text Pre-processing in Sentiment Analysis, Procedia Computer Science 17 (2013) 26 – 32.

[7] L. R. Horn, A natural history of negation, CSLI Publications, Stanford, 2001.

[8] L. R. Horn, Duplex negatio affirmat...: The Economy of Double Negation in L. M. Dobrin, L. N. & R. M. Rodriguez (Eds.), CLS 27: Papers from the 27th Regional Meeting of the Chicago Linguistic Society. Part Two: The Parasession on Negation, Chicago Linguistic Society, Chicago, 1991, pp. 80-106.

[9] J. Jasinski, Sourcebook on Rhetoric, Sage Publications, Inc., California, 2001.

[10] O. Jespersen, Negation in English and Other Languages, Forgotten Books, London, 2012.

[11] R. Kabacoff, R in Action: Data analysis and graphics with R, 2nd. ed., Manning Publications Co., 2015.

[12] E.D. Liddy, Natural Language Processing in Encyclopedia of Library and Information Science, 2nd. ed., Marcel Decker, Inc., NY, 2001.

[13] B. Liu, Sentiment Analysis and Subjectivity in: N. Indurkhya, F. J. Damerau (Eds.), Handbook of Natural Language Processing, 2nd. ed.,Chapman & Hall, Boca Raton, FL, 2010.

[14] K. Muller, H. Wickham, tibble: Simple Data Frames. R package version 3.0.5., 2021. URL: https://CRAN.R-project.org/package=tibble.

[15] J. Ooms, pdftools: Text Extraction, Rendering and Converting of PDF Documents. R package version 2.3.1., 2020. URL: https://CRAN.R-project.org/package=pdftools.

[16] T. W. Rinker, sentimentr: Calculate Text Polarity Sentiment. R package version 2.7.1., 2019. URL: http://github.com/trinker/sentimentr.

[17] J. D. Salinger, The Catcher in the Rye, Little, Brown and Company, Boston, 1951.

[18] J. Silge, D. Robinson, tidytext: Text Mining and Analysis Using Tidy Data Principles in R, JOSS, 1(3) (2016). doi: 10.21105/joss.00037.

[19] J.M. Sinclair, Corpus, Concordance, Collocation, Oxford University Press, Oxford, 1991.

[20] A.S. Spencer, Paul's Literary Style: A Stylistic and Historical Comparison of II Corinthians 11:16-12:13, Romans 8:9-39, and Philippians 3:2-4:13, University Press of America, Inc., Lanham/Oxford, 1998.

[21] A. Stefanowitsch, Corpus Linguistics. A Guide to the Methodology. Textbooks in Language Sciences, Language Science Press, Berlin, 2020.

[22] H. Wickham, ggplot2: Elegant Graphics for Data Analysis, Springer-Verlag, New York, 2016.

[23] H. Wickham, R. Francois, L. Henry, K Muller, dplyr: A Grammar of Data Manipulation. R package version 1.0.3., 2021. URL: https://CRAN.R-project.org/package=dplyr.

[24] T. van der Wouden, Negative Contexts (Linguistics), Ph.D. thesis, University of Groningen, Groningen, The Netherlands, 1994.
[25] I. H. Witten, Text Mining, University of Waikato, New Zealand, 2004.
[26] B. Zimmerman, Edgar Allan Poe: Rhetoric and Style, McGill-Queen's University Press, Montreal & Kingston, 2005.