What Makes a Good Fact Check? Insights from X's Community Notes

Levi Goldberg

Fall 2025

Abstract

This research project examines the effectiveness of different fact-checking formats on social media through a data analysis on X's Community Notes. Community Notes, a crowd-sourced fact-checking system, has shown promise in reducing engagement with misleading posts, but the inconsistent quality of these Notes necessitates investigation into the features of those that are successful. Building on existing research suggesting that effective fact-checks are positively phrased, concise, and created soon after the misleading post, this study analyzes these attributes in crowd-sourced fact-checking Notes on X ($n = 777,669$). These characteristics were then correlated with community-provided ratings of the Notes to assess which features most strongly predict perceived effectiveness. The analysis is broken down by the theme of the misinformation (e.g. political, entertainment, etc.) to examine the features of effective fact checks in different contexts.

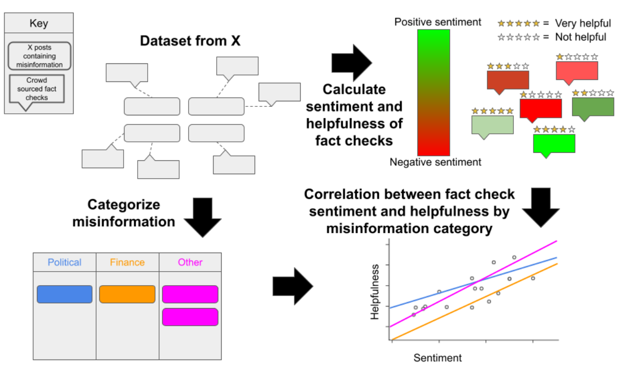

Figure 1: Graphical Abstract

1 Introduction

When the Community Notes program was launched on Twitter (now X) under the name of Birdwatch, its goal was to provide a transparent platform for misinformation detection and correction. If X users encountered a post that they thought contained misleading information, they could add credibility indicators to explain why the post was misleading and a context providing Note to attempt to correct the misinformation. These Notes would then be rated by other users to determine how beneficial they were as fact checks. All the data collected by X regarding Community Notes and their ratings are available for download on their website.

Since these Notes are provided by users, they may not be as effective at correcting misinformation as fact checks provided by professionals, but this system allows for greatly enhanced scalability and relies on "the wisdom of the crowds" - the notion that collective knowledge from a diverse group can exceed the knowledge of any individual [1]. On a platform like X with millions of daily posts, it would be impossible for professional fact checkers to review every one. The idea behind Community Notes is that a large group of individuals can review a much wider range of posts, while relying on their collective knowledge to create efficient fact checks.

Prior research in this problem space has found evidence that Community Notes are effective at detecting misinformation, but these Notes are not all equally effective at actually correcting the misinformation. Although Community Notes may provide the solution to fact checking at scale on social media, it remains unclear which types of Notes are the most effective at convincing misinformed users of the truth. Professional fact-checkers have years of experience to inform how they phrase a response to misinformation, but Community Note contributors have no such expertise. X provides their Notes contributors with some limited advice for the format of their fact checks, and I believe that a comprehensive study on the relationship between a Note's format and its effectiveness could provide much more useful, data driven insights for crowd-sourced fact checkers.

I propose that the phrasing, length, and timing of a Note has a direct relationship with that Note's effectiveness at correcting misinformation, as reflected in the crowd-sourced helpfulness rating of that Note. My hypothesis is that Notes that are more positively phrased, shorter, and created soon after the original misinformation will be rated as more helpful. To test this hypothesis, I analyzed the Community Notes data set, performed sentiment analysis on each Note to calculate a positivity score, and ran statistical tests to determine the influence of these factors on the success of a Note. To examine how the influence of these factors changes in different contexts, I categorized each Note by the theme of misinformation in the post it was responding and tested each category separately.

2 Related Works

2.1 Prior Research on Community Notes

The natural first step in evaluating the effectiveness of Community Notes is determining if they are able to reduce the spread of misinformation. A study by Slaughter et al. on the diffusion of misinformation on X determined that posts that received context-providing Notes had a significantly lower rate of engagement [2]. The researchers found that posts receiving a Note had an average of a 40% reduction in the number of comments and reposts. However, this reduction lessened significantly the longer between the creation of the post and the addition of the Note, suggesting that the sooner a fact check is created the more effective it will be.

This study also determined that there was a significant gap in the effectiveness of context-providing Notes on political versus non-political posts, with Notes on political posts being seen as more biased and less trustworthy. This suggests that the effectiveness of context-providing Notes may differ with the type of misinformation being fact checked.

The research thus far is extremely favorable towards the effectiveness of Community Notes at countering misinformation. However, it still remains to be seen if this style of countering misinformation can entirely replaced professional fact-checkers, which is precisely what Borenstein et al. attempted to determine [3]. By using an LLM to annotate the hundreds of thousands of Notes and the posts they were attached to in the Community Notes data set, they found that the highest-rated Notes and Notes responding to the most complex misinformation overwhelmingly included references to professional fact checking sources. This suggests that whether or not a professional is involved in fact checking a potentially misleading post, the research provided by these professionals is still crucial to successfully correcting misinformation. One limitation of this study is that the LLM used to annotate the dataset was unable to process X posts that contained non-text media, such as images or videos. This is a major problem for anyone working with the Community Notes data, including myself, since the scale of the dataset necessitates automating parts of the analysis.

Another flaw with Community Notes is its susceptibility to organized efforts to influence its content. This is, in part, because the algorithm behind determining which Notes would be displayed alongside a misleading post is open source, allowing groups of users to "game" the system. One possible solution to this is a new and more opaque algorithm, as posited by De et al. [4]. These researchers created a framework for AI-generated "Supernotes" that synthesize the content of existing Notes on a given post to provide a concise, accurate fact check. This system could still be gamed, but the algorithm behind Supernotes includes a simulated jury trained on the data of which Community Notes have been rated the most helpful over time to rate several different Supernote candidates and promote the one that is most likely to be rated the most helpful. De et al.'s analysis of the effectiveness of these Supernotes found that users rated them as more helpful than traditional Notes. Additionally, they determined that the most important aspect of the prompt used to generate the Supernotes was a precise description of the format it should follow. Since these Supernotes were consistently rated the most effective, this suggests that there exists a relationship between the format of a Note and its effectiveness. The primary drawback of this framework for improving Community Notes is that a Supernote cannot be generated until several notes have already been written by human users. Furthermore, another deficiency of the entire Community Notes platform is that even though it is more scalable than professional fact checking, there are simply not enough active contributors to review every single post on X.

2.2 The Format of Fact Checks

Having established that Community Notes are generally effective despite some limitations, I now explore potential improvements. Notes include quick-to-provide credibility indicators and a more time-consuming context statement. If credibility indicators alone proved effective in countering the spread of misinformation, the platform's scalability could improve. The previously discussed research from Drolsbach et al. suggested that credibility indicators on their own were not as effective as a context-providing statement, which was further confirmed in a study by Lu et al. [5]. For this project, the researchers used AI to attach credibility indicators to misleading social media posts to study how they would affect the diffusion of the post. They found that although these indicators helped some users to identify misinformation, they did little to lower the rate of engagement with or the spread of misleading information. There was also some evidence that credibility indicators are more effective at changing viewers' beliefs when there is also social influence (i.e., comments from other users stating the post is misleading) present, which further supports the notion that a combination of credibility indicators and context-providing responses is the most efficient way of countering misinformation. This conclusion was also supported by Ecker et al., who investigated whether credibility indicators on their own could backfire and cause misinformation to spread faster, which they found no evidence for [6]. Their research also suggested that short fact checks that succinctly explained why the information is incorrect were more effective at countering misinformation than longer, more in-depth explanations.

Further regarding the format of these context statements is a paper by Burel et al. in which, rather than attaching a Note to misleading posts on X, the authors designed a bot to message the post's creator to research how different types of responses would be received by the people spreading misinformation [7]. They found that spreaders of misinformation were most likely to respond positively to being fact checked if the statement was phrased politely.

One format of responding to misinformation used by previously cited papers, including Burel et al. and Ecker et al., is a narrative fact check, in which a story is told to explain why the information is misleading. There was some belief that these may be more effective than non-narrative fact checks, due to the way a narrative format can enhance comprehension and retention. However, a different study by Ecker et al. determined that when the information provided in a narrative and non-narrative refutation has minimal differences, there is no significant difference in the refutation's effectiveness at countering misinformation [8].

Thus, the research suggests that the most effective format for a fact check is a combination of credibility indicators and a refutation of the misinformation that is provided in a timely manner, concise, and positively phrased. This motivates my hypothesis that Notes with these attributes will receive higher helpfulness ratings.

3 Methods

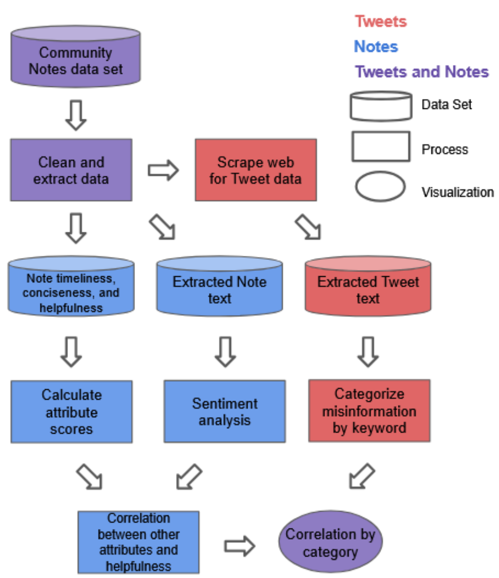

Figure 2: Data Architecture Diagram

3.1 Data Extraction

The open-source data from Community Notes is stored in tab-separated value (TSV) files. One of these TSV files, the "Notes file," contains data about each of the approximately 2 million Notes that have been created since the program was launched. The Notes file contains information about each Note like its ID, the ID of the post it was attached to, when it was created, and the text of the Note. Another twenty of these files, the "ratings files," contain data about each of the approximately 28.6 million Note ratings that have been created in this time. Each entry in the ratings files contains the ID of the Note that was rated and a number of binary indicators to label the Note as "helpful" or "unhelpful."

The Note ID was directly associated with the text of each Note in the data set, but the text of each post was not so easy to find. The post IDs contained within the data set can be used to find a JSON webpage containing the text and time of creation of each post. I used the `requests` library [9] in Python to scrape through these JSON pages and extract these attributes for each post that received a Note.

3.2 Processing the Posts that Received Notes

While using the `requests` library to extract the text of each post that received a Note, I removed any posts (and thus any Notes attached to that post) consisting of mostly non-text media, such as photos and videos, or non-English text from the dataset using the `langdetect` library [10]. This was because my method of categorizing the type of misinformation in a post used a dictionary to associate themes with keywords and then scanned through the text of each post and labeled it with the themes corresponding to any keywords that appeared. Posts containing only photos or videos and posts in other languages could not be categorized by this method.

The theme categories I used were politics, international, entertainment, finance, science/technology, health/wellness, environment, and culture. The keywords associated with these themes were chosen to be unique to each theme, although if multiple keywords were identified, a post could be labeled as belonging to multiple categories.

The final output of these processes was a "Tweets database" file where each line contains the ID of a post, the time it was created, the number of Notes that were attached to it, its text, and the categories assigned based on its text.

A pseudo code style summary of these processes is provided below.

```
For line in Notes File
   Scrape web for Tweet text and timeliness
   If Tweet text is valid
      Categorize Tweet text
      Extract credibility indicators
      Add entry to Tweets database
   Else
      Ignore Tweet
```

3.3 Processing the Notes

Once all of the non-text posts and their associated Notes had been removed, the remaining Notes needed to be cleaned to remove any special characters, punctuation, and URLs to prepare them for sentiment analysis. In Python, the `unicodedata` library [11] was used to remove special fonts, the `emoji` library [12] was used to remove emojis, the `re` library [13] was used to remove URLs, and the `langdetect` library [10] was used to ensure Notes were in English. Removing punctuation and standardizing the text as lowercase was done using built in Python methods.

To analyze the sentiment of each note, I used the VADER (Valence Aware Dictionary and sEntiment Reasoner) sentiment tool [14]. This open-source tool is tuned specifically for sentiments expressed in social media, which makes it ideal for studying Community Notes on X. When a text string is inputted to VADER, it calculates the percentage of the text with positive sentiment, the percentage with negative sentiment, and the percentage with neutral sentiment. It also calculates a compound score, which is a normalized, weighted composite of the positive, negative, and neutral portions with additional consideration given to how the sentiment of each portion of the text affects the sentiment of the rest. The compound score is a number between -1 and 1, with values greater than 0.05 reflecting an overall positive sentiment, values less than -0.05 reflecting an overall negative sentiment, and values between -0.05 and 0.05 reflecting an overall neutral sentiment. For my use case, I focused only on the compound sentiment scores from VADER, which were calculated for each Note in the data set.

Next, the timeliness of each Note needed to be calculated by comparing its time of creation with the time of creation of the post it was responding to. Another easy metric to calculate was the length of each Note.

The final Note metric that needed to be determined was the crowd-sourced helpfulness score. Ratings for each Note are provided by other X users who can tag the Note with a number of different indicators, some of which label the post as helpful (e.g. helpfulInformative, helpfulGoodSources, helpfulUnbiasedLanguage, etc.) and some of which label the post as unhelpful (e.g. notHelpfulOffTopic, notHelpfulIrreleventSources, notHelpfulIncorrect, etc.). In total, there are nine labels that fall under the category of helpful and thirteen that are unhelpful. To compute an overall score for the post, I subtracted the proportion of unhelpful ratings from the proportion of helpful ratings for each Note. For Notes that received multiple ratings, their overall score was the average of the scores for each rating. This meant that Notes that had a higher number of unhelpful ratings than helpful ones would be given a negative score, and vice versa would be given a positive score.

The final output of these processes was a "Notes database" file where each line contains the ID of a Note, the ID of the post it was attached to, the number of ratings the Note received, its overall helpfulness score, its positivity rating, its text, and the specific helpfulness ratings it received.

A pseudo code style summary of these processes is provided below:

```
For line in Notes file
   Extract Note ID, text, and timeliness
   If Note text is valid
      Analyze sentiment of text
      Add entry to Notes database
   Else
      Ignore Note
For line in each Ratings file
   Extract Note ID and ratings
   Calculate overall helpfulness score
   If Note ID in Notes database
      Update Note entry
   Else
      Ignore rating
```

3.4 Evaluation

After finishing the data processing, I used a natural join between the Notes database and the Tweets database to match each Note entry with the entry of the post it was attached to. This allowed me to then split up the Notes based on the category of misinformation in each post. Next, I used R to calculate the correlation between each of the Note metrics (positivity score, length, and timeliness) with the Notes helpfulness score. These correlations were calculated on the aggregate of the Note data, as well as on each individual category. Using the aggregate data, I trained several regression models to predict the helpfulness of a Note based on its sentiment, timeliness, and conciseness. Finally, I produced a number of auxiliary data visualizations in R, such as the distribution of each Note metric.

3.5 Limitations

The first major limitation of this research is that it did not examine any non-text or non-English posts on X. This, in addition to removing Notes that did not receive any ratings and removing corrupted lines in the data set, narrowed the analysis from 871,996 posts with 2,205,806 corresponding Notes to 447,061 posts with 777,669 corresponding Notes. Non-text and non-English posts needed to be removed from the dataset since it would be impossible to categorize them based on keywords.

This method of keyword categorization is another limitation of my design. A post could fall within a given category, but if it doesn't contain any of the keywords associated with that category, then it won't be labeled as such. I tried to overcome this limitation by making an expansive keyword dictionary, but I had to be careful to choose only keywords that uniquely identified that category and would (almost) never appear in another context.

4 Results

4.1 Data Distributions

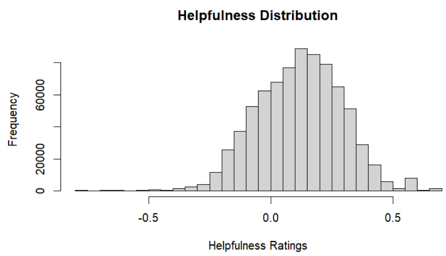

Figure 3: Distribution of helpfulness scores among all Notes

Figure 3 shows the distribution of helpfulness scores among all Notes in the data set. This distribution is approximately normal with a mean helpfulness score of 0.12, which means that a higher proportion of Notes are perceived as helpful, rather than unhelpful.

Figure 4: Distribution of positivity scores among all Notes

Figure 4 shows the distribution of positivity scores among all Notes in the data set. This distribution is clearly not normal and has a large peak at 0 and is relatively flat to both the left and the right. The mean positivity score was -0.08, which means that a higher proportion of Notes had negative sentiment than positive sentiment.

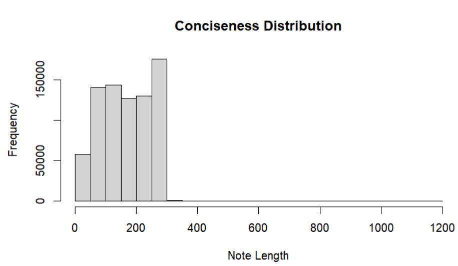

Figure 5: Distribution of conciseness among all Notes

Figure 5 shows the distribution of the lengths of each Note. This distribution, with a mean length of 167.7 characters, is also not normal. Note lengths are densely clustered between 0 and 400, with a few outliers beyond this range.

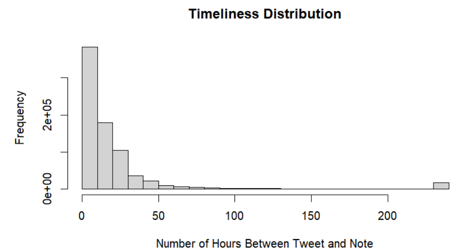

Figure 6: Distribution of timeliness among all Notes

Figure 6 shows the distribution of the number of hours between when the original post was made and when the Note was created. This distribution is also not normal. The mean number of hours was 120.2, which is about 5 days, although this is significantly impacted by outliers in the data. The median number of hours was only 10.3. For this visualization, all timeliness values greater than 240 hours (10 days) were set equal to 240 hours to make the scale of the histogram more digestible.

4.2 Aggregate Results

Note Attribute	Correlation with Note Helpfulness	p-value
Sentiment	0.0153	2.142×10^{-41}
Conciseness	0.0609	$< 2.2 \times 10^{-16}$
Timeliness	0.0198	3.287×10^{-68}

Figure 7: Correlations with helpfulness rating for Note sentiment, conciseness, and timeliness

The primary results of this research are shown in Figure 7. Since sentiment, conciseness, and timeliness are not normally distributed, I used Spearman's Rank-Correlation test, which is appropriate for non-normal data. Correlation values close to 1 denote a strong positive relationship between two variables, correlation values close to -1 denote a strong negative relationship, and correlation values close to 0 denote a weak relationship. Each of the Note attributes I investigated had a very weak, positive correlation with Note helpfulness. Although these correlations are weak, the p-values for each correlation are very close to 0. The p-value is the probability of observing the results in our data given that there is no correlation between the two attributes, so p-values close to 0 suggest that it is *very* unlikely that helpfulness shares no correlation with sentiment, conciseness, and timeliness.

The positivity of the correlation between Note sentiment and helpfulness suggests that positive phrasing is associated with increased helpfulness, which supports my hypothesis. However, the positivity of the correlations between conciseness and timeliness with helpfulness suggests that lengthier Notes and Notes with a longer delay after the original misinformation are associated with increased helpfulness ratings. This goes against my hypothesis that shorter, more concise Notes and more timely Notes with a shorter delay would be associated with increased helpfulness. Since conciseness had the highest correlation with helpfulness, this suggests that it has the largest impact on the helpfulness of a Note.

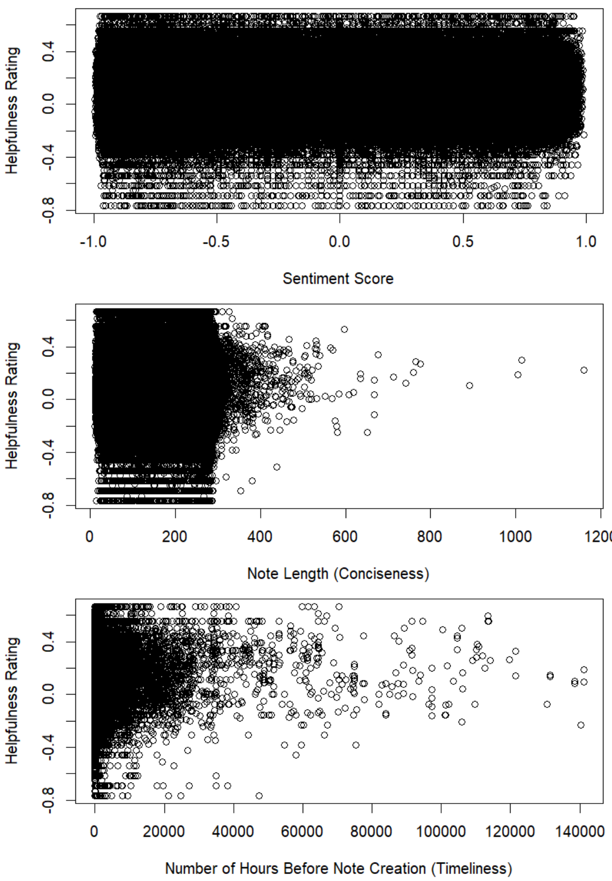

Figure 8: Note helpfulness vs sentiment, conciseness, and timeliness

Figure 8 shows the Note attributes of sentiment, conciseness, and timeliness graphed against helpfulness ratings. Each of these graphs has very high spread and no clear trend, which is likely the reason that each of the correlations are so weak

4.2.1 Multiple Linear Regression

Having established that the Note attributes of sentiment, conciseness, and timeliness share a weak, positive correlation with helpfulness ratings, I now wanted to see if I could predict the helpfulness rating of a Note based on its sentiment, conciseness, and timeliness. To accomplish this, I trained a multiple linear regression model on the 777,669 Notes in the data set. The results of this model are summarized in Figure 9.

Multiple Linear Regression Model to Predict Helpfulness Rating	
Adjusted R^2	0.0037
F-test p-value	$< 2.2 \times 10^{-16}$
Average residual	0.1399

Figure 9: Multiple linear regression results

The adjusted R^2 value means that only about 0.37% of the variation in Note helpfulness ratings can be explained by the influence of sentiment, conciseness, and timeliness. This weak performance is likely due to the high spread of data shown in Figure 8.

The F-test p-value is the probability that a model with no predictors would fit the data better than this model. Since this p-value is so low, I can conclude that using sentiment, conciseness, and timeliness provide a more effective predictive model than no predictors.

The average residual of 0.1399 means that the difference between the actual helpfulness ratings in the data set and the predicted helpfulness ratings using each Note's sentiment, conciseness, and timeliness was, on average, 0.1399. This is roughly the difference of one helpful or unhelpful binary indicator being selected by the Note rater, which means that although this model is insufficient to explain all of the variation in the data, it is able to make fairly accurate predictions of a Note's helpfulness rating.

4.2.2 Logistic Regression

Having established that a multiple linear regression model can make somewhat accurate predictions of a Note's helpfulness rating, I wanted to see if I could simply predict whether a Note would be overall rated as helpful or unhelpful. Notes with a helpfulness rating greater than or equal to 0 were labeled "Helpful" and Notes with a helpfulness rating less than 0 were labeled "Unhelpful." The logistic regression model was trained on 70% of the data (544,368 Notes) and tested on the remaining 30% of the data (233,301 Notes). The results are summarized in Figure 10.

Logistic Regression Model to Predict Helpfulness Category		
Accuracy	75%	
F1 Score	85%	
	Actually Helpful	Actually Unhelpful
Predicted Helpful	173,640	59,009
Predicted Unhelpful	405	247

Figure 10: Logistic regression results

The logistic model achieved an accuracy of 75% an F1 score of 85%. The accuracy is simply the number of correct predictions divided by the total number of predictions. The F1 score is a balance of the precision (proportion of predicted helpful ratings that were actually helpful) and the sensitivity (proportion of actual helpful ratings that were correctly predicted). These high scores seem to suggest that the logistic model performed very well. However, the model predicted nearly every one of the Notes to be helpful, and since approximately 75% of them were actually helpful, the accuracy of the model is misleading.

4.3 Categorical Results

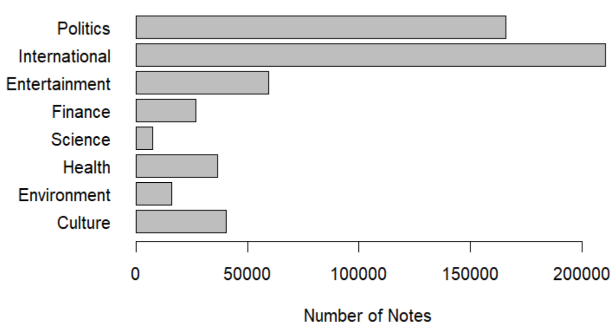

Figure 11: Distribution of Notes Among Misinformation Categories

Figure 11 shows the number of Notes that were in response to each category of misinformation. The type of misinformation receiving the most Notes was international followed by politics, entertainment, culture, health, finance, environment, and science. Using Spearman's Rank-Correlation test, I calculated the correlation with helpfulness ratings for sentiment, conciseness, and timeliness for each category, which is summarized in Figure 12.

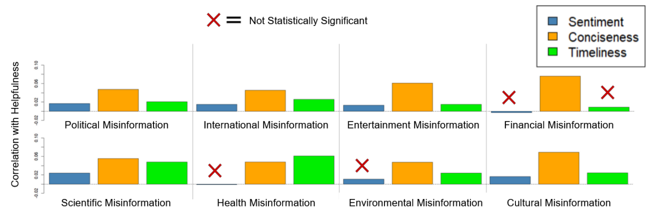

Figure 12: Note sentiment, conciseness, and timeliness correlation with helpfulness for each category

Each of these correlations were statistically significant at a significance level of 0.05, except for: sentiment to helpfulness and timeliness to helpfulness for Notes responding to financial misinformation, sentiment to helpfulness for Notes responding to health misinformation, and sentiment to helpfulness for Notes responding to environmental misinformation. For all categories except for health, the highest correlation was between conciseness and helpfulness. The general trend among all categories was a weak, positive relationship with helpfulness for all three Note attributes of interest. Since the categorical results are almost identical to the aggregate results, it seems like the relationship between Note sentiment, conciseness, and timeliness with Note helpfulness is approximately the same for each category of misinformation.

5 Future Work

A major limitation for this project was the removal of all non-text and non-English posts, since the category of misinformation for these posts could not be determined with my method of keyword identification. A comprehensive solution to this problem would be to use machine learning to train an AI model to categorize each post. This model would need to be able to determine the misinformation category of both textual posts and media posts. Since it would not rely on keyword identification for categorizing posts, it would likely create more accurate categorizations and leave fewer posts as uncategorized.

Another prominent direction for future work would be to analyze other Note attributes and determine their relationship with helpfulness. None of the Note attributes examined in this study had a strong correlation with helpfulness, but perhaps there are other attributes that do. One attribute of interest would be each Note's modality, which is the level of certainty expressed in the Note's text. There are sentiment analysis tools that could be used to calculate this. Another attribute could be the local time of day each Note was created. This would require tracking down data about each of the user IDs in the data set to find their local time zone.

6 Conclusion

Among all Notes, there was a weak, positive correlation with helpfulness for sentiment, conciseness, and timeliness. This refutes my hypothesis that sentiment and helpfulness would be positively correlated and the correlation with helpfulness for conciseness and timeliness would be negative. Of these three attributes, conciseness had the highest correlation with helpfulness, suggesting it has the most impact on the helpfulness of a Note. When broken down by the category of misinformation being responded to, these results did not change significantly, suggesting that these factors have the same influence on Note helpfulness, regardless of the context of misinformation. Together, sentiment, conciseness, and timeliness only explained about 0.37% of the variation in Note helpfulness, which suggests there are other Note attributes with a significant impact on helpfulness. Exploring other Note attributes is a key direction for future research.

7 Works Cited

[1] Surowiecki, J. (2004). *The wisdom of crowds: Why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations*. Doubleday.

[2] Slaughter, I., Peytavin, A., Ugander, J., & Saveski, M. (2025). Community notes reduce engagement with and diffusion of false information online. *Proceedings of the National Academy of Sciences*, *122*(38), e2503413122.

[3] Borenstein, N., Warren, G., Elliott, D., & Augenstein, I. (2025). Can community notes replace professional fact-checkers? *arXiv preprint arXiv:2502.14132*.

[4] De, S., Bakker, M. A., Baxter, J., & Saveski, M. (2024). Supernotes: Driving consensus in crowd-sourced fact-checking. *arXiv preprint arXiv:2411.06116*.

[5] Lu, Z., Li, P., Wang, W., & Yin, M. (2022). The effects of ai-based credibility indicators on the detection and spread of misinformation under social influence. *Proceedings of the ACM on Human-Computer Interaction*, *6*(CSCW2), 1–27.

[6] Ecker, U. K., O'Reilly, Z., Reid, J. S., & Chang, E. P. (2020). The effectiveness of short-format refutational fact-checks. *British journal of psychology*, *111*(1), 36–54.

[7] Burel, G., Tavakoli, M., & Alani, H. (2024). Exploring the impact of automated correction of misinformation in social media. *AI Magazine*, *45*(2), 227–245.

[8] Ecker, U. K., Butler, L. H., & Hamby, A. (2020). You don't have to tell a story! a registered report testing the effectiveness of narrative versus non-narrative misinformation corrections. *Cognitive Research: Principles and Implications*, *5*, 1–26.

[9] Chandra, R. V., & Varanasi, B. S. (2015). *Python requests essentials*. Packt Publishing Ltd.

[10] Nakatani, S. (2010). *Language detection library for java*. https://github.com/shuyo/language-detection

[11] Foundation, P. S. (2025). *Unicodedata — unicode database (python standard library)* [Module in Python Standard Library, version 3.x]. Python Software Foundation. https://docs.python.org/3/library/unicodedata.html

[12] Taehoon Kim, K. W. (2025). *Emoji: A python library for converting emoji names and unicode characters* [Version 2.15.0 (Python 3.8+). https://pypi.org/project/emoji/]. https://pypi.org/project/emoji/

[13] Van Rossum, G. (2020). *The python library reference, release 3.8.2*. Python Software Foundation.

[14] Hutto, C., & Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the international AAAI conference on web and social media*, *8*(1), 216–225.