"Obama never said that": Evaluating fact-checks for topical consistency and quality

Katya Simpson

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2022

Committee:
Shane Steinert-Threlkeld
Amy X. Zhang

Program Authorized to Offer Degree:
Department of Linguistics

University of Washington

Abstract

"Obama never said that": Evaluating fact-checks for topical consistency and quality

Katya Simpson

Chair of the Supervisory Committee:
Shane Steinert-Threlkeld
Department of Linguistics

This thesis examines topical consistency between claims and fact-checks in the Birdwatch dataset published by Twitter. The dataset has tweets (the claims), notes (context-adding annotations written by Birdwatch users), and quality labels (ratings from the community of Birdwatch users). High quality notes can be thought of as potential "fact-checks" on the tweets. We find topics by clustering contextual word type embeddings (following a method introduced by Sia et al. [2020]) and evaluate two research questions (1) Do notes that have high topic overlap with their associated tweet get better ratings? and (2) Can this topic modeling method be used to measure the helpful extra context that Birdwatch notes add to a tweet? Kullback-Leibler divergence is used to proxy topic overlap between the documents. We find that there is a statistically significant relationship between topic overlap and helpfulness but cannot establish a relationship between helpfulness and added context.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

Many people contributed to this work and supported me during the CLMS program. I want to thank Shane Steinert-Threlkeld for his thoughtful guidance and encouragement during every step of this thesis. I'm also very thankful to Amy X. Zhang for providing valuable feedback and Emily M. Bender for teaching me how to think about the societal impacts of our work. The Birdwatch Team at Twitter showed me the potential of their community-based methods during my internship. I'm grateful that their data is available to researchers, and I'm excited about the impact Birdwatch will have. During this program, Devin Brown and Jessica Sweeney were my partners in crime and made doing an MS during a pandemic bizarrely fun. I'm thankful to Taylor St Baristas in Manhattan and Winemak'her in Brooklyn for quite a lot of caffeine and being friendly places to work. Finally, I'm so grateful to my fiancé Nickolai Riabov for unconditionally supporting me in three cities, through many late nights and for making sure we always have Kit-Kats.

DEDICATION

To my parents, Todd and Anar Simpson, for teaching me about computer science, life, and their intersection.

Chapter 1
INTRODUCTION

Engaging with misinformation and fact-checks is an increasingly standard part of our experience on social media. At least two social media companies launched community based, crowd-sourced efforts to reduce misinformation in recent years: Facebook's community review in 2019 and Twitter's Birdwatch program in 2021 [Yasseri and Menczer, 2021]. These efforts run in parallel to automatic fact-checking methods, but have not received much attention from the natural language processing community yet. Twitter has open-sourced the Birdwatch dataset, a resource that has claims (tweets), context-adding annotations (Birdwatch notes), and quality labels (ratings). High quality Birdwatch notes can be thought of as fact-checks on their associated tweets. We use the Birdwatch dataset to investigate a reasonably intuitive assumption about a crowd-sourced misinformation platform: **notes that are rated by the community as helpful should share topics with the content that they're fact-checking**. If that intuition is confirmed, this could be a step towards a larger goal of identifying spam or malicious behavior on crowd-sourced misinformation fighting platforms. We use a topic modeling method introduced by Sia et al. [2020] that defines topics as clusters of word type embeddings. The relative simplicity and low computational complexity of the method lends itself to social media settings.

More formally we ask two research questions to guide the analysis:

RQ1. Do notes that have high topic overlap with their associated tweet get better ratings?

RQ2. Can this topic modeling method be used to measure the helpful extra context that Birdwatch notes add to a tweet?

This thesis aims to lay the groundwork for future analyses of the Birdwatch dataset (and other crowd-sourced fact-checking data) with topic models.

Chapter 2

LITERATURE REVIEW

We apply methods from natural language processing (NLP) to the misinformation fact-checking space, and look to three areas to guide and ground our work. Within NLP literature, Latent Dirichlet Allocation (LDA) [Blei et al., 2003] is the standard method to find topics in an unsupervised manner. We look at LDA and its alternatives, and at topic modeling as it relates to fact-checking. We also learn from studies on Twitter's Birdwatch program within the fact-checking and misinformation literature.

2.1 Topic Model Choice

Chen et al. [2016] and Pang et al. [2016] note the difficulties of using LDA on short documents because of the heavy noise and lack of sufficient word co-occurrences. They propose methods to modify LDA, primarily as part of classification tasks.

Sia et al. [2020] introduce a computationally-efficient alternative to classical topic models like LDA. The key theoretical addition in their work is to think of topics as "clusters of word types" where word types can be represented by embeddings. They tested different clustering techniques, using contextualized and non-contextualized embeddings, and incorporating corpus statistics. They found that using term-frequency weighted K-Means clustering (with re-ranking of top word types with term-frequencies) yields topics that are comparably coherent to those generated with LDA.

The computational efficiency and relative simplicity of the method proposed by Sia et al. [2020] makes it a good candidate for the fast-moving fact-checking space.

2.2 Topic Modeling and Fact-Checking

Elements of topic modeling have been applied to automatic fact verification. Si et al. [2021] propose a model that checks for topical consistency (using LDA) between a claim and pieces of evidence, and classifies the evidence as supporting or refuting the claim. Hardalov et al. [2021], in a survey paper on stance detection (often used in fact-checking tasks), note that the benefits of using unsupervised topic models are limited because they tend to be noisy. Hanselowski et al. [2018] evaluate methods in a Fake News Challenge and found that methods that used features from topic models did not capture the semantic relationships necessary to do well on the task. These challenges emphasize the need to evaluate new topic modeling methods in the fact-checking area.

As far as the author is aware, this thesis is the first attempt to use the method introduced by Sia et al. [2020] in a fact-checking context. This thesis is also limited in comparison to the goals usually seen in fact-checking literature: we aim to learn about topic overlap between content and crowd-sourced fact-checks, not to classify the content.

2.3 Background on Twitter's Birdwatch

Twitter introduced Birdwatch in 2021 [Coleman, 2021] in an effort to identify and annotate misinformation on their platform. Birdwatch is community-based and relies on users to perform three mechanisms: labeling tweets as `misleading` or `not misleading`, writing annotations ("notes") on the tweets, and rating each other's notes for quality. The data generated from these mechanisms is freely available in the US on the Birdwatch User Guide.

This thesis refers to the Birdwatch program and the Birdwatch dataset. The Birdwatch program is what a Birdwatch user would interact with when they log on to Twitter and is the primary design focus of the Twitter Birdwatch team. The dataset is the output of the program: the data generated by users writing and rating notes.

The Birdwatch dataset has not been widely studied in the NLP community, likely because of how new and relatively small it is (though the widespread analysis of Twitter's tweet data

[Antonakaki et al., 2021] may indicate that the Birdwatch dataset will inevitably also be analyzed at scale). It has been evaluated from a security standpoint, examining sociotechnical vulnerabilities of the program [Benjamin, 2021]. The benefits and pitfalls of crowd-sourced fact-checking in the program have been studied by Yasseri and Menczer [2021]. Allen et al. [2021] used Birdwatch data in an experiment that found that partisanship of Birdwatch users is highly correlated with the type of notes they write. The crowd-sourced nature of Birdwatch, and the many design considerations that are necessary for good social media data use, are briefly examined in the Ethical Considerations section (section 6).

Chapter 3

METHODOLOGY

We replicate the method from Sia et al. [2020], with some modifications, to create clusters of word types from the Birdwatch dataset and treat those clusters as learned topics. These topics are evaluated for coherence in two ways: with Normalized Pointwise Mutual Information as a way to proxy human coherence and with Silhouette Indices as a distance metric. The learned topics will be mapped back to the notes and tweets to create topic probability distributions for each document. Finally, the topic overlap of tweet/note pairs will be measured with the inverse of Kullback–Leibler divergence. [1]

3.1 Birdwatch Terminology

- **Note**: an annotation that is in response to a tweet. Notes usually include linked sources. Notes that are rated as helpful by the community can be thought of as "fact-checks" on their associated tweet.

- **Tweet**: we think of a tweet as a "sister" document (or "claim") that a note references. A tweet can have multiple notes.

- **Rating**: users can mark a note as "Helpful", "Somewhat Helpful", or "Not Helpful" and label it with several other tags. The ratings help the Birdwatch program manage fact-checking quality. A note can have multiple ratings.

The term "document" is used to refer to both notes and tweets.

[1] The implementations and analyses for this thesis are on Github

3.2 Data Collection and Preprocessing

The data was downloaded from the Birdwatch github site as two TSV files (one for notes and one for ratings) that includes data up to Jan 30th, 2022. This represents nearly a full year of data since the Birdwatch program was launched Coleman [2021]. The notes data contains tweet IDs that map to the fact-checked tweets. The DocNow hydrator [Docnow, 2021] was used to obtain the tweet metadata (including the text) from the list of tweet IDs in accordance with Twitter's terms of service. Maintaining the mapping of a note and its associated tweet is not necessary while generating and clustering the embeddings but will be necessary when testing for the topic overlap between documents in the Experiment section 4.

Birdwatch was piloted in the US and most of the notes are written in English, although they could fact-check a non-English tweet. We use a English-language BERT model [Devlin et al., 2018] to generate word embeddings and omit a small proportion of the tweets that do not have English metadata labels. This approach can be expanded to non-English text using multilingual models, though we will not do this as part of this thesis.

There are 19333 notes and 14506 tweets in the dataset where 85% of the notes label a tweet as "misleading". 80% of the tweets get a single note. 72.5% of the notes have some sort of quality rating and only those notes will be evaluated for topic overlap although all the documents will be used to generate topics.

The documents are lower-cased with usernames, hashtags, urls, and digits omitted. The preprocessing step returns two outputs for each document: (1) a cleaned string that will be used by BERT (2) a list of cleaned tokens with stopwords omitted that will be used to calculate corpus statistics. The first output is a departure from the method outlined by Sia et al. [2020], who omitted stopwords prior to generating the BERT embeddings. Qiao et al. [2019] find that omitting stopwords does not harm the performance of a BERT ranking task although they receive as much attention as non-stopwords. We use that finding to assume that BERT is ambivalent to stop-word removal and leave them in for this task because

of the short document length, hoping to generate higher quality embeddings. We remove stopwords prior to calculating term-frequencies in 3.4 because we want to cluster over a meaningful vocabulary to create topics.

3.3 Train/Test Split

We use a 60/40 train/test split. The train data is used to calculate corpus statistics and generate and cluster the word type embeddings. The test data is used to evaluate the top words from the output clusters. The test/train split is necessary because the output topics are generated with term-frequency statistics and the primary evaluation method (Normalized PMI) also depends on corpus statistics.

3.4 Corpus Statistics

Term-frequencies are calculated from the documents in the train dataset. Sia et al. [2020] found that term frequencies (tf) are the most successful corpus statistics for this purpose and we follow their standard definition:

$$\mathbf{tf} = \frac{n_t}{\sum_{t'} n_{t'}}$$

where n_t is the count of a word type t. The final vocabulary contains 23,090 word types.

3.5 Generating BERT Embeddings

Token embeddings are created using the BERT base model Devlin et al. [2018] (*bert-base-uncased* pipeline from Huggingface). The full document is used as the context window and the last hidden layer is taken as the token's embedding. Following the method from Sia et al. [2020], subword representations are averaged. This method can generate different embeddings for the identical tokens because of their different context windows. These word token embeddings are averaged to create word type embeddings.

3.6 Clustering

The final step to find topics is to cluster the word type embeddings. Sia et al. [2020] use PCA to reduce the dimensionality of the embeddings prior to clustering, noting that the method allows for significant dimension reductions (up to 80%). Their method is followed: we use the tf weights calculated in 3.4 with a K-Means algorithm implementation [Pedregosa et al., 2011], and re-rank the word types in each cluster according to the tf weights to find the most representative top word types.

Two hyperparameters are considered: the number of clusters for the K-Means algorithm (the number of topics) and the number of dimensions for PCA reduction.

We searched the following hyperparameters:

- 20, 50, and 100 clusters
- 100, 300, 500, and 768 (no PCA) dimensions
- Three random seeds to initialize the K-Means algorithm

The output can now be evaluated using the top ten words of each cluster to find the best hyperparameters. As an example of the output of the search, we present the top ten words of the the first cluster generated (with twenty total clusters, one hundred dimensions, and random seed = 372). They are: “us”, “country”, “central”, “north”, “national”, “state”, “american”, “usa”, “city”, and “nation”.

3.7 Evaluation of Topic Clusters

In order to evaluate the coherence of the generated topic clusters and find the best set of hyperparameters, we follow Sia et al. [2020] and use Normalized Pointwise Mutual Information (NPMI) [Bouma, 2009]. NPMI is defined as

$$\mathbf{NPMI}(x, y) = \frac{\ln \frac{p(x,y)}{p(x)p(y)}}{-\ln(p(x, y))}$$

where $p(x)$ and $p(y)$ correspond to the probability of seeing words x and y, and $p(x, y)$ corresponds to the probability of words co-occurring. The probabilities are calculated from the test data. We calculate NPMI for the top 10 words of each cluster over every hyper-parameter permutation. The window for the PMI calculation indicates if it will capture a syntactic or semantic relationship [Jurafsky and Martin, 2019]. We use the first quartile of note length (14 tokens) as the window; intending to capture semantic relationships. The NPMI scores are averaged over three random seeds. NPMI has a range from [-1,1] where 1 would indicate that two words only occur together and -1 indicates that they would be completely independent of each other.

NPMI has been found to correlate to human judgements [Lau et al., 2014]. However, it's also possible to evaluate clusters with alternative metrics that aren't based on word co-occurrence. We look into a distance-based internal clustering validation method, hoping to reinforce any findings from the NPMI scores. We include silhouette indices [Liu et al., 2010] to capture how far a top word's embedding is to its cluster centroid and how far it is to other cluster's centroids. Similar to NPMI, it has a range from [-1,1] where -1 indicates that the top word may be in between or belong to several clusters, 0 indicates that there are overlapping clusters, and 1 indicates that the top word is perfectly contained in one distinct cluster. The mean score is then calculated over all the top words in a cluster [Pedregosa et al., 2011].

The evaluation metrics are presented in Table 3.1. The NPMI scores are all negative which makes sense when we consider how short the documents are and how few (approximately 15,000) are available in the test set. High joint probabilities $p(x, y)$ are relatively scarce and are often 0. That is to say, the words exist separately but don't occur together, resulting in an NPMI score of -1. This isn't entirely unexpected and aligns with known problems about LDA which also depends on word co-occurrences and does poorly when used on short social media text [Chen et al., 2016, Pang et al., 2016].

As an example, we can look at the NPMI scores for a fake cluster of words that should intuitively be highly correlated with each other. Consider ["biden", "putin", "russia", "usa"].

Table 3.1: Evaluation Scores (averaged over three random seeds)

NClusters	NDims	NPMI_Score	Silhouette_Index
20	100	-0.65	0.04
20	300	-0.67	0.06
20	500	-0.66	0.05
20	768	-0.62	0.06
50	100	-0.78	0.04
50	300	-0.73	0.05
50	500	-0.73	0.05
50	768	-0.72	0.05
100	100	-0.81	0.04
100	300	-0.76	0.04
100	500	-0.75	0.04
100	768	-0.73	0.04

We see that the pairwise NPMI is relatively low (Table 3.2), with the average being pulled to -1 because "putin" and "usa" don't co-occur in the test set

The silhouette scores in Table 3.1 are close to zero because of high dimensionality of the data. Kriegel et al. [2009] note that one of the problems with evaluating clusters in high dimensions is that the concept of distance becomes less meaningful. Interpreting distance metrics in high dimensions is a known problem [Tomašev and Radovanović, 2016] and limit the usefulness of silhouette indices and other distance based metrics.

Table 3.2: NPMI with missing co-occurrences

w1	w2	NPMI
biden	putin	0.16
biden	russia	0.05
biden	usa	0.03
putin	russia	0.29
putin	usa	-1.00
russia	usa	0.12
usa	biden	0.01

Assuming that NPMI will have a baseline of close to -1 for this task, we take the permutation of hyperparameters that results in the highest NPMI (which, perhaps coincidentally, has the highest silhouette score) and use 20 clusters and 768 dimensions.

Topic clusters (and all of their constituent word type embeddings) are re-generated with these parameters for analysis of the research questions. PCA is not used but should be reconsidered if computational complexity becomes more of a concern (for example, if Birdwatch is more widely used and the dataset becomes much larger).

3.8 Finding a Document's Topics and Topic Overlap

Topic overlap is defined as the similarity between a tweet's topic probability distribution and a note's topic probability distribution. Sia et al. [2020] propose an extension to their method to find the top topics per document which is slightly modified for this analysis. Using the word type embeddings, the pairwise cosine similarity between each token in a document and cluster centroid is calculated. The token similarities are summed over the clusters to create a document similarity vector. This vector is then normalized with a softmax function yielding a probability distribution for the document over the possible topics.

Kullback–Leibler (KL) divergence is used to measure topic overlap using the document probability distributions. KL divergence captures the average number of extra bits required to encode a note's topic distribution N using code optimized for a tweet's topic distribution T [Murphy, 2012]. More generally, it represents the average amount of information needed to discriminate between N and T. It is not a symmetric measure. $KL(N||T)$ represents the case where the probability distribution of the note is compared to the tweets (or, how many average bits of information it would take to encode the note with code optimized for the tweet). This direction intuitively captures the research question where the note is a response to a tweet, and is expected to have something in common with it. For a complete analysis, both directions will be computed.

Chapter 4

EXPERIMENTS AND ANALYSIS

4.1 Research Question 1

Do notes that have high topic overlap with their associated tweet get better ratings?

For a "yes" answer to RQ1, a negative relationship is expected: if the KL divergence is high, then the topic distributions are dissimilar, and helpfulness would be low.

To address this research question we also need information about the ratings. Helpfulness ratings are stored in two categories. The deprecated `helpful` category has two choices "Helpful" and "Not Helpful" and was used for notes written before June 2021. The currently used `helpfulnessLevel` category adds a "Somewhat Helpful" choice. We combined the ratings from before and after June 2021. These options are mapped to numeric counterparts (Not Helpful : -1, Somewhat Helpful : 0, Helpful : 1) and the outcome variable for this analysis is defined as the average of the numeric helpfulness ratings for a note. It should be noted that note/tweet pairs created before the launch of the `helpfulnessLevel` category may have a slightly different distribution in comparison to pairs created after June 2021, but the effect of this is left to further research.

Finally, the helpfulness level of the notes is linearly regressed against KL divergence of note/tweet pairs. Robust standard errors are used because the outcome variable is not normal. The results are in Table 4.1 and as seen in line 1, the relationship between helpfulness and $KL(N||T)$ is statistically significant and negative.

Document length has a strong effect on this analysis. If a document has a low amount of words, the document similarity vector (the summed cosine similarity between tokens and cluster centroids) will be flattened because of the low information content. We can gain

Table 4.1: Regression Results with Document Length Limits (*** denotes p < 0.001)

Regressor	Note Min	Tweet Min	Coef(SE)
$KL(N\|\|T)$	0	0	$-0.291(0.051)^{***}$
$KL(T\|\|N)$	0	0	$-0.050(0.053)$
$KL(N\|\|T)$	8	8	$-0.051(0.098)$
$KL(T\|\|N)$	8	8	$0.056(0.104)$
$KL(N\|\|T)$	0	8	$-0.377(0.055)^{***}$
$KL(T\|\|N)$	0	8	$-0.293(0.083)^{***}$
$KL(N\|\|T)$	8	0	$-0.056(0.082)$
$KL(T\|\|N)$	8	0	$0.006(0.058)$

intuition for this by looking at the uniformity of the distribution by document length. KL divergence from a uniform distribution is calculated for each document's topic distribution.There is a clear trend in Fig 4.1, for both notes and tweets, where shorter documents are more similar to the uniform distribution.

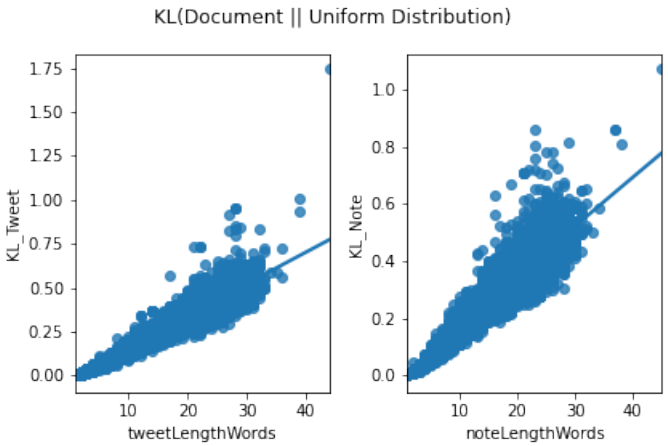

Figure 4.1: KL Divergence of a document's topic distribution and a uniform topic distribution as a function of length

A uniform topic distribution suggests that this method is not well-suited for very short documents and fails to find clear top topics for them. If we omit "short" tweets (below

the first quartile without stopwords, or 8 tokens), the relationship between KL divergence and helpfulness becomes statistically significantly negative for the $T||N$ direction as well. Interestingly, this is not the case when short notes are omitted. This could be because helpful notes may add extra context, modifying the topic distribution. This line of reasoning is continued in the analysis of RQ2 (section 4.2).

We also tried a model specification geared towards separately identifying the impact of KL divergence and document length on the helpfulness level.

$$\begin{aligned} helpfulness \sim KL(N||T) + KL(N||T) \times noteLength \\ + KL(N||T) \times tweetLength \\ + tweetLength + noteLength \end{aligned}$$

The results are in Table 4.2. This model showed:

- KL divergence is a statistically-significant predictor of the helpfulness in the presence of the other features
- there is no heterogeneous effect. The KL divergence interactions with either note length or tweet length are not statistically significant.

Table 4.2: Regression results testing for the effect of length on helpfulness

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
Intercept	0.2830	0.026	10.826	0.000	0.232	0.334
KL_nt	-0.4743	0.209	-2.270	0.023	-0.884	-0.065
noteLengthWords	0.0081	0.002	5.381	0.000	0.005	0.011
KL_nt:noteLengthWords	0.0086	0.008	1.086	0.278	-0.007	0.024
tweetLengthWords	-0.0031	0.001	-2.178	0.029	-0.006	-0.000
KL_nt:tweetLengthWords	0.0093	0.008	1.164	0.244	-0.006	0.025

This tells us that document length itself does not strengthen the relationship between the KL divergence and the helpfulness.

4.2 Research Question 2

Can this topic modeling method be used to measure the helpful extra context that Birdwatch notes add to a tweet?

Birdwatch users can label notes in a variety of ways after marking them as helpful (see figure A.1 as an example). Labels are not required to submit a rating so we assume that the presence of at least one label per note is sufficient to associate the label's trait with the note. For example, a note may have only one "Easy to Understand" label even though it has multiple ratings.

Intuitively, we can think of a note's extra context as a shift in the topic probability distribution when compared to its tweet. Figure 4.2 shows an example. This research question will also be addressed using document topic distributions and focus on $KL(N||T)$ because it more clearly captures the note-tweet relationship. We try to approach the question from three different angles.

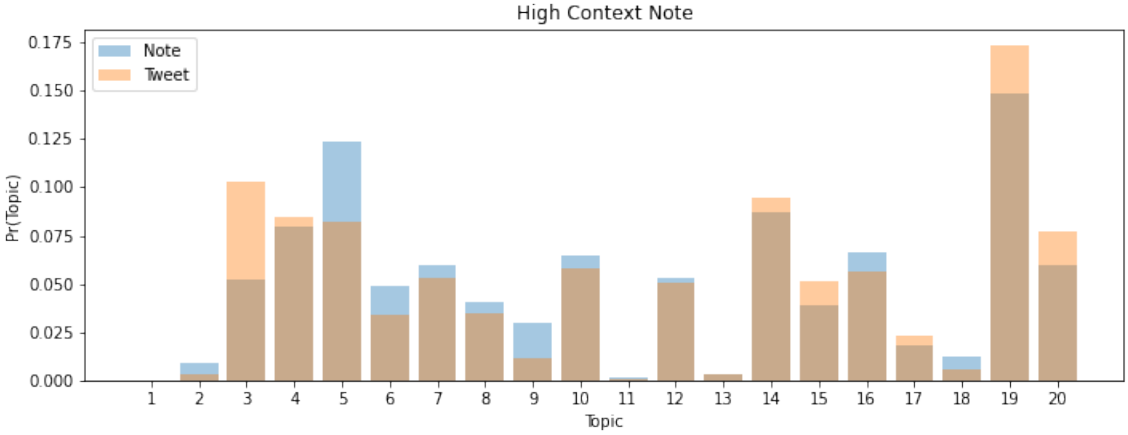

Figure 4.2: Document topic distributions for a note that adds context

First, we find a subset of the data with `Provide Important Context` labels and ask "Will helpfulness increase as $KL(N||T)$ increases?"

$$helpfulness \sim KL(N||T)$$

This is the same linear regression that was used in RQ1 but on subset of the original data. The result is statistically significant and negative: if the topics are similar then helpfulness

still increases. However, the coefficient -0.21 is slightly lower than when calculated over the entire dataset (row 1 of Table 4.1).

Second, we return to considering the entire dataset and ask "Does the presence of the add context label increase the effect of $KL(N||T)$ on helpfulness?"

$$helpfulness \sim KL(N||T) + context\_add + KL(N||T) \times context\_add$$

From our results in Table 4.3, we find that the interaction of the `Provides Important Context` label with $KL(N||T)$ does nothing to increase the effect of $KL(N||T)$ on helpfulness in a statistically meaningful way.

Table 4.3: Regression results testing for the effect of context labels and KL on helpfulness

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
Intercept	0.1709	0.020	8.694	0.000	0.132	0.209
KL_nt	-0.3912	0.098	-3.987	0.000	-0.583	-0.199
add_context	0.2826	0.022	12.848	0.000	0.240	0.326
add_context:KL_nt	0.1765	0.112	1.577	0.115	-0.043	0.396

Third, we continue to consider the entire dataset and ask "What is the effect of $KL(N||T)$ on the probability of having add context label?".

$$context\_add \sim KL(N||T)$$

The label's presence is a binary variable so a logistic regression is used and we find that there is not a statistically significant effect (as seen in Table 4.4).

Birdwatch's rating mechanisms are not complete from an annotation perspective. The lack of certain label cannot be a indicator that the label truly does not apply. It's possible that there is a large proportion of notes that add context but have not been labeled as such. This casts some doubt on non-significant results from the second and third lines of inquiry

Table 4.4: Logistic regression results

	coef	**std err**	**z**	**P> |z|**	**[0.025**	**0.975]**
Intercept	0.8411	0.035	24.201	0.000	0.773	0.909
KL_nt	-0.2303	0.186	-1.235	0.217	-0.596	0.135

which assume that *context_add* is a binary variable that maps to a ground-truth.

Finally, we return to the question posed in RQ1 and evaluate if the average length of the notes differs between the ones with context-adding labels and the ones without, using a two-sided t-test. It yields a t-statistic of -10.985 corresponding to a p-value < 0.0001. That is, the context adding subset has statistically significantly longer notes than the notes that don't have that label. This adds some credence to interpretation from RQ1: omitting short notes leads to an insignificant relationship because helpful, longer notes may add context and shift the topic distribution increase KL divergence. Note that the caveat about Birdwatch's incomplete annotations applies to this analysis as well.

Chapter 5

DISCUSSION

We find a statistically significant negative relationship between the independent variable $KL(N||T)$ and the dependent variable (helpfulness). We can answer a narrow version of RQ1 with some confidence: notes that have higher topic overlap with their tweets seem to get better, or more helpful, ratings. The directional nature of KL divergence plays a role in this interpretation. While $KL(N||T)$ was found to be a statistically significant variable, $KL(T||N)$ was not. One possible explanation is that notes can add context to the tweet's original content and alter the document's topic distribution. Then, the additional average information to encode a tweet (based on code optimized for the note) would increase to accommodate the topic distribution shift. Examining a symmetric KL measure (by averaging the two directions) may be a good next step to understand the results in a more broad way: Table A.1 shows that a symmetric KL measure is statistically significant with no length limits, implying that the $KL(N||T)$ is relatively strong.

Omitting short tweets results in a stronger relationship between $KL(N||T)$ and helpfulness, and a statistically significant relationship with $KL(T||N)$. However, omitting short notes results in no significant relationships at all. A possible explanation for this is that helpful, longer notes may add more context than short notes (intuitively, it would take more tokens to both address the tweet's topic and add in new context to refute or support it) which would shift the topic distribution. Thus, KL divergence would increase while helpfulness also increases. This explanation has some weight because the group of notes that are labeled as adding context are on average longer than notes that are not. However, there may be more reasons that can't be easily captured by a topic model: it's possible that short notes share topics but are unhelpful (imagine a tweet that says "The earth is flat!" and a note that uses

a sarcasm indicator "yes the the earth is flat! \s").

The answer to RQ2 is inconclusive. The method does seem to capture the extra context when considering a subset of data with `Provides Important Context` labels in comparison to the full dataset. However, when considering the presence of the label as a binary variable, we get statistically insignificant results. This may be because the presence of a variable does not map perfectly to a binary trait. There may be a large amount of helpful notes that add context but are not labeled as such.

Document length creates difficulties with this method and the analysis of the research questions:

1. Evaluating topic coherence with word co-occurrence measures, like NPMI, may be less effective because of the scarcity of joint probabilities for certain pairs of words

2. Finding the topic probabilities for short documents can result in uniform distributions without clear top topics

It's possible that (1) could be remediated by using a test set with more documents to fill out the word co-occurrence matrix. These issues are connected: if (1) was remediated, it may be possible to find the topics that clearly have the highest NPMI scores and use them to weed out "catch all" topics, forcing a less uniform distribution for short documents.

Overall, the method by Sia et al. [2020] seems to work appropriately for the limited scope of these research questions. Given that this method aims to minimize computational complexity it's possible that this could be used to find "troll" responses that are off-topic as they are created.

Chapter 6

ETHICAL CONSIDERATIONS

This thesis uses text data from Birdwatch without incorporating important conversations about the program itself. We intend any applications or future work that draw on this thesis to be considered alongside studies about the Birdwatch program and crowd-sourced fact-checking.

Following the guidelines from Williams et al. [2017] and Fiesler and Proferes [2018] we try to protect social media user's privacy and do not include specific note text, participant IDs, tweet IDs, or Twitter handles. We believe this practice is especially relevant for the Birdwatch dataset because of its small size (tracking down the alias of note's author would be relatively easy). The data is considered only in aggregate. The title of this thesis is a play on a typical note.

Chapter 7

CONCLUSION

This thesis examined topic overlap between Birdwatch notes and their associated tweets. We wanted to validate a reasonable assumption about fact-checking (that helpful fact-checks share topics with the content they are written about) and test for it with a topic model. We examined the research questions using a method adopted from Sia et al. [2020] as an alternative to LDA and found that there is a statistically significant relationship between topic overlap $KL(N||T)$ and helpfulness. We extended this question to see if notes add helpful additional context and found no conclusive relationship. While the methodology addressed the original research questions, future work and access to more data will be necessary for this method's use in applied settings.

Future work may include researching a multilingual approach to capture the relationship between non-English tweets and their notes, using more tweet training data to increase word pair co-occurrences (or using evaluation methods that don't require corpus statistics or distance metrics), and further work to understand how the Birdwatch annotation process may affect downstream tasks.

We hope that this work, and future improvements on it, can help improve crowd-sourced fact-checking programs. Using topic overlap to flag a low-quality fact-check at the time of writing could prompt a user to write a higher-quality alternative, and may be tied into NLP work that aims to find spam. Documents with topics that are typically highly annotated could be presented to human fact-checkers for review.

BIBLIOGRAPHY

J. N. L. Allen, C. Martel, and D. Rand. Birds of a feather don't fact-check each other: Partisanship and the evaluation of news in twitter's birdwatch crowdsourced fact-checking program. 2021.

D. Antonakaki, P. Fragopoulou, and S. Ioannidis. A survey of twitter research: Data model, graph structure, sentiment analysis and attacks. *Expert Systems with Applications*, 164: 114006, 2021.

G. Benjamin. Who watches the birdwatchers?: Sociotechnical vulnerabilities in twitter's content contextualisation. In *11th International Workshop on Socio-Technical Aspects in Security, ESORICS 2021*, 2021.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

G. Bouma. Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, 30:31–40, 2009.

Q. Chen, L. Yao, and J. Yang. Short text classification based on lda topic model. In *2016 International Conference on Audio, Language and Image Processing (ICALIP)*, pages 749–753. IEEE, 2016.

K. Coleman. Introducing birdwatch, a community-based approach to misinformation, Jan 2021. URL https://blog.twitter.com/en_us/topics/product/2021/introducing-birdwatch-a-community-based-approach-to-misinformation.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional

transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Docnow. Documenting the now, Jan 2021. URL https://github.com/DocNow/hydrator.

C. Fiesler and N. Proferes. "participant" perceptions of twitter research ethics. *Social Media+ Society*, 4(1):2056305118763366, 2018.

A. Hanselowski, A. PVS, B. Schiller, F. Caspelherr, D. Chaudhuri, C. M. Meyer, and I. Gurevych. A retrospective analysis of the fake news challenge stance-detection task. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1859–1874, Santa Fe, New Mexico, USA, Aug. 2018. Association for Computational Linguistics. URL https://aclanthology.org/C18-1158.

M. Hardalov, A. Arora, P. Nakov, and I. Augenstein. A survey on stance detection for mis- and disinformation identification. *ArXiv*, abs/2103.00242, 2021.

D. Jurafsky and J. H. Martin. *Chapter 6, Vector Semantics and Embeddings*. Prentice Hall, Pearson Education International, 2019.

H.-P. Kriegel, P. Kröger, and A. Zimek. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *Acm transactions on knowledge discovery from data (tkdd)*, 3(1):1–58, 2009.

J. H. Lau, D. Newman, and T. Baldwin. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539, 2014.

Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu. Understanding of internal clustering validation measures. In *2010 IEEE international conference on data mining*, pages 911–916. IEEE, 2010.

K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

X. Pang, B. Wan, H. Li, and W. Lin. Mr-lda: an efficient topic model for classification of short text in big social data. *International Journal of Grid and High Performance Computing (IJGHPC)*, 8(4):100–113, 2016.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Y. Qiao, C. Xiong, Z. Liu, and Z. Liu. Understanding the behaviors of bert in ranking. *ArXiv*, abs/1904.07531, 2019.

J. Si, D. Zhou, T. Li, X. Shi, and Y. He. Topic-aware evidence reasoning and stance-aware aggregation for fact verification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1612–1622, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.128. URL https://aclanthology.org/2021.acl-long.128.

S. Sia, A. Dalmia, and S. J. Mielke. Tired of topic models? clusters of pretrained word embeddings make for fast and good topics too! In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1728–1736, Online, Nov. 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.135. URL https://aclanthology.org/2020.emnlp-main.135.

N. Tomašev and M. Radovanović. Clustering evaluation in high-dimensional data. In *Unsupervised learning algorithms*, pages 71–107. Springer, 2016.

M. L. Williams, P. Burnap, and L. Sloan. Towards an ethical framework for publishing twitter data in social research: Taking into account users' views, online context and algorithmic estimation. *Sociology*, 51(6):1149–1168, 2017.

T. Yasseri and F. Menczer. Can crowdsourcing rescue the social marketplace of ideas? *arXiv preprint arXiv:2104.13754*, 2021.

Appendix A

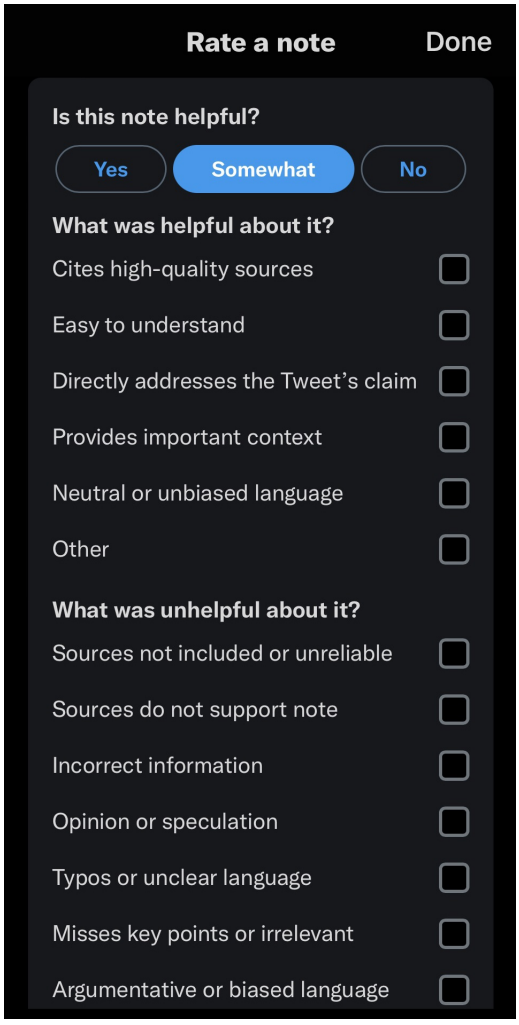

Figure A.1: Screenshot of the Birdwatch user interface with note labels

Table A.1: Symmetric KL Regression Results with Document Length Limits (*** denotes p < 0.001)

Regressor	Note Min	Tweet Min	Coef(SE)
KL_avg	0	0	-0.194(0.056)***
KL_avg	8	8	-0.002(0.103)
KL_avg	0	8	-0.386(0.068)***
KL_avg	8	0	-0.017(0.070)