Dipartimento di Ingegneria e Scienza dell'Informazione

Corso di Laurea in
Informatica

Elaborato finale

Emotion dynamics of content co-creation in fanfiction communities

Supervisore

Prof. Alberto Montresor

Laureando

Matteo Circa

Anno accademico 2021/2022

Ringraziamenti

Prima di procedere con la trattazione, vorrei dedicare qualche riga a tutti coloro che mi sono stati vicini in questo percorso di crescita personale e professionale.

In primis, un ringraziamento speciale al mio relatore Alberto Montresor, per la sua immensa pazienza, per i suoi indispensabili consigli, per le conoscenze trasmesse durante tutto il percorso di stesura dell'elaborato.

Un ringraziamento speciale va ai miei supervisori Cristian Consonni e David Laniado, che mi hanno aiutato a condurre le ricerche, oggetto dell'elaborato, presso l'azienda Eurecat - Centre Tecnològic de Catalunya.

Ringrazio l'azienda Eurecat - Centre Tecnològic de Catalunya, in cui ho svolto un tirocinio formativo della durata di due mesi e complementare alla redazione della tesi, per l'ospitalità e per le abilità acquisite sul campo.

Un ringraziamento particolare va al mio collega Andrea Bragante per avermi accompagnato in questa stimolante esperienza e per gioire, insieme a me, dei traguardi raggiunti.

Ringrazio la mia famiglia che mi ha sempre sostenuto, appoggiando ogni mia decisione.

Infine, vorrei dedicare questo piccolo traguardo a me stesso, che possa essere l'inizio di una lunga e brillante carriera professionale.

Contents

Summary

In few years, there will be more books in computers and digital devices than does in our material literary collections, and with this the dynamics of reading a work take on new forms.

The end of *deep reading* [1] is a commonplace in public debates, especially when it comes to young people, books and the digital age. In contrast, however, young readers are writing and commenting on literary and non-literary texts at an unprecedentedly increasing rate. These users who act as both producers and consumers, in the publishing industry, are known as *prosumers*. They are capable of influencing the publications of works, generating discussions but most importantly engaging the communities around them. The content generated turns out to be of great value and gives rise to what are called the new avenues for publishing works by authors.

Specifically, prosumers are identified as the millions of users who participate and contribute to online communities of interest, called *fanfiction communities*. Fanfiction communities are communities that collect existing stories and create new content from them. Platforms such as Wattpad, Fanfiction, Archive of Our Own, and specific communities on Reddit represent this reality, and go one step beyond canonical content by offering alternative content. Here readers share their thoughts and emotional reactions to specific paragraphs, or to the entire work. The comments that readers write briefly and spontaneously interrupt their reading activity, before continuing it. This provides access to detailed data of what readers think about the work as long as they read it, providing real-time data, as a kind of *thinking aloud* [2]. The goal is to bring readers and users who actively participate in these online communities back to the center of attention.

The phenomenon of *digital social reading* needs to be considered for two main reasons. The first is that there is a great amount of reading activity taking place on *digital social platform* such as Wattpad, which as of 2019 boasts more than 80 million monthly visitors. The second is that user comments, in the margins of chapters, are an invaluable source for empirically studying their judgments, and the emotional valence of these.

This thesis presents several analyses of how fiction is transmitted and experienced by readers, via the Archive of Our Own (abbreviated as AO3) shared reading platform. The dataset analyzed contains hundreds of thousands of fan-written stories, millions of users, and tens of millions of interactions between them. Specifically, the research consists of collecting and analyzing user-generated content from fanfiction communities, and studying the dynamics of content co-creation. To characterize the generated themes and content, their relevance, temporal aspects, and conflict and collaboration dynamics, as well as the roles of authors and non-authors, their performance, and influence in communities, specific metrics are implemented that identify structural and emotional properties of comment text.

The study focuses on the analysis of emotions and feelings of users' comments, with respect to a work. There are lexicons with associated experimental studies, that is, dictionaries where words are linked to feelings. The LIWC [3] dictionary can analyze some text and capture multiple categories of emotions. Comments from a work are aggregated and sentiment analysis is conducted on these to highlight emotions and how they vary over time, to infer patterns and abstract global considerations valid for all communities.

Since fan-created content takes inspiration from well-known books, comic books, movies, and TV shows, new communities arise around the title of a work, a universe, or an author. The dataset analyzed covers seven of these communities, and they were selected to ensure some diversity in both size and content type. From communities with millions of users and hundreds of thousands of works to communities with a few thousand users and hundreds of works, the list of chosen communities

[1] deep reading: it is the active process of reflective and deliberate reading to improve comprehension and enjoyment of a text

[2] thinking aloud: it is the expression of one's thoughts as they arise

[3] https://www.liwc.app/

includes the following names: Marvel, Harry Potter, Sherlock Holmes, Percy Jackson, The Lord of the Rings, Twilight, and Warriors. The prosumers' behaviour was modeled using their footprints left on the Archive of Our Own website, thanks to features made available by the platform itself, such as leaving a comment to discuss or share insights and reflections, and leaving a *kudos* [4] to support the authors' work.

The analysis finds its foundation in two data structures: one containing general information of the works of the different communities, such as author, title, summary, tags, number of kudos, number of comments; the other containing the information of the comments related to a single work, organized in *threads* [5], with specific information such as the author of that comment, an identifier to the comment to which the user is responding, date and time of posting, and text of the comment.

Social and textual dimensions are explored for works written in the English language and spread throughout much of the world. The investigation is divided into three groups of reflection: analyzing certain categories of emotions and studying their different presence, in terms of percentages, among different communities; analyzing individual communities to infer context and find similar characteristics; and analyzing authors' behaviours to see how their involvement changes when they comment in response under their own works versus when they respond to works that are not theirs. Focusing on authors and their activities, as well as seeing the network of comments under a work from the author's perspective, is key to better understanding these *complex sociotechnical digital networks*.

The work is conducted on a collective scale: sentiment values are calculated as the sum of all readers' comments. The result expresses the emotional valence of a collective act of reading. In this sense LIWC is used with its default features. Comment sentences are broken into *tokens*, that is, taken by single word at a time, and aggregated in their entirety, or by entities, such as work, community, work chapters, author, non-author, or even by temporal aspects such as day or month. Emotional arcs are then plotted using the *moving window procedure* technique, to maintain a minimum of distortion of the original sentiment values: the scores for each *text chunk* [6], in fact, are simply recalculated as the average of all surrounding scores (by default, always around 10% of the total). Finally, and unlike other studies, the comments are grouped into a single text chunk and not as the average of the sentiment values of each individual comment. This approach was necessary because LIWC takes into account word frequency to calculate sentiments: scores for long chunks are statistically more reliable than shorter ones. In fact, on short comments, you have discrete variables, i.e., the comments consist of one, two or three words and the percentages of sentiments do not distribute correctly as they do on longer texts, where moreover the noise generated by parts of human language that are difficult to capture, such as sarcasm, is also mitigated.

In general, the sentiments of the comments are easy to interpret, as they are often short and a direct expression of the readers' thoughts and emotions. [7]

The results of the analysis show the growth of an overall increase in the use of fanfiction platforms from 2020, the year the Covid-19 pandemic begins. Moreover, on long texts, positive emotions are strongly negatively affected compared to any other category of emotion. A further educational result suggests that authors of works behave differently when commenting on others' works with regard to the expression of negative words. Compared with an average commentator, they criticize more and use fewer words of appreciation.

The entire research work is part of the larger European project Möbius [7], an initiative funded under the European Commission's Horizon 2020 program that aims to modernize the European publishing industry by reshaping traditional value chains and business models, uncovering the potential of prosumers and offering new enriched media experiences.

[4] kudos: it is praise and honor received for an achievement, similar to likes on social platforms
[5] thread: in social media it is a string of posts that make up a conversation
[6] text chunk: it is a portion of text
[7] https://mobius-project.eu/

1 Background

This chapter briefly outlines the working context. Section 1.1 describes the fanfiction community analyzed in this project. Next, Section 1.2 describes the source data structures, their characteristics and contents. Finally, Section 1.3 discusses the data structures created to conduct the research.

1.1 Analyzed fanfiction community: Archive of Our Own

Archive of Our Own [1] is a fan-created, fan-run, nonprofit, noncommercial open source repository for transformative fanworks, like fanfiction, fanart, fan videos, and podfic. The site was created in 2008 by the Organization for Transformative Works (abbreviated OTW) [2] and went into open beta in 2009. As of May 2022, Archive of Our Own hosted more than 9 million works in over 49000 fandoms. With an AO3 account, you can share your own fanworks, get notified when your favorite works, series, or users update, participate in challenges, keep track of works you've visited and works you want to check out later. In addition, readers share their thoughts and emotional reactions to the entire work through comments after registering on the site.

There is a tagging system on the platform to classify content; you can browse the archive by searching by fandom (Books & Literature, Movies, Music & Bands, etc.), then by movies for example, and finally by individual work. You can also sort and filter by ratings, warnings, categories, characters, crossovers, and read a work chapter by chapter.

The history of AO3 is of relevance. In 2007, a site called FanLib was created with the goal of monetizing fanfiction. Fanfiction was authored primarily by women, and FanLib, which was run entirely by men, drew criticism. This ultimately led to the creation of the nonprofit Organization for Transformative Works, which created Archive of Our Own. [1]

1.2 The starting point

This part covers the technical aspects of how the data is structured and stored on the permanent storage. Each fandom has its data stored across two MongoDB collections [3]. The first collection contains all the metadata of the works, while the second one is dedicated to the comments. Collections are then further split into documents [4].

The data structure concerning the data of works in general is structured as follows:

- `_id`: the work id on AO3.
- `authors`: a list of users who have contributed to writing the work. Most of the times the works are released under a single user name, although there are a few exceptions where there are collaborations on producing the fanfiction.
- `title`: the title of the work.
- `nchapters`: the number of chapters that have been released so far.
- `expected_chapters`: the number of chapters that the author is expecting to release.
- `complete`: it indicates whether the author is going to publish new chapters in the future or the work has been completed.

[1] https://archiveofourown.org/

[2] Organization for Transformative Works: is a nonprofit organization established by fans to serve the interests of fans by providing access to and preserving the history of fanworks and fan culture in its myriad forms

[3] MongoDB collections: a MongoDB collection can be seen as the equivalent of the tables in relational databases

[4] documents: a document in a NoSQL database corresponds to rows in a table in a SQL database, but is much more flexible

- `status`: it describes the status of the work as a human readable string. The possible values are In Progress or Complete.
- `date_updated`: the last time the authors edited their work.
- `hits`: the number of clicks the work has received. This number increases each time the work is loaded, therefore it might not correlate perfectly with the number of distinct users that access a specific work.
- `language`: the main language the work is written in.
- `summary`: a short work summary written by the author of the work.
- `tags`: a list of optional and user-defined tags added by the author (e.g.: Mystery, Crossover, Drama).
- `nkudos`: the number of kudos the work has received.
- `nbookmarks`: the number of bookmarks a work has received.
- `ncomments`: the number of total comments a work has received.
- `words`: the number of total words of a work.
- `chapters`: a nested structure where each item in the list represents one chapter and is described by two inner fields:
  - `id`: identifier of the chapter;
  - `title`: the title of the chapter.
- `date_published`: the date and time when the work was added to AO3.
- `kudos`: the users who left a kudo on the work. Each item in the list represent one username.
- `bookmarks`: a nested structure where each item represents one bookmark having the following attributes:
  - `author`: the username of the user who created the bookmark;
  - `datetime`: the date when the bookmark was created;
  - `text`: the optional message that the user chose to attach to the bookmark;
  - `type`: it can be either a Recommendation, Public or Private, although by far the most common value is Public.
- `fandoms`: list of fandom names in which the work is featured, since crossovers, which are fanfictions across multiple universes, may contain a list with more than one item.

The data structure for comments on a work contains the following information:

- `id`: the comment id on AO3. This identifier can be used to retrieve the thread rooted in the comment with a given id.
- `author`: the username of the user who left the comment.
- `chapter`: the chapter number the user has commented on.
- `datetime`: the date and time the user have left the comment.
- `parent_id`: the id of the comment the user has replied to. In case it is a top level comment, the value is null.
- `work_id`: the id of the work associated to this comment. This is the same identifier named _id in the above collection.
- `text`: the content of the comment.

1.3 My data structures

Part of the study related to the community Comments are aggregated by:

- Day
  - Per community
  - Per work
- Month
  - Per community
  - Per work
- Authors
- Non authors
- Chapters of the work

The data structure of the comments are supplemented by adding the following information:

- `work_author`: a list of users that have contributed to writing the work.
- `nkudos`: the number of kudos the work has received.
- `average_thread_length`: the average thread length of the work.
- `text_length`: the total number of words in the `text` after tokenizing [5] it.
- `ncomments`: the number of comments, 1 by default otherwise the number that follows from the aggregation.
- `Posemo`: Positive emotion (abbreviated Posemo) percentage estimated in the `text`.
- `Negemo`: Negative emotion (abbreviated Negemo) percentage estimated in the `text`.
- `Anx`: Anxiety (abbreviated Anx) percentage estimated in the `text`.
- `Anger`: Anger percentage estimated in the `text`.
- `Sad`: Sadness (abbreviated Sad) percentage estimated in the `text`.

Part of the study related to the prominence of the author Comments are aggregated to allow various types of comparison. In specific:

- Per community
  - Author vs. non author
  - Authors on their works vs. others
  - Authors on their works vs. author on other's works
- Per work
  - Comparison of the author's behaviour with that of outsiders commentators
- Per author
  - Comparison of the author's behaviour when it is and when it is not

[5] tokenizing: tokenization is essentially splitting a phrase, sentence, paragraph, or an entire text document into smaller units, such as individual words or terms. Each of these smaller units are called tokens

2 Research questions

The main idea of the project is to move from fan-created stories to traditional books. We study these communities and how users interact. The questions we will try to answer are: “Will a story be successful? What are the trending topics? What are the different types of users?”. We will try to characterize the users and the content. How? We will take a work and see the user interactions and comments, and on these we will do an emotion-sentiment analysis.

The investigation first wants to analyze certain categories of emotions and study their different presence, in terms of percentages, among different communities. This is done by taking a category of emotion and comparing its percentage values among different communities.

It is interesting to do the same thing but within a community and diluted over time, by month for example. This is done by looking at the percentage of a category in each month’s comments of a work and drawing a line to see the trend. Next we will try to derive a context and find similar characteristics among communities. Of relevance will be comparing the trends of the analyzed emotion categories with each other and noticing if there are some that are dependent on others and understanding how these vary as activity in the community changes.

The last part of the research will focus on analyzing authors’ behaviors to see how their involvement changes between author and non-author activity, that is, when they comment to others’ works. We will try to answer questions such as: “What role does the author play? Is he or she involved in the comments? Does he/she respond? How?” Again: “In a community where people talk more to each other is there less anger? Which emotions will prevail in a community where the author is more engaged in the comments?” A further analysis is to go look at the comments written by the author compared to those of others. Does the author tend to write positive or negative comments compared to others? This will involve going to define the role of the author, emotionally speaking.

Finally, we will cover variations in emotion at different levels of depth and side-by-side with some thread structure metrics. Will comments with lower indentation be the most controversial?

3 Tools

As the last step before diving into the analysis, here is a quick overview of the tools that were used throughout the research.

3.1 Python

The following are the libraries used for data analysis:

- *NumPy* [1], short name derivation from for Numerical Python, is a Python library that allows performing numerical calculations. The main benefit of NumPy is that it allows for extremely fast data generation and handling. NumPy has its own built-in data structure called `nparray` which is similar to the normal Python list but can store and operate on data much more efficiently.

- *pandas* [2] is a high-level data manipulation tool that is built on the Numpy package. Its key data structure is called the DataFrame and it allows to store and manipulate tabular data in rows of observations and columns of variables.

- *SciPy* [3] builds on top of NumPy as well using its array data type and related functionality, to provide common tools for scientific programming such as linear algebra or statistics.

The following are the libraries used for data visualization:

- *Matplotlib* [4] is a multi-platform data visualization library built on a NumPy array. One of Matplotlib's most important features is its ability to play well with many operating systems, graphics backends, and output types. In recent years, however, the interface and style of Matplotlib have begun to show their age.

- Luckily, recent Matplotlib versions make it relatively easy to set new global plotting styles, and people have been developing new packages that build on its powerful internals to drive Matplotlib via cleaner, more modern APIs. *Seaborn* [5], for example, is one of them and it can be used as a wrapper around Matplotlib's API. However, even with wrappers like these, it is still often useful to dive into Matplotlib's syntax to adjust the final plot output.

- *Bar Chart Race* [6] is a library to make animated bar chart races in Python with Matplotlib.

The following is the library used to interact with the MongoDB Server:

- *PyMongo* [7] is the client library we use to interact with the MongoDB Server. MongoDB is one of the most important NoSQL databases nowadays and it makes working with data simple. It is built on a philosophy that prioritizes performance and efficiency, which distinguishes them from the traditional relational databases and that makes storing and querying a large amount of data very efficient.

The following is the package used for sentiment analysis:

[1] https://numpy.org/
[2] https://pandas.pydata.org/
[3] https://scipy.org/
[4] https://matplotlib.org/
[5] https://seaborn.pydata.org/
[6] https://pypi.org/project/bar-chart-race/
[7] https://pypi.org/project/pymongo/

- *liwc* [8] is the Python package we use to load and parse the LIWC dictionary from the `.dic` file format and count category matches on provided texts. This is not an official LIWC product nor is it in any way affiliated with the LIWC development team or Receptiviti [9].

3.2 Jupyter Notebook

Jupyter [10] is an interactive web tool known as a computational notebook, that can be used to combine software code, computational output, explanatory text, and multimedia resources in a single document. Using a Jupyter notebook is only one way to code in Python. However, notebooks require everything to be laid out in cells, which gives the possibility to run them on the fly and in any order. In this research, Python 3 (ipykernel [11]) is used as the Kernel for the notebook.

3.3 LIWC

The core logic of Linguistic Inquiry and Word Count (LIWC) comes from decades of scientific research demonstrating that people's language can provide extremely rich insights into their psychological states, including their emotions, thinking styles, and social concerns. Sometimes, these insights are fairly obvious and straight-forward. For example, if someone is using a lot of words like happy, excited, and elated, they are probably feeling happy, and we can use this information to reliably estimate their current emotional state. Oftentimes, however, the relationships between verbal behavior and psychology are much, much less obvious. For example, people who are more confident and higher in social standings tend to use "you" words at relatively high rates, and "me" words at relatively low rates. Here, too, decades of empirical research — particularly research using LIWC as a scientific instrument — provides us with specialized ways of understanding, explaining, and quantifying psychological, social, and behavioral phenomena.

LIWC comes with over built-in dictionaries created to capture people's social and psychological states. Each dictionary consists of a list of words, word stems, emoticons, and other specific verbal constructions that have been identified to reflect a psychological category of interest. For instance, the "cognitive processes" dictionary includes over 1,000 entries that reflect when a person is actively processing through information, both in general and more specific ways. The "affiliation" dictionary includes over 350 entries that reflect a person's need to connect with others, including words like "community" and "together" among others.

Do not forget that LIWC, like all text analysis tools, is a relatively crude instrument. It can sometimes make errors in identifying and counting individual words. Consider the word "mad" — a word that is counted in the anger dictionary. Usually, today, the word "mad" does reflect some degree of anger. Sometimes, however, it expresses joy ("he's mad for her") or mental instability ("mad as a hatter"). Fortunately, this is seldom a problem because LIWC takes advantage of probabilistic models of language use. Yes, in a given sentence, the word "mad" might be used to express positive emotion. However, if the author is actually experiencing positive emotion, they would generally tend to use more than one positive emotion word, and most likely few other anger words, which should result in a high positive emotion score and low anger score. An important thing to remember is that the more words that you analyze, the more trustworthy are the results. A text of 10,000 words yields far more reliable results than one of 100 words. Any text with fewer than 25-50 words should be looked at with a certain degree of skepticism. In this analysis, a filter of at least 50 words is put on the text analyzed. [2] [5]

In this research, the dictionary (file `.dic`) is used to analyze words in LIWC categories instead of the LIWC application. The percentage calculation is then done merely by dividing the number of words in each category by the total number of words recognized in the tokenization phase. The

[8] https://github.com/chbrown/liwc-python

[9] Receptiviti: is the world's most powerful, extensively validated, and widely-used psychology-based language analysis platform for understanding human emotions, personalities, motivations, and psychology from language

[10] https://jupyter.org/

[11] ipykernel: is powerful interactive Python shell and a Jupyter kernel to work with Python code in Jupyter notebooks and other interactive frontends

version used is the previous version of LIWC-22 [12], which is the fourth version known as LIWC2015.

The most recent evolution, LIWC-22 (Pennebaker et al., 2022), has significantly altered both the dictionary and the software options to reflect new directions in text analysis. However, for example, in the study of Pennebaker et al. (2015), the correlations between the word frequencies counted with LIWC2015 and those obtained with LIWC2007 were very large — most of them were above 0.95 — indicating that the new version tends to detect very similar linguistic patterns from one corpus to another as the old versions. The same is assumed to occur between LIWC-22 and LIWC2015.

3.3.1 The LIWC2015 Framework

Both the standard downloadable and web based versions of the LIWC2015 application rely on an internal default dictionary that defines which words should be counted in the target text files. Words contained in texts that are read and analyzed by LIWC2015 are referred to as *target words*. Words in the LIWC2015 dictionary file will be referred to as *dictionary words*. Groups of dictionary words that tap a particular domain (e.g., negative emotion words) are variously referred to as subdictionaries or word categories.

For each text chunk, LIWC2015 reads one target word at a time. As each target word is processed, the dictionary file is searched, looking for a dictionary match with the current target word. If the target word is matched with a dictionary word, the appropriate word category scale (or scales) for that word is incremented.

The LIWC2015 Dictionary is the heart of the text analysis strategy. The default LIWC2015 Dictionary is composed of almost 6,400 words, word stems, and select emoticons. Each dictionary entry additionally defines one or more word categories or subdictionaries. For example, the word cried is part of five word categories: sadness, negative emotion, overall affect, verbs, and past focus. Hence, if the word cried is found in the target text, each of these five subdictionary scale scores will be incremented. As in this example, many of the LIWC2015 categories are arranged hierarchically. All sadness words, by definition, belong to the broader "negative emotion" category, as well as the "overall affect words" category. Note too that word stems can be captured by the LIWC2015 system. For example, the dictionary includes the stem hungr* which allows for any target word that matches the first five letters to be counted as an ingestion word (including hungry, hungrier, hungriest). The asterisk, then, denotes the acceptance of all letters, hyphens, or numbers following its appearance.

Each of the default LIWC2015 categories is composed of a list of dictionary words that define that scale. Table 1 provides a list of the LIWC2015 dictionary categories analyzed, scales, sample scale words, and relevant scale word counts.

Table 1: LIWC2015 Output Variable Information

Category	**Abbrev**	**Examples**	**Words in category**	**Internal Consistency (Uncorrected α)**	**Internal Consistency (Corrected α)**
Psychological Processes					
Positive emotion	posemo	love, nice, sweet	620	.23	.64
Negative emotion	negemo	hurt, ugly, nasty	744	.17	.55
Anxiety	anx	worried, fearful	116	.31	.73
Anger	anger	hate, kill, annoyed	230	.16	.53
Sadness	sad	crying, grief, sad	136	.28	.70

In Table 1, we include the internal reliability statistics, that are the alpha reliabilities. The LIWC Anger scale, for example, is made up of 230 anger related words and word stems. In theory, the more that people use one type of anger word in a given text, the more they should use other anger words in the same text. To test this idea, we can determine the degree to which people use each of the 230 anger words across a select group of text files and then calculate the intercorrelations of the word use. Uncorrected internal consistency alphas are based on Cronbach estimates; corrected alphas are based on Spearman Brown. [6]

[12] LIWC-22: is the latest version of LIWC

3.3.2 A practical example

For example, if LIWC were analyzing the first line of the novel Paul Clifford by Edward Bulwer-Lytton (1842):

"*It was a dark and stormy night*"

the program would first look at the word "it" and then see if "it" was in the dictionary. It is and is coded as a function word, a pronoun, and, more specifically, an impersonal pronoun. All three of these LIWC categories would then be incremented. Next, the word "was" would be checked and would be found to be associated with the categories of verbs, auxiliary verbs, and past tense verbs. After going through all the words in the novel, LIWC would calculate the percentage of each LIWC category. So, for example, we might discover that 2.34% of all the words in a given book were impersonal pronouns and 3.33% were auxiliary verbs. The LIWC output, then, lists all LIWC categories and the rates that each category was used in the given text. [8]

3.3.3 Example of usage with the Python package [3]

This example reads the LIWC dictionary from a file named `English_LIWC2015_Dictionary.dic`, which looks like this:

```
%
1   funct    # id category
2   pronoun
[...]
%
a   1 10     # word id_category
abdomen*     146 147
about    1 16 17
[...]
```

Loading the lexicon

```python
import liwc
parse, category_names = liwc.load_token_parser('LIWC2007_English100131.dic')
```

- `parse` is a function from a token of text (a string) to a list of matching LIWC categories (a list of strings).
- `category_names` is all LIWC categories in the lexicon (a list of strings).

Analyzing text

```python
import re

def tokenize(text):
    for match in re.finditer(r'\w+', text, re.UNICODE):
        yield match.group(0)

gettysburg = '''Four score and seven years ago our fathers brought forth on
  this continent a new nation, conceived in liberty, and dedicated to the
  proposition that all men are created equal. Now we are engaged in a great
  civil war, testing whether that nation, or any nation so conceived and so
  dedicated, can long endure. We are met on a great battlefield of that war.
  We have come to dedicate a portion of that field, as a final resting place
  for those who here gave their lives that that nation might live. It is
  altogether fitting and proper that we should do this.'''.lower()
gettysburg_tokens = tokenize(gettysburg)
gettysburg_length = len(list(tokenize(gettysburg)))
```

Now, count all the categories in all of the tokens, and print the results:

```
from collections import Counter
gettysburg_counts = Counter(category for token in gettysburg_tokens
                            for category in parse(token))
print(gettysburg_counts)
#=> Counter({'funct': 58, 'pronoun': 18, 'cogmech': 17, ...})
```

And finally calculate the percentages by simply dividing the word count of that category by the total number of words recognized in the tokenization stage:

```
d = { 'category': [], 'percentage': [] }

for category in gettysburg_counts:
    d['category'].append(category)
    d['percentage'].append((gettysburg_counts[category]
                            / gettysburg_length) * 100)

df = pd.DataFrame(data=d)
df
#=> DataFrame({
               'category': ['funct', 'pronoun', 'cogmech', ...],
               'percentage': [56.9%, 17.6%, 16.7%, ...]
               })
```

3.3.4 The categories analyzed

This analysis focuses on the following categories of emotions:

- **Emotional expression**
  - Positive and negative emotion
  - Anxiety
  - Anger
  - Sadness

They map on LIWC's categories as follows:

- **Emotional expression**
  - Posemo and Negemo
  - Anx
  - Anger
  - Sad

There are also other categories that capture features of language (pronouns, temporal verbs, etc.) and processes of perception (See, Hear, Feel), which were not explored in depth in this research.

4 Results

The main results of the analysis of the communities of Marvel, Harry Potter, Sherlock Holmes, Percy Jackson, The Lord of the Rings, Twilight, and Warriors are presented below.

4.1 Background

First, it is important to have an idea of the size of communities in terms of Number of Works, Number of Users, Number of Authors, and Number of Comments. Wanting to emphasize that the Number of Users also encompasses Authors, who are nothing more than users who write works. As can be seen in Table 2, the research is spread over small, medium and large communities. This provides a deeper understanding of how communities differ, especially emotionally, according to their size.

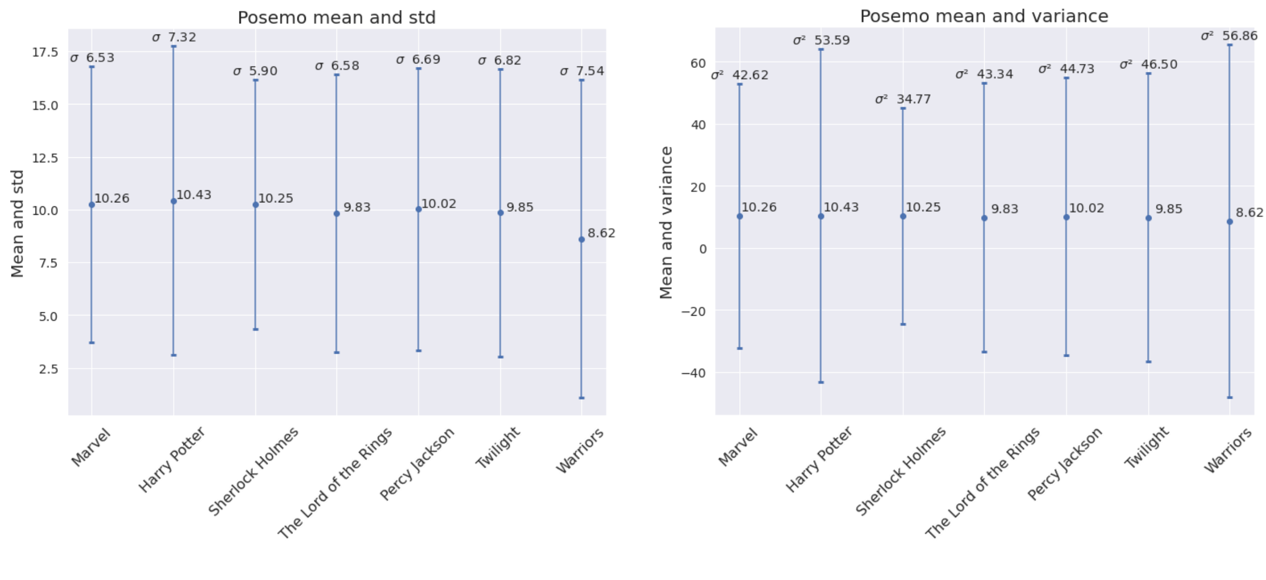
Table 2: List of the seven communities analyzed on AO3, and described in numbers

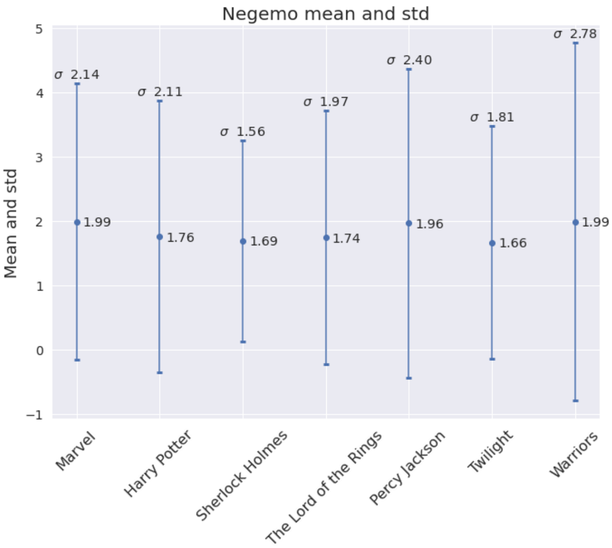
Community	Number of works	Number of Users	Number of Authors	Number of Comments
Marvel	295 172	4 592 163	1 718 165	10 946 620
Harry Potter	153 505	2 534 172	699 432	6 350 061
Sherlock Holmes	69 712	1 048 751	406 434	2 749 016
Percy Jackson	12 296	136 802	30 538	311 157
The Lord of the Rings	10 650	106 235	27 601	289 388
Twilight	5 452	59 907	8 515	141 402
Warriors	1 302	8 226	2 426	24 472

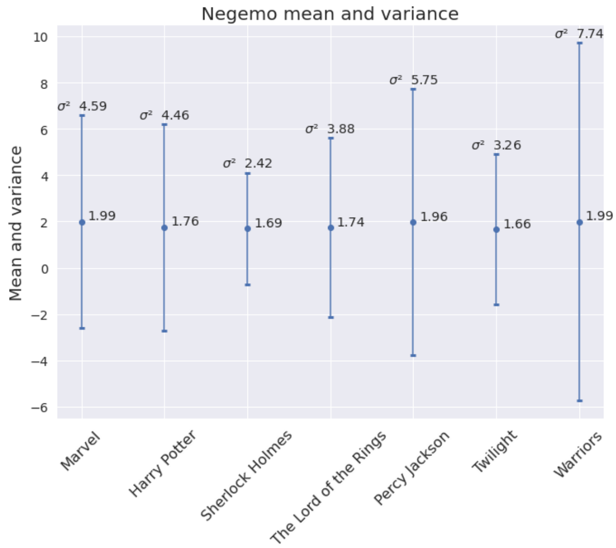
Remember that the standard deviation (std) tells us, on average, how far each score lies from the mean in the data set, while the variance tells us the degree of spread in the data set. The more spread the data, the larger the variance is in relation to the mean.

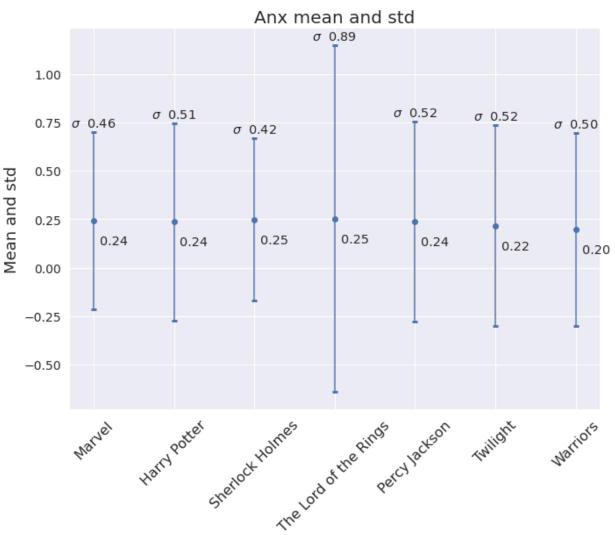
It does not make much sense to put categories side by side, also since one word might repeat in multiple categories. You have to take a category, like Posemo and compare it between different communities (Figure 1).

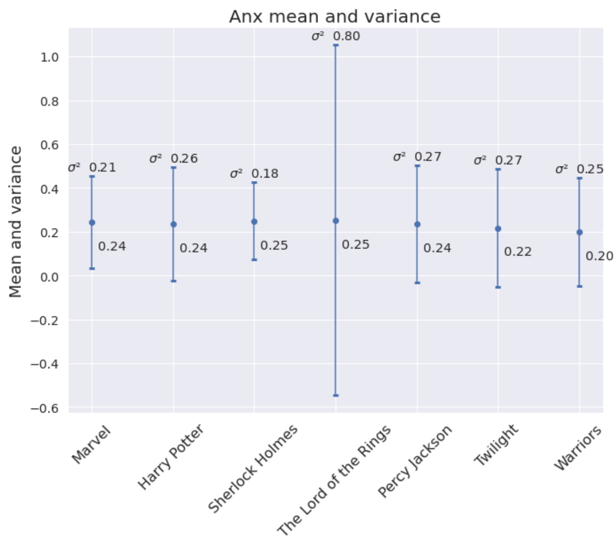
Figure 1: Error bars of the mean, std and variance for the five emotions Posemo, Negemo, Anx, Anger and Sad

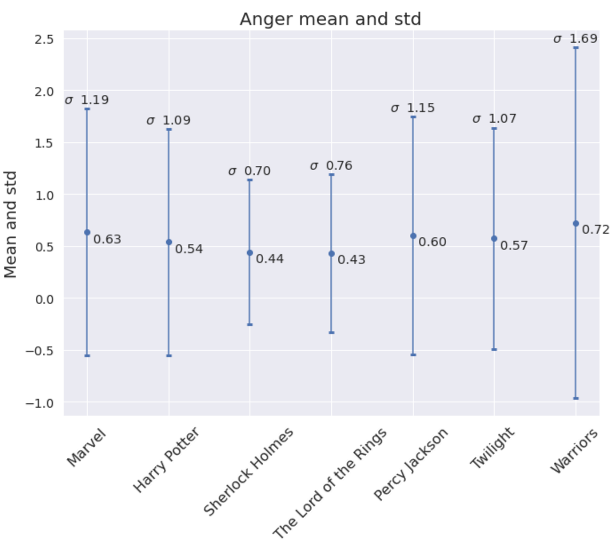


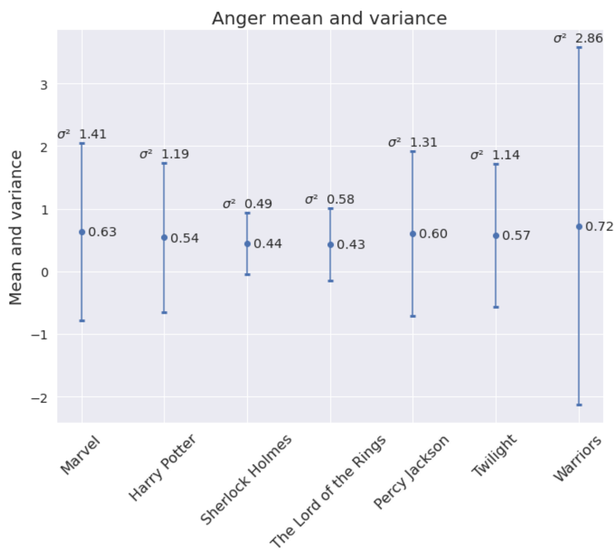

Negemo mean and std
Mean and std
σ 2.14
1.99
σ 2.11
1.76
σ 1.56
1.69
σ 1.97
1.74
σ 2.40
1.96
σ 1.81
1.66
σ 2.78
1.99
Marvel
Harry Potter
Sherlock Holmes
The Lord of the Rings
Percy Jackson
Twilight
Warriors

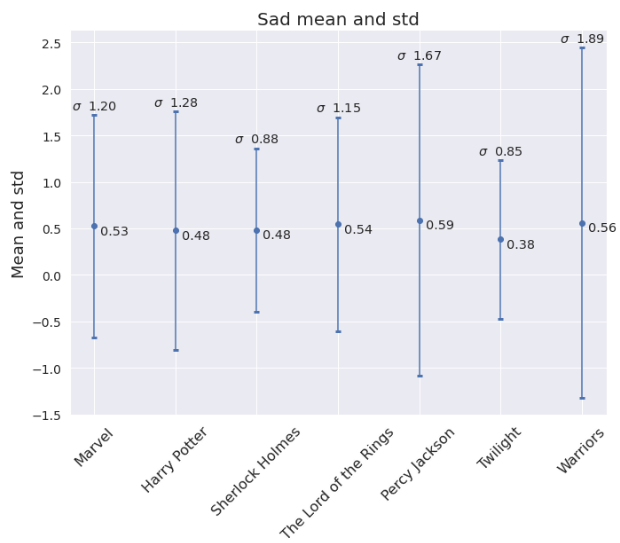

Negemo mean and variance
Mean and variance
σ² 4.59
1.99
σ² 4.46
1.76
σ² 2.42
1.69
σ² 3.88
1.74
σ² 5.75
1.96
σ² 3.26
1.66
σ² 7.74
1.99
Marvel
Harry Potter
Sherlock Holmes
The Lord of the Rings
Percy Jackson
Twilight
Warriors

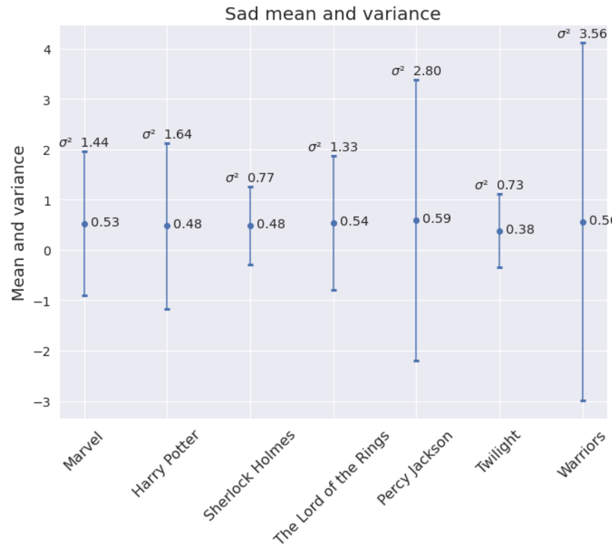

Anx mean and std
Mean and std
σ 0.46
0.24
σ 0.51
0.24
σ 0.42
0.25
σ 0.89
0.25
σ 0.52
0.24
σ 0.52
0.22
σ 0.50
0.20
Marvel
Harry Potter
Sherlock Holmes
The Lord of the Rings
Percy Jackson
Twilight
Warriors


Anx mean and variance
Mean and variance
σ² 0.21
0.24
σ² 0.26
0.24
σ² 0.18
0.25
σ² 0.80
0.25
σ² 0.27
0.24
σ² 0.27
0.22
σ² 0.25
0.20
Marvel
Harry Potter
Sherlock Holmes
The Lord of the Rings
Percy Jackson
Twilight
Warriors


Anger mean and std
Mean and std
σ 1.19
0.63
σ 1.09
0.54
σ 0.70
0.44
σ 0.76
0.43
σ 1.15
0.60
σ 1.07
0.57
σ 1.69
0.72
Marvel
Harry Potter
Sherlock Holmes
The Lord of the Rings
Percy Jackson
Twilight
Warriors


Anger mean and variance
Mean and variance
σ² 1.41
0.63
σ² 1.19
0.54
σ² 0.49
0.44
σ² 0.58
0.43
σ² 1.31
0.60
σ² 1.14
0.57
σ² 2.86
0.72
Marvel
Harry Potter
Sherlock Holmes
The Lord of the Rings
Percy Jackson
Twilight
Warriors





This comparison shows that emotions occur overall with the same amount among the seven communities. Specifically, it is noted that:

- The community with highest Posemo is Harry Potter with 10.43%, the one lowest is Warriors with 8.62%.
- The communities with highest Negemo are Marvel and Warriors with 1.99%, the lowest is Twilight with 1.66%.
- The communities with highest Anx are Sherlock Holmes and The Lord of the Rings with 0.25%, the lowest is Warriors with 0.20%.
- The community with the highest Anger is Warriors with 0.72%; the lowest is The Lord of the Rings with 0.43%.
- The community with the highest Sad is Percy Jackson with 0.59%; the lowest is Twilight with 0.38%.

4.2 In search of correlations

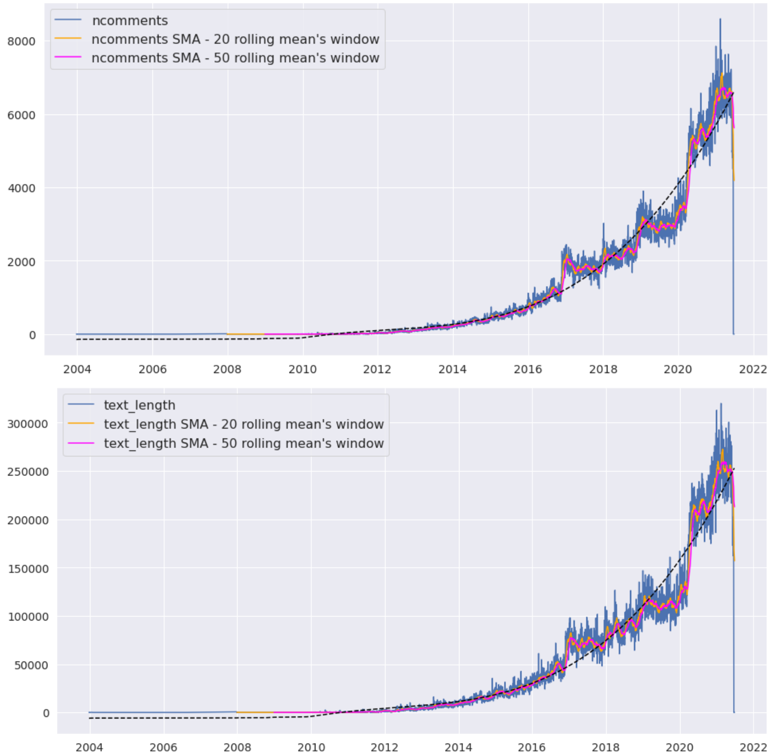
It is interesting to see the trend of emotions within a community and diluted over time, by day and month for example. Let us look at the percentage of the analyzed LIWC categories' in each year's comments and draw a line to see the trend.

A *moving average* can help filter noise and create a smooth curve from an otherwise noisy curve. It is important to note moving averages lag because they are based on historical data, not current ones. The most commonly used Moving Averages (MAs) are the simple and exponential moving average. Simple Moving Average (SMA) takes the average over some set number of time periods. So a 10 period SMA would be over 10 periods (usually meaning 10 days or 10 months). The Simple Moving Average formula is a very basic arithmetic mean over the number of periods, as shown in the Equation 4.1.

$$SMA = \frac{A_1 + A_2 + ... + A_n}{n} \tag{4.1}$$

Looking at all communities, there is a substantial increase in activity over time, especially after the beginning of the Covid-19 pandemic, and then a drastic drop before 2022. As can be seen in Harry Potter from Figure 2, the number of comments and the length of comments, in terms of words, go hand in hand.

Figure 2: Trend by day of the number of comments and length of text (Harry Potter community)



Regarding emotions, differences are observed between communities as large as Marvel, Harry Potter, and Sherlock Holmes and as small or medium-sized as Percy Jackson, The Lord of the Rings, Twilight, and Warriors. For larger communities, this average behavior is observed:

- Posemo declined in the early years, but this was due to a lack of content that arrived only in the utmost years. Therefore, if you look from 2014 onward, when the percentage calculation of LIWC categories becomes more accurate, you can see that Posemo grows slightly.
- Negemo, Anx, and Anger increase steadily, though varying little.
- Sad stays constant.

An example of this is shown in Figure 3 in the left column, from the Harry Potter community. For small and medium-sized communities, this average behavior is observed:

- Posemo has steadily declined over the years.
- Negemo, Anx, and Anger increase steadily.
- Sad rises little or stays constant.

An example of this is shown in Figure 3 in the left column, from the Twilight community.

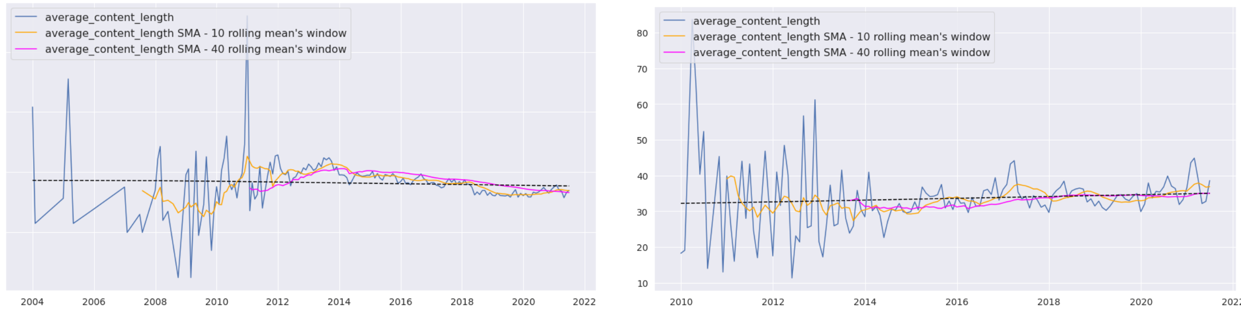
In general, it can be seen that the increase in activity over time, with some peaks, has affected the percentage of Posemo in communities, which has steadily decreased over the years. Does this mean that on longer texts and in the case of a higher number of comments the Posemo category has left space for other categories or is it simply standard that on a longer text the percentage of Posemo or any other category is reduced? Or does it mean that longer texts lead to a more careful review by commenters and therefore less Posemo? Maybe other categories take over? Does the Posemo category win out or is it more present, in terms of percentage, when texts are shorter?

These doubts sparked the hunt for correlations, but first it was necessary to identify the most careful reviews by commenters such as *average content length*, a simple ratio of comment length to number of comments.

Figure 3: Trend by month for the five emotions Posemo, Negemo, Anx, Anger and Sad (Harry Potter community on the left and Twilight community on the right)

It emerges that the average length of content varies differently depending on the size of the community. In fact, in larger ones it gradually decreases over time, while in smaller or medium-sized ones it gradually rises over time, as observable from Figure 4, where on the left we find a large community like Marvel and on the right a medium-sized one like Percy Jackson.

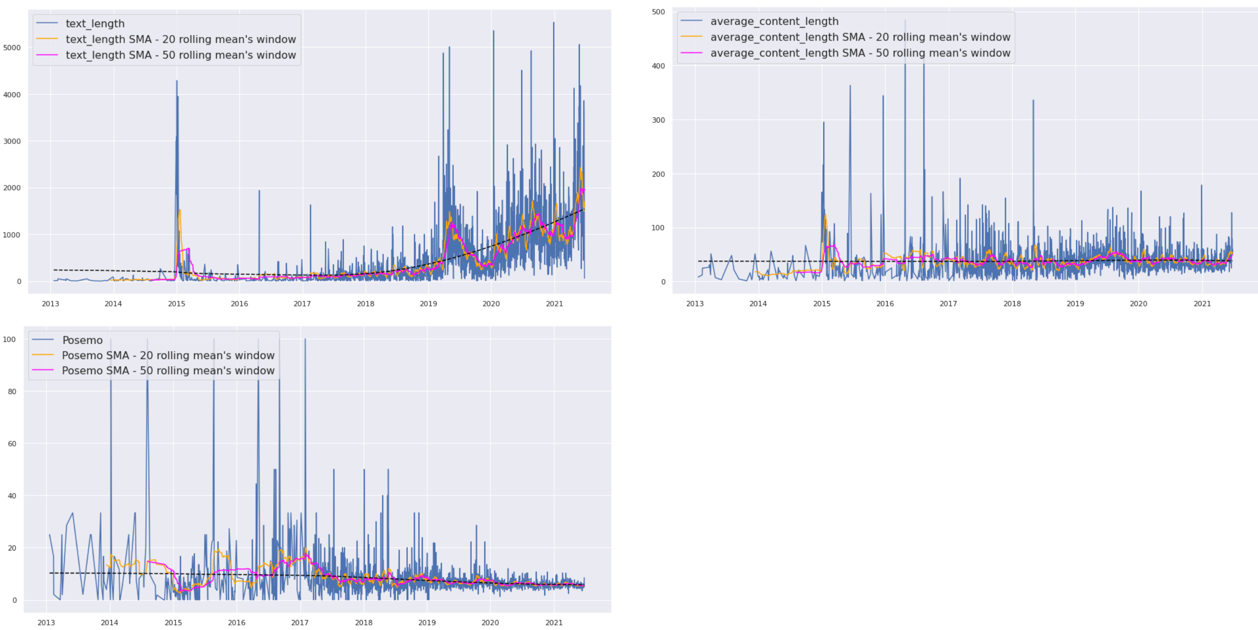
Figure 4: Trend by month of the average content length (Marvel community on the left and Percy Jackson community on the right)



Looking at trends in average content length, it follows that it is not always true that increased activity in the community leads to an increase in average content length. Moreover, overall it is not possible to compare Posemo with average content length, since the latter remains mostly flat over time.

This last point is still not entirely convincing. In fact, if we observe a generic spike in activity, especially in medium-small communities, the average length of content suddenly jumps up and simultaneously Posemo drops dramatically. This seems to say that the average length of content affects Posemo and thus it would seem that longer texts lead to more careful review by commenters and thus with less Posemo. An example of peak activity is that of the Warriors community in the year 2015, shown in Figure 5.

Figure 5: Daily trends in text length, average content length, and Posemo (Warriors community)



These growing doubts and more are answered in the correlation matrices, analyzed in the next section.

4.2.1 Correlation matrices between emotions and more

Let us go into detail by first analyzing a *correlation coefficient*. A correlation coefficient is a numerical measure of some type of correlation, meaning a statistical relationship between two variables. The variables may be two columns of a given data set of observations, often called a sample, or two components of a multivariate random variable with a known distribution. A simple *Pearson Correlation Coefficient* is used for the purpose of this research.

About Pearson Correlation Coefficient

Consider a dataset with two features: x and y. Each feature has n values, so x and y are n-tuples. Say that the first value x_1 from x corresponds to the first value y_1 from y, the second value x_2 from x to the second value y_2 from y, and so on. Then, there are n pairs of corresponding values: (x_1, y_1), (x_2, y_2), and so on. Each of these x-y pairs represents a single observation.

The Pearson (product-moment) correlation coefficient is a measure of the linear relationship between two features. It corresponds to the ratio of the covariance of x and y to the product of their standard deviations. It is often denoted with the letter r and called Pearson's r. You can express this value mathematically with the Equation 4.2:

$$r = \sum_{i=1}^{n}((x_i - mean(x))(y_i - mean(y)))^2 (\sum_{i=1}^{n}(x_i - mean(x))^2 \sum_{i=1}^{n}(y_i - mean(y))^2)^{-1} \tag{4.2}$$

The mean values of x and y are denoted with mean(x) and mean(y). This formula shows that if larger x values tend to correspond to larger y values and vice versa, then r is positive. On the other hand, if larger x values are mostly associated with smaller y values and vice versa, then r is negative.

Here are some important facts about the Pearson Correlation Coefficient:

- The Pearson Correlation Coefficient can take on any real value in the range $-1 \leq r \leq 1$.
- The maximum value $r = 1$ corresponds to the case in which there is a perfect positive linear relationship between x and y. In other words, larger x values correspond to larger y values and vice versa.
- The value $r > 0$ indicates positive correlation between x and y.
- The value $r = 0$ corresponds to the case in which there is no linear relationship between x and y.
- The value $r < 0$ indicates negative correlation between x and y.
- The minimal value $r = -1$ corresponds to the case when there's a perfect negative linear relationship between x and y. In other words, larger x values correspond to smaller y values and vice versa.

The above facts are summerized in Table 3. [4]

Table 3: Table summarizing Pearson Correlation Coefficient

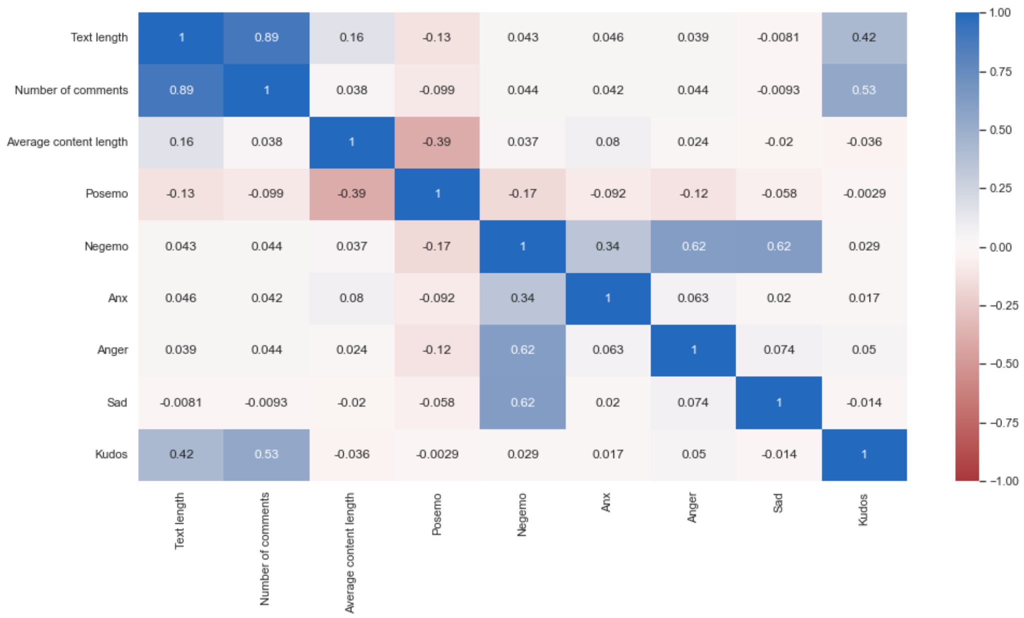
Pearson's Value	**Correlation between x and y**
equal to 1	perfect positive linear relationship
greater than 0	positive correlation
equal to 0	no linear relationship
less than 0	negative correlation
equal to -1	perfect negative linear relationship

About correlation matrix

A correlation matrix is a table containing correlation coefficients between variables. Each cell in the table represents the correlation between two variables. The value lies between -1 and 1. A correlation matrix is calculated for each of the seven communities, and from these a single one is derived by overlaying them and simply averaging the values. These below are the results, which can be observed in Figure 6:

- While Negemo, Anx, Anger, Sad are not correlated with the text length of comments, Posemo has a slight negative correlation. Thus, **on longer texts Posemo would seem to be more negatively affected than the other emotions**.
- Posemo and Negemo have a negative correlation. It follows that **Posemo and Negemo are the most negatively affected emotions compared to the others**. Furthermore, Posemo and Anger have a weak negative correlation. This result reflects what we expect from a dictionary like LIWC, where opposite sentiments like Posemo and Negemo have different words.
- Negemo is positively correlated with Anx, Anger, and Sad. So Negemo carries with it feelings of Anx, but especially Sad and Anger. However, this is a structural feature of LIWC, as some words are shared among all four of these emotions.
- The number of kudos is associated with text length and number of comments. As a result, **the works that received the most appreciation are those with the most comments and words in response**.
- The average content length is strongly negatively correlated to Posemo. It is confirmed that **as the average content length increases, Posemo is negatively affected**.
- Works with more Posemo tend to have less Negemo and Anger.

Figure 6: Correlation matrix between emotions and more (by work)

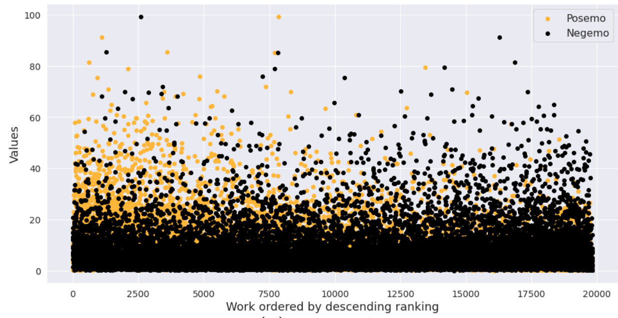


4.3 How to combine the number of kudos, comments and emotions?

When a work has a lot of comments, we expect that it has to do with a different kind of popularity maybe from a work with few comments. Is a work with high number of kudos liked? Do the most positive works have fewer comments than kudos and the most negative works have more comments than kudos? We see an answer to this question with Figure 7, where the works are arranged on the horizontal axis in decreasing values of Posemo and Negemo, and on the vertical axis the values of the ratio $\frac{number\ of\ kudos}{number\ of\ comments}$ are displayed for each work.

While for Warriors, Twilight, The Lord of the Rings, and Percy Jackson we do not have a clear finding, partly because of the smaller number of works, we do find confirmation for Sherlock Holmes, Harry Potter, and Marvel. In fact, in the latter, looking at the left margin of the graphs, we can see that the yellow dots are totally above the black ones, and therefore the top positive works have a higher ratio, which means that the number of kudos greatly exceeds the number of comments.

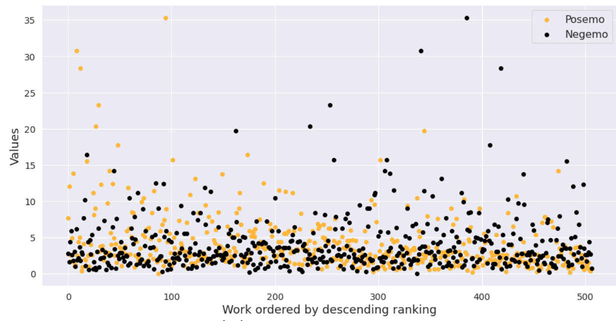
Figure 7: Ratio of number of kudos to number of comments in works ordered decreasingly by values of Posemo and Negemo



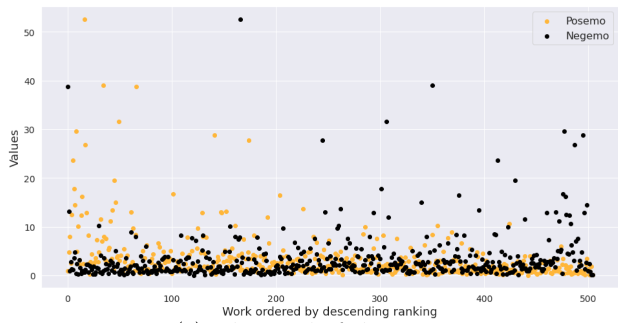
(a) Marvel

(b) Harry Potter

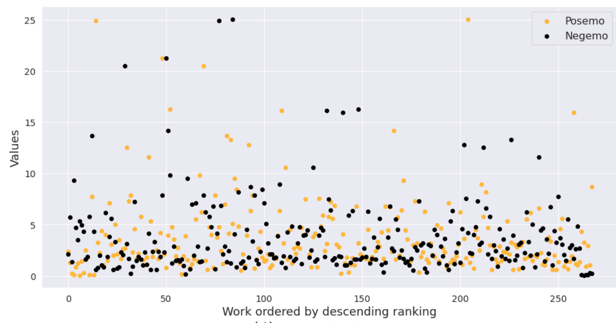
(c) Sherlock Holmes



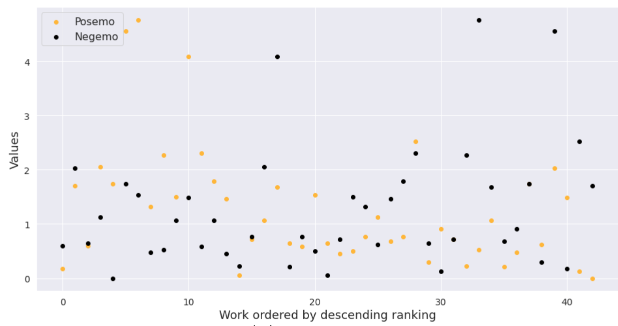
(d) Percy Jackson



(e) The Lord of the Rings



(f) Twilight



(g) Warriors

So yes, it is confirmed that the most positive works have fewer comments than kudos and the most negative works have more comments than kudos.

4.4 Prominence of the author

This section discusses the importance of authors in fanfiction communities. Attention is thus focused on them, in particular their behaviour is compared with that of a normal non-author user. Herein lies the novelty in the analysis conducted in this research paper, and it allows for the revelation of important findings, also from the perspective of evaluating the work and the community more in general.

We will shortly analyze the importance of the author from three perspectives, to foster different counter analyses and uncover patterns that are likewise difficult to capture. In Subsection 4.4.1 we will first take a look between the *Authors vs. non authors* comparison, where simply by author we mean a user who has published at least one work and by non authors those who have never published a work and thus have only commented on the platform. We then go into detail in Subsection 4.4.2, where we compare *Authors on their works vs. others*. Here we consider on the one hand only comments by an author related to their own works, while on the other hand we group all the remaining comments, regardless of whether they are by authors or not. So it is as if under each work we distinguish the comments of the author of that work from the rest. Finally in Subsection 4.4.3 with the comparison *Authors on their works vs. authors on other's works*, we consider only authors. In particular we distinguish comments under their works from those made to others' works. This somewhat anticipates the "How does an author act when they are and where they are not?" discussion.

In overall, the results shown in this section are not dependent on community size and occur in the same way and with the same intensity in all seven communities.

4.4.1 Authors vs. non authors

In general, this comparison shows that:

- The average content length of authors is shorter than that of non-authors in large communities, while it is the opposite in small communities.
- Non authors outperform authors relatively to Sad, Anger and Negemo, while authors respond with more Anx and Posemo.
- Authors always write at least 50% of the total words compared to non-authors, and in larger communities such as Sherlock Holmes, Harry Potter and Marvel they even outnumber them, although they are just over 10% of community users.

An example of this behaviour is shown in Figure 8, for the Marvel community.

4.4.2 Authors on their works vs. others

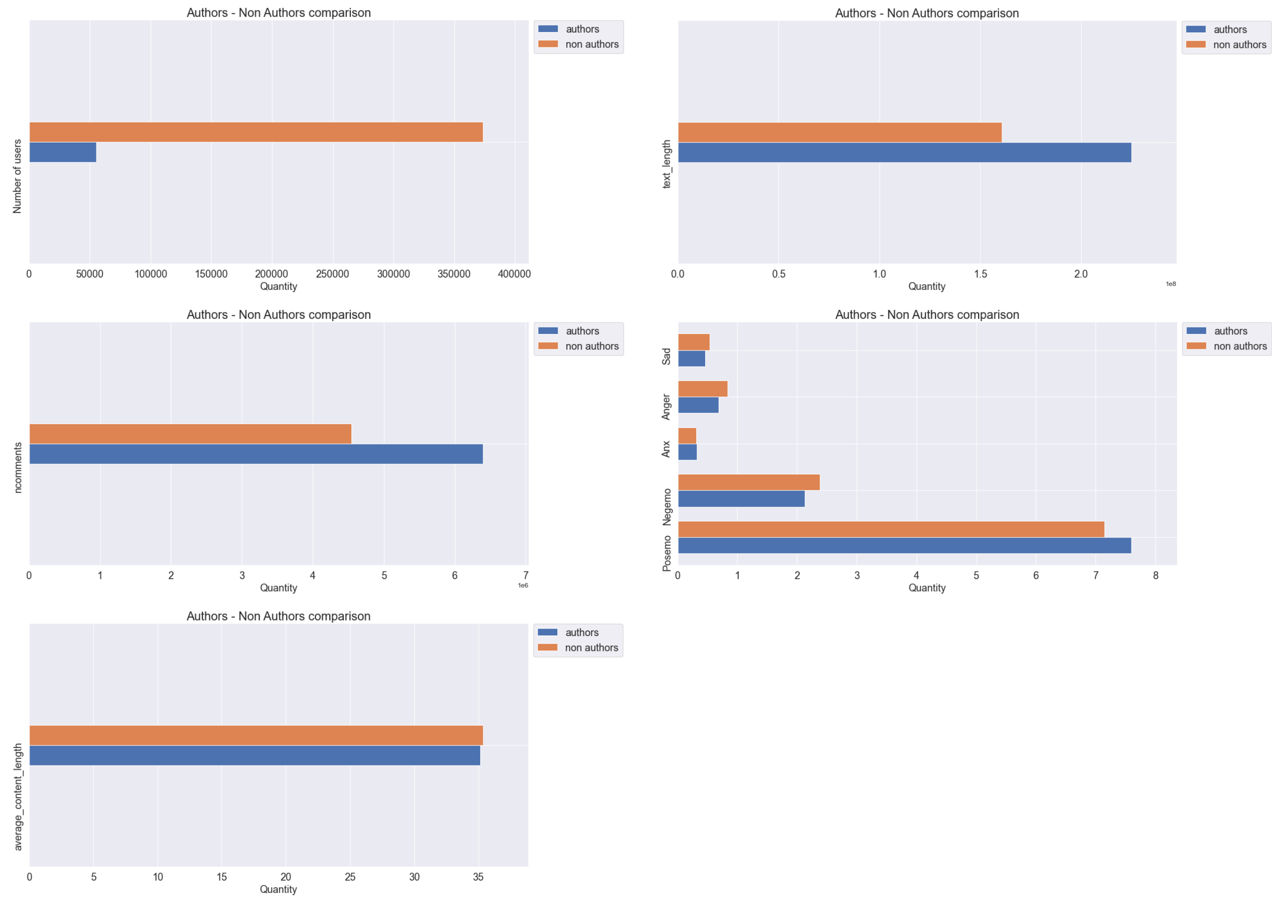
A general overview

This counter analysis is very interesting and shows us how:

- Most of the longer comments under their works are not from the authors.
- Most of the text and comments under works are not from the author, but from the others.
- **Authors use less Sad, Anger, Anx, Negemo, and much more Posemo under their work than others**.

An example of this behaviour is shown in Figure 9, for the Marvel community.

Figure 8: Comparison of authors and non authors for the Marvel community



Correlation matrix

Before seeing the results, it is necessary to keep in mind that almost all threads are started by other users and closed by the author of the work. The main observations are as follows, as illustrated in Figure 10:

- The number of words and comments that the author writes are associated with the number of words and comments that others write. It would seem reasonable to think that it is **the author** who **is active in the threads and responds to almost all the others' comments**.
- When others respond with longer comments on average, though with less Posemo, the author writes longer comments on average, that tend to have less Posemo.
- **The Posemo of the author seems to be positively associated with the Posemo of the others**, though the others write shorter comments on average.
- **The Negemo of the author seems to be negatively associated with the Posemo of the others**, which express more feelings like Negemo, Anger and Sad.
- **The Anx of the author seems to be positively associated with the Anx of the others**.
- **The author's Anger seems to be associated with an increase in Negemo and Anger, and a decrease in Posemo of others**.
- **The Sad of the author seems to be positively associated with the Negemo and Sad of the others**.

Figure 9: Comparison of authors on their works and others for the Marvel community

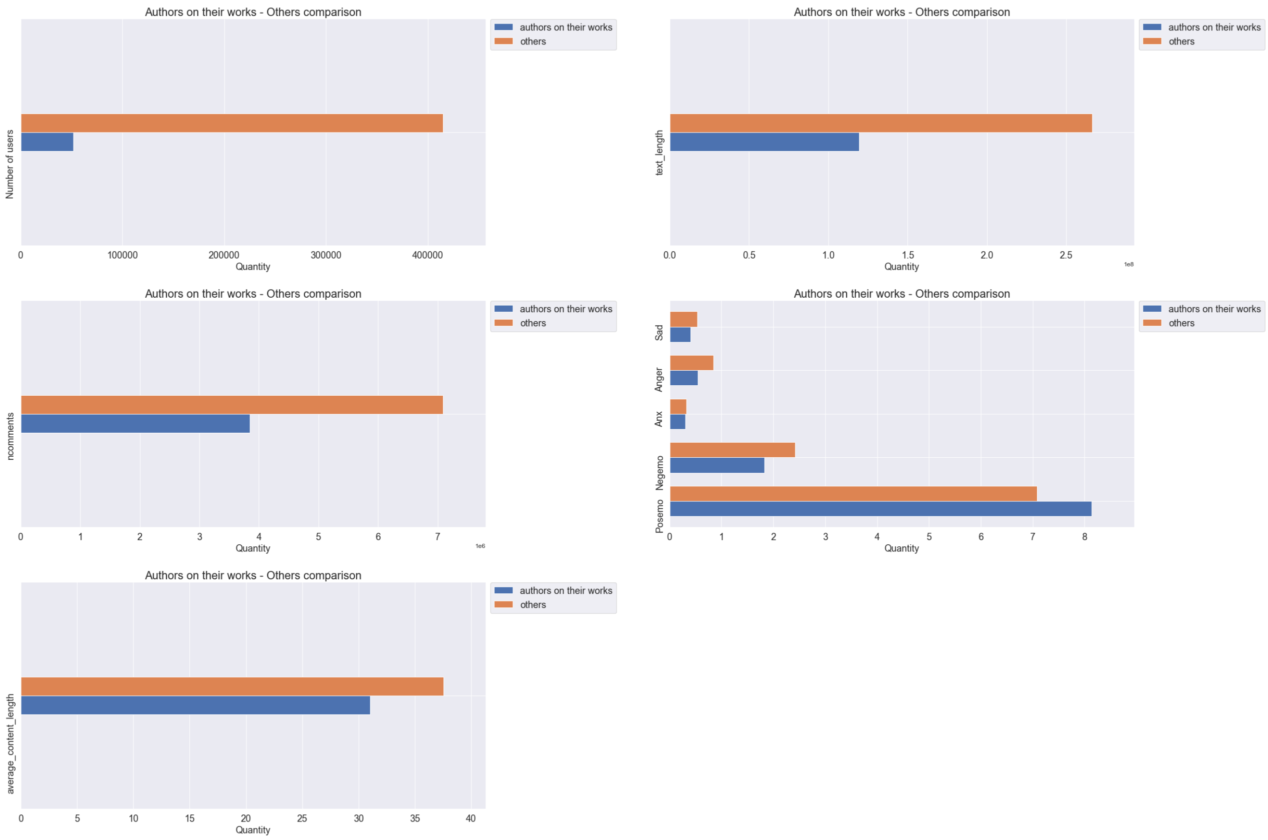


Figure 10: Correlation matrix between emotions and more in authors on their works vs. others (by work)

	Author text_length	Author ncomments	Author average_content_length	Author Posemo	Author Negemo	Author Anx	Author Anger	Author Sad	Others text_length	Others ncomments	Others average_content_length	Others Posemo	Others Negemo	Others Anx	Others Anger	Others Sad
Author text_length	1	0.81	0.15	-0.099	0.059	0.033	0.061	-0.0044	0.78	0.61	0.15	-0.12	0.04	0.046	0.034	-0.0082
Author ncomments	0.81	1	-0.036	-0.024	0.048	0.03	0.049	-0.0057	0.77	0.76	0.028	-0.082	0.047	0.044	0.046	-0.0085
Author average_content_length	0.15	-0.036	1	-0.41	0.089	0.029	0.11	-0.02	0.062	-0.026	0.55	-0.26	0.0097	0.037	0.0025	-0.0041
Author Posemo	-0.099	-0.024	-0.41	1	-0.21	-0.09	-0.17	-0.054	-0.083	-0.065	-0.2	0.43	-0.027	-0.019	-0.031	0.016
Author Negemo	0.059	0.048	0.089	-0.21	1	0.41	0.56	0.57	0.059	0.053	0.057	-0.15	0.34	0.11	0.22	0.21
Author Anx	0.033	0.03	0.029	-0.09	0.41	1	0.064	0.033	0.036	0.034	0.029	-0.032	0.095	0.17	0.034	0.018
Author Anger	0.061	0.049	0.11	-0.17	0.56	0.064	1	0.025	0.06	0.051	0.07	-0.13	0.2	0.035	0.27	0.015
Author Sad	-0.0044	-0.0057	-0.02	-0.054	0.57	0.033	0.025	1	-0.0042	-0.0037	-0.015	-0.051	0.25	0.012	0.061	0.34
Others text_length	0.78	0.77	0.062	-0.083	0.059	0.036	0.06	-0.0042	1	0.91	0.14	-0.13	0.056	0.053	0.053	-0.0055
Others ncomments	0.61	0.76	-0.026	-0.065	0.053	0.034	0.051	-0.0037	0.91	1	0.0058	-0.091	0.055	0.044	0.058	-0.0089
Others average_content_length	0.15	0.028	0.55	-0.2	0.057	0.029	0.07	-0.015	0.14	0.0058	1	-0.37	0.042	0.073	0.017	0.008
Others Posemo	-0.12	-0.082	-0.26	0.43	-0.15	-0.032	-0.13	-0.051	-0.13	-0.091	-0.37	1	-0.18	-0.085	-0.13	-0.074
Others Negemo	0.04	0.047	0.0097	-0.027	0.34	0.095	0.2	0.25	0.056	0.055	0.042	-0.18	1	0.35	0.62	0.6
Others Anx	0.046	0.044	0.037	-0.019	0.11	0.17	0.035	0.012	0.053	0.044	0.073	-0.085	0.35	1	0.072	0.0077
Others Anger	0.034	0.046	0.0025	-0.031	0.22	0.034	0.27	0.061	0.053	0.058	0.017	-0.13	0.62	0.072	1	0.062
Others Sad	-0.0082	-0.0085	-0.0041	0.016	0.21	0.018	0.015	0.34	-0.0055	-0.0089	0.008	-0.074	0.6	0.0077	0.062	1

4.4.3 Authors on their works vs. authors on other's works

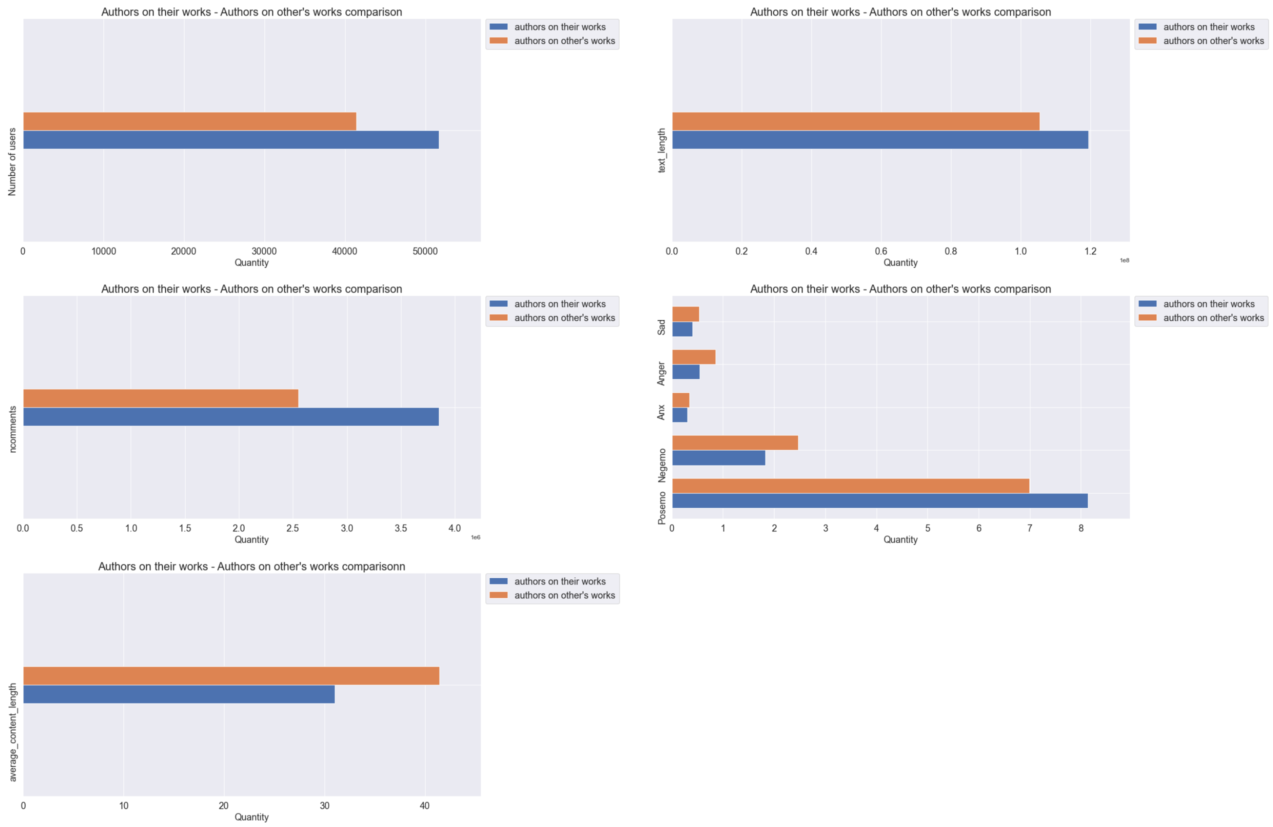
A general overview

Below are the most surprising results:

- Authors, with respect to their own works, write more and with more comments but with a shorter average content length than when they write to others' work. This means they have more to say to others than they do to themselves.
- Shockingly, we find that **authors are more critical of other people's work**. Their words under the works of others reflect feelings with more Sad, Anger, Anx, and with a good increase of Negemo, but especially much less Posemo! This is good evidence because there is a huge gap of words written between authors on their works and authors on other's works, and then we would expect to find more Posemo when authors comments to works that are not their own.

An example of this behaviour is shown in Figure 11, for the Marvel community.

Figure 11: Comparison of authors on their works and authors on other's works for the Marvel community



Correlation matrix

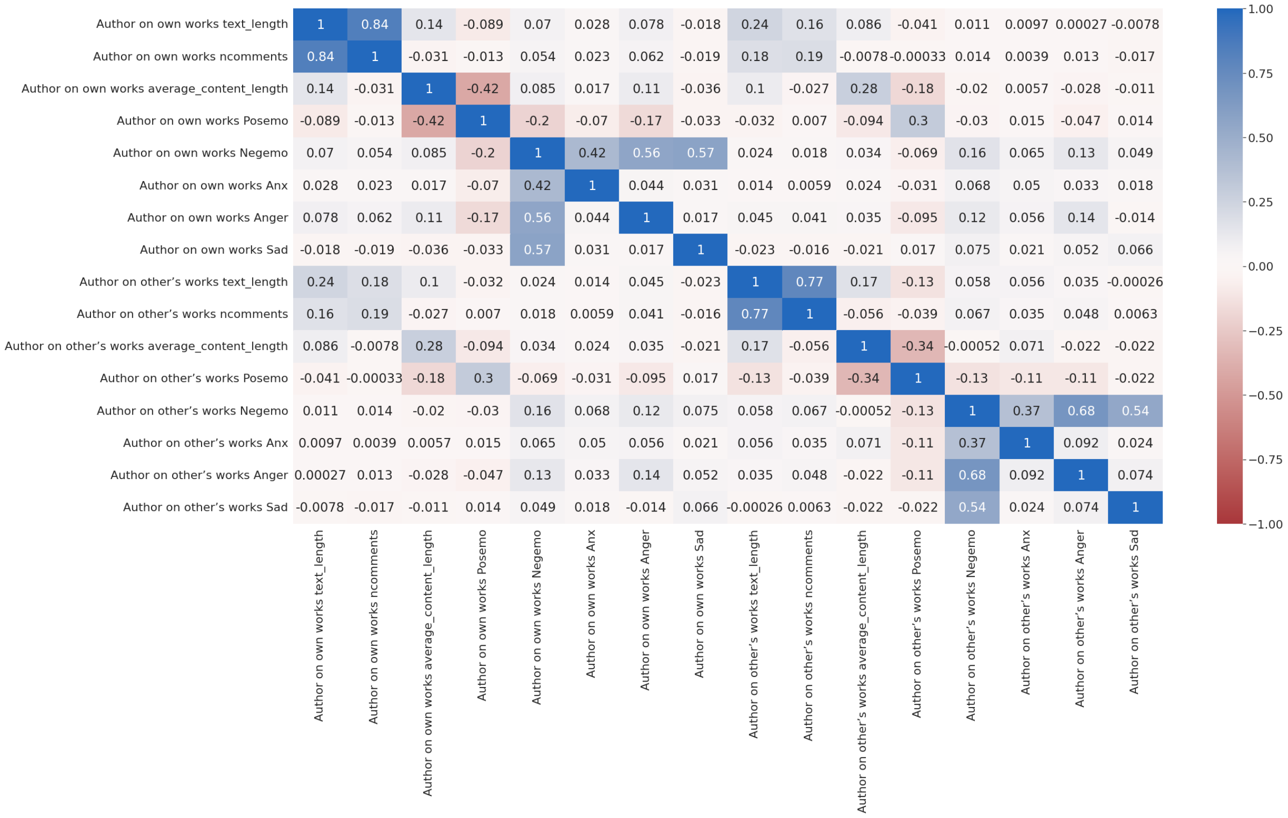
The main observations are as follows, as illustrated in Figure 12:

- When authors write more text and comments under their own works, they also tend to be more present in the works of others.
- **When authors write longer comments on average below their works, they contain less Posemo, but more Anger**. At the same time, they also write longer comments on average in their non-author activity, though with less Posemo.

- **The Posemo of the authors under their works seems to be positively associated with the Posemo of their non-author activity**. In addition, authors' comments with more Posemo under their work tend to have less Negemo and Anger.

- An interesting last point to consider is that **the authors' Negemo and Anger under their works seems to be positively associated with the Negemo and Anger expressed in their non-author activity**.

Figure 12: Correlation matrix between emotions and more in authors on their works vs. authors on other's works (by user)



4.5 Comparison of the most used Posemo and Negemo words

To conclude the analysis of authors' behaviour of when they comment under their own works versus when they comment on others' works, we now analyze the most frequently used words in the Posemo and Negemo categories. With Section 4.5.1 we have insight into the standard behaviour between author and others under a work. We oppose this with the analysis of Section 4.5.2, to highlight different behaviours of the authors and see if they deviate from the standard behaviour of a commenting user. Will they be more critical of other people's works than a standard commenter? How will the authors relate to users who comment on one of their works? Will they be polite? Let us go and find out.

4.5.1 Author on own works vs. others

Posemo

As shown in Table 4, the author comments in response to others using words of **thanks** and **courtesy**, such as *glad* and *thank*. Others, on the other hand, looking at Table 5, show **appreciation** for what was read and to **compliment** the author on the work done, with words such as *love*, *good*, *great*, *amazing*, and *beautiful*. Remember that the frequency field in the tables simply indicates the number of times that word has been counted in the comments.

Table 4: Ranking of the most frequently used Posemo words by authors under their works

	word	**categories**	**frequency**
1	glad	Adj Affect Posemo	1 256 585
2	thank	Verb Affect Posemo FocusPresent	942 333
3	love	Affect Posemo Social Bio Drives Affiliation	626 673
4	enjoyed	Affect Posemo	504 877
5	liked	Verb Affect Posemo FocusPast	473 990
6	happy	Adj Affect Posemo	435 633
7	good	Affect Posemo Drives Reward	340 083
8	hope	Verb Affect Posemo CogProc Discrep Tentat Focu...	340 079
9	well	Function Adverb Affect Posemo Informal Nonflu	322 267
10	kind	Affect Posemo	273 159

Table 5: Ranking of the most frequently used Posemo words by others

	word	**categories**	**frequency**
1	love	Affect Posemo Social Bio Drives Affiliation	146 895
2	good	Affect Posemo Drives Reward	56 529
3	well	Function Adverb Affect Posemo Informal Nonflu	38 627
4	great	Adj Affect Posemo Drives Reward	34 606
5	loved	Affect Posemo Social Bio Drives Affiliation	34 226
6	cute	Adj Affect Posemo	27 824
7	hope	Verb Affect Posemo CogProc Discrep Tentat Focu...	26 605
8	amazing	Adj Affect Posemo	25 951
9	sweet	Adj Affect Posemo Percept Bio Ingest	23 991
10	happy	Adj Affect Posemo	19 723

Negemo

As shown in Table 6, the author responds with words of **misunderstanding** and to **apologize**, such as *sorry*, *worry*, *sad*, and *wrong*, although there is no shortage of **criticism** with words such as *poor*, *weird*, and *hate*. Others, on the other hand, as observable from Table 7, use words of **appreciation** such as might be *sad*, *cry* and *crying*, and of **criticism** such as *poor*, *wrong* and *shit*.

Table 6: Ranking of the most frequently used Negemo words by authors under their works

	word	**categories**	**frequency**
1	sorry	Adj Affect Negemo Sad	173 597
2	bad	Affect Negemo Drives Risk	103 144
3	worry	Verb Affect Negemo Anx FocusPresent	79 329
4	poor	Adj Affect Negemo Drives Power Money	55 780
5	wrong	Adj Affect Negemo Drives Risk	50 126
6	sad	Adj Affect Negemo Sad	49 471
7	hurt	Affect Negemo Sad Percept Feel	40 973
8	shit	Affect Negemo Anger Bio Body Informal Swear	40 969
9	hate	Verb Affect Negemo Anger FocusPresent	40 811
10	weird	Adj Affect Negemo	37 018

Table 7: Ranking of the most frequently used Negemo words by others

	word	categories	frequency
1	sad	Adj Affect Negemo Sad	11 010
2	bad	Affect Negemo Drives Risk	9 384
3	poor	Adj Affect Negemo Drives Power Money	7 768
4	sorry	Adj Affect Negemo Sad	7 193
5	shit	Affect Negemo Anger Bio Body Informal Swear	5 626
6	fucking	Adj Affect Negemo Anger Bio Sexual Informal Swear	5 458
7	hurt	Affect Negemo Sad Percept Feel	5 395
8	cry	Verb Affect Negemo Sad	5 243
9	wrong	Adj Affect Negemo Drives Risk	5 177
10	crying	Verb Affect Negemo Sad	5 005

4.5.2 Author on own works vs. author on other's works

Posemo

The words used by the authors under their works are the same as those just analyzed from Table 4. On the other hand, as far as words directed toward others' works are concerned, as can be observed from Table 8, there are no differences with Table 5, which lists Posemo's most frequently used words by others. Thus it can be said that the authors behave like others and appreciate in the same way, so they have a standard behaviour of a usual commentator. The fact that they have written their own works does not lead them to be jealous of their creations, and they appreciate the work of others without any worries.

Table 8: Ranking of the most frequently used Posemo words by authors on other's works

	word	categories	frequency
1	love	Affect Posemo Social Bio Drives Affiliation	1 318 838
2	good	Affect Posemo Drives Reward	438 015
3	well	Function Adverb Affect Posemo Informal Nonflu	330 482
4	loved	Affect Posemo Social Bio Drives Affiliation	311 233
5	great	Adj Affect Posemo Drives Reward	253 640
6	amazing	Adj Affect Posemo	210 062
7	sweet	Adj Affect Posemo Percept Bio Ingest	187 565
8	hope	Verb Affect Posemo CogProc Discrep Tentat Focu...	182 749
9	perfect	Adj Affect Posemo CogProc Certain Drives Rewar...	178 782
10	happy	Adj Affect Posemo	178 560

Negemo

The words used by the authors under their works are the same as those just analyzed from Table 6. Here we notice much more criticism on the part of the author, compared to a usual commentator. This can be seen by placing the Tables 7 and 9 side by side, where we find the word *poor* in first place, a clear indicator of deep criticism, words like *fucking* and *hurt* are more frequent, words like *wrong* and *fuck* appear, and others like *sad* come down. Thus we can say that **authors comment to others' works with much more criticism and less appreciation than normal commentators**. In this case they deviate from standard behaviour, and show jealousy toward their own works.

Table 9: Ranking of the most frequently used Negemo words by authors on other's works

	word	categories	frequency
1	poor	Adj Affect Negemo Drives Power Money	87 655
2	bad	Affect Negemo Drives Risk	82 692
3	sad	Adj Affect Negemo Sad	80 019
4	fucking	Adj Affect Negemo Anger Bio Sexual Informal Swear	77 424
5	shit	Affect Negemo Anger Bio Body Informal Swear	66 468
6	hurt	Affect Negemo Sad Percept Feel	57 382
7	sorry	Adj Affect Negemo Sad	53 291
8	damn	Affect Negemo Anger Informal Swear	52 019
9	fuck	Verb Affect Negemo Anger Bio Sexual FocusPrese...	47 597
10	wrong	Adj Affect Negemo Drives Risk	46 362

4.6 Interweaving with structural metrics of comment threads

In this last section, an attempt is made to interweave the trend of emotions with structural metrics of the comment network under the works. In this way, it is straightforward to understand how emotions vary, for example, according to the level of depth in the comment thread, and to the width or length of the threads. The following results are valid for all communities and an example has been reported in Figure 13 for the Marvel community. Recall that the values of the analyzed metrics are arranged in a descending fashion on the horizontal axis and analyzed by works. In correspondence on the vertical axis, the values for the emotions Posemo in yellow and Negemo in black are drawn, again for each work. It emerges that:

- The most liked works, that are the ones with more Posemo, have few comments compared to those with more negative emotions (Subfigure (a)).

- It has been observed that the depth level with the most comments is always the first one, that is, comments that are in direct response to the work. Then **works that have more comments in direct response to the work itself**, meaning greater breadth, **have lower Posemo values**. Vice versa for Negemo (Subfigure (b)).

- Works with more Posemo have on average fewer comments per user than works with more Negemo (Subfigure (c)).

- The works with more Posemo are those visited by fewer users, unlike those with more Negemo (Subfigure (d)).

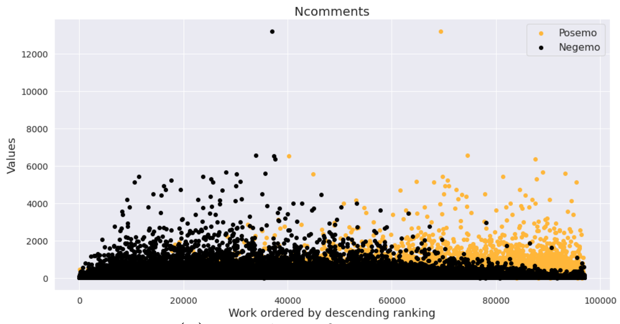
- Works with lower conversation tree height [1] are more positive, vice versa for Negemo (Subfigure (e)).

The second point triggers a deepening and suggests comparing the five emotions analyzed so far, contrasting the first level comments of the others, all except the author, with those of the other levels, again of the others. It involves seeing the emotions in comments that are at different levels in the comment threads.

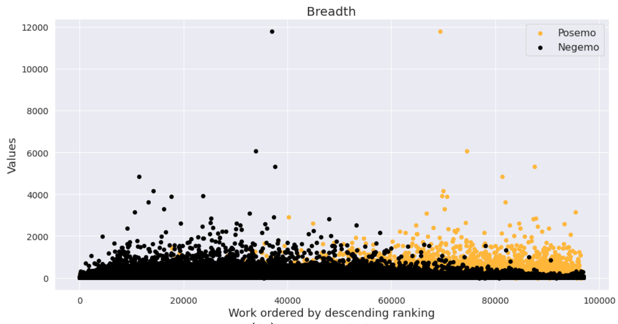
From Figure 14, taken from the Marvel community for example, it can be seen that comments in direct response to the work, that is, those at level one, manifest more emotions such as Sad, Anger, and Negemo than comments at higher depths in the threads. Feelings such as Posemo, on the other hand, are more present in comments in direct response to the work, perhaps synonymous with users leaving a comment to appreciate the work. These considerations apply to all seven communities.

[1] conversation tree height: is the length of the deepest conversation

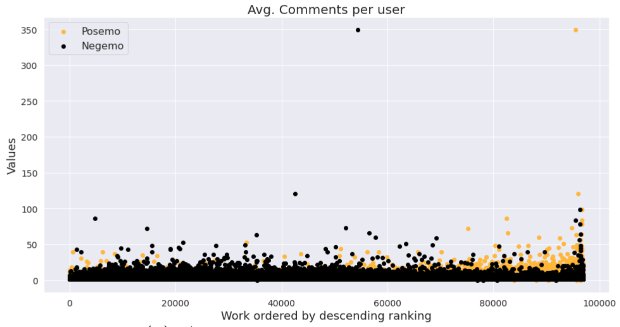
Figure 13: Posemo and Negemo variations in works ordered by descending ranking of field analyzed (Marvel community)



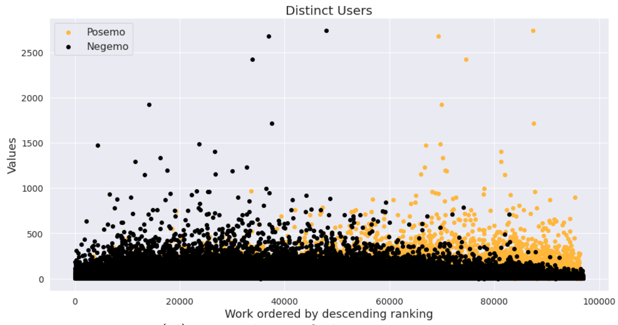
(a) Number of comments



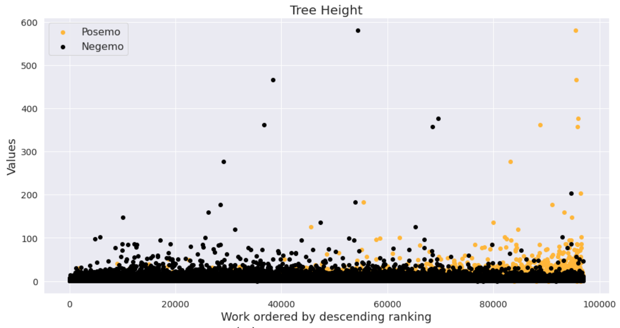
(b) Breadth



(c) Average comments per user

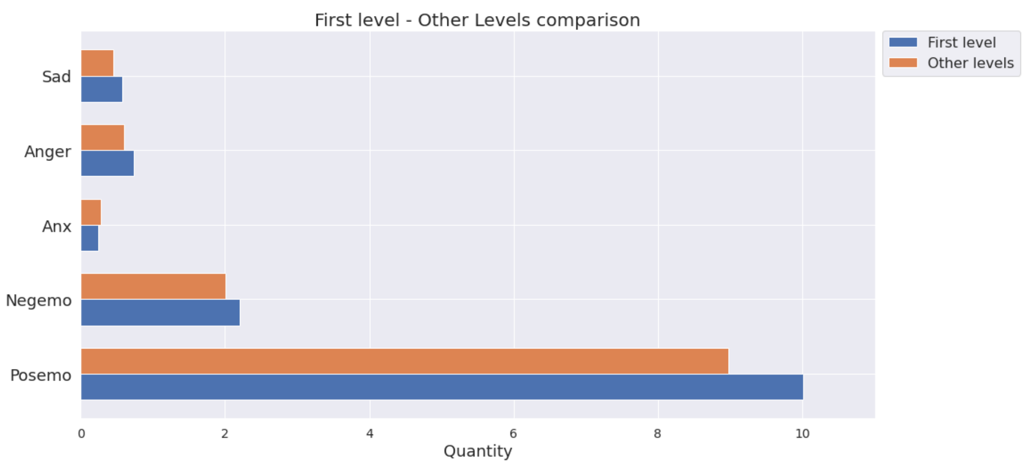


(d) Number of distinct users



(e) Tree height

Figure 14: First level vs. other levels comparison (Marvel Community)



5 Conclusions

The huge amount of content available on Archive of Our Own makes it hard to collect and process such a quantity of data. This research started as an exploratory analysis on a subset of the data and it was aimed to uncover patterns in the prosumers' behaviour that can potentially be extended to the rest of the communities as well. Assuming that the same recurring patterns were observed in most of the sampled fandoms - and the selected fandoms were different enough - we could use this as an argument for generalizing our observations to the whole platform.

We first understood how emotions relate in a fanfiction community context. Posemo and Negemo are the emotions that most negatively affect each other and thus give a sharp edge to the evaluation of a work. Posemo, on the other hand, is the only emotion affected by the length of comments. On longer ones, in fact, it decreases dramatically. Regarding the appreciation of a work, described by the number of kudos, it turned out that it is the works with more comments, and therefore more words in response, that are appreciated. Although the most positive works, meaning a greater number of Posemo, are those that have fewer comments than kudos, while the most negative works are those that have more comments than kudos.

Focusing then on the author allowed us to guess deeper aspects and see these communities from a different perspective. The author plays a key role in comments under their works and always participates, generating most of the content. When responding to comments under their work they are biased, and use less Sad, Anger, Anx, Negemo, and much more Posemo than other users who comment on them do. However, when it comes to commenting to works that are not theirs, the authors become much more critical. Their words reflect feelings with more Sad, Anger, Anx, and a good increase in negative words, but mostly they are much less positive. The fact that they use much less Posemo is strongly confirmed by the fact that in their non-author activity they write much less, and thus one would have expected a higher percentage of Posemo, since the latter is affected by word count, and is higher on shorter comments and thus when there are fewer words involved. Authors' attitudes are again confirmed by the fact that under works that are not theirs they use words that express more criticism more frequently, and new words emerge that are equally negative.

The last analysis, on the comparison of emotions with the structural metrics of the network of comments under a work revealed as many important truths. We saw how the works that have the most comments in direct response to the work itself are those with sentiments of greater Negemo. But this does not mean that the first level in the comment tree under a work, that is, comments in direct response to the work, has less Posemo. In fact compared to deeper levels it has more Posemo, but in contrast it has in greater numbers sentiments such as Sad, Anger and Negemo, though with less variation.

The analysis centers on the ultimate goal of delineating these new content creation communities from an emotion and author perspective. The results obtained identified specific and general aspects of fanfiction communities, shedding light on hitherto obfuscated aspects.

Bibliography

[1] Archive of our own, wikipedia. https://en.wikipedia.org/wiki/Archive_of_Our_Own. Last accessed on 25/05/2022.

[2] Liwc. https://www.liwc.app/help/howitworks/. Last accessed on 30/05/2022.

[3] Liwc python package. https://github.com/chbrown/liwc-python/. Last accessed on 31/05/2022.

[4] Pearson correlation coefficient. https://realpython.com/numpy-scipy-pandas-correlation-python/#pearson-correlation-coefficient. Last accessed on 16/06/2022.

[5] Seraj S. Boyd R., Ashokkumar A. and Pennebaker James W. *The Development and Psychometric Properties of LIWC-22*. University of Texas at Austin, 02 2022.

[6] Jordan K. Pennebaker James W., Boyd R. and Blackburn K. *The development and psychometric properties of LIWC2015*. University of Texas at Austin, 2015.

[7] Rebora S. Pianzola F. and Lauer G. Wattpad as a resource for literary studies. quantitative and qualitative examples of the importance of digital social reading and readers' comments in the margins. *PLOS ONE*, 15(1):1–46, 01 2020.

[8] Tausczik Yla R. and Pennebaker James W. The psychological meaning of words: Liwc and computerized text analysis methods. *Journal of Language and Social Psychology*, 29(1):24–54, 2010.