How Topic Modeling and Sentiment Analysis Articulate (Mis)Representation in Written Media

Aaron Bronstone, Thien Bui, Riaz Kelly, AJ LeSure, Daniel Linder

Carleton College, Northfield MN

CSs400: Senior Integrative Exercise (COMPS) "Quotes in Context"

March 13, 2024

Abstract

We aim to find context for quotes used in sports articles in the form of transcripts, and use natural language processing techniques to determine ways in which quotes can be taken out of context and misconstrued by sports articles. This research focuses on exploring topic-related and sentimental discrepancies between articles and the sources they cite. We use a multi-corpus topic distribution comparison method and NRC-lex emotion analysis to find differences between sport articles and sport interview transcripts. We found several factors that influence a topic modeling approach to determining context: differences in media language styles, irrelevant topic types, and the number of topics related to the quote and its context. Through sentiment analysis we found that articles tend to use more negative language than the transcripts the quotes came from. Finally, we analyzed matched pairs of articles and transcripts via the quotes they share, and see that the average article focuses on less topics in general than transcripts.

1. Introduction

Quotations are a very powerful tool in written media with regards to communication and persuasion. When used correctly, quotes can be used in constructive manners with crediting others for ideas and thoughts, but they can also be used in ill-mannered ways to paint bad pictures of others and broadcast their words to audiences. Quotes are picked from primary sources to aid a new argument, whether that argument builds on original ideas or opposes them. However, do these new arguments always properly represent the original argument, idea or circumstances the quote was taken from, or do some misuse the original context?

Context is a very subjective concept and there are many independent factors that affect how context can and should be used. For quotations, a non-exhaustive list of circumstances to consider can include the personnels involved when said quotes were uttered, the events taking place prior to and after the quotation, and/or the surrounding conversation from where the quote originates. Even these circumstances are subjective within themselves; how much knowledge of preceding events is required to understand why this football coach said this controversial quote? What is the relationship between who said the quote and who receives it? Did the person saying the quote intend for those words to come out that way or be perceived the way they were?

To explore this idea, we seek ways in which we can turn the subjective nature of context into a formal, objective measure using various computational techniques. Under the right training conditions, we had hoped that our computations would enable us to quantitatively determine if news outlets are using quotes in articles with the proper context, in relation to their original meanings. While there have been a number of studies looking into "fake news" on a political level, we were unable to find any that delve specifically into the realm of sports journalism. This novelty, as well as personal interest and the lightheartedness that exists in sports compared to politics, had led us to choose this as our topic of focus. We decided to use articles as our primary media source due to their vast availability, and using only the articles which contain quotes from interview transcripts, we hope to identify discrepancies as it relates to our elusive definition of *context*. As for how we identify the source of the context to be respected or taken out of, we decided to use the transcripts where these quotations originated from.

In order to determine whether articles are taking quotes from transcripts out of their original context, thus distorting their meanings, we use a variety of Natural Language Processing (NLP) techniques[1]. The two primary subfields of NLP we focused on for this project were topic modeling, where we analyze *what* the articles and transcripts are talking about, and sentiment analysis, which focuses on *how* the articles and transcripts talk.

1a. What is topic modeling?

Topic modeling is a subset of natural language processing, where the goal is to identify hidden themes that documents (i.e. written text) can share to determine *what* the documents are talking about. A topic modeler can group documents together, and from these groupings we hope to obtain some insights regarding the contexts of the contents being discussed. For example, if a transcript is categorized as containing topics that include words like "Nick Saban", "football", "Crimson Tide", "Alabama", while the articles that quoted this interview are categorized by topics that does not include these words, it could indicate contextual misrepresentation given that we are considering the transcript as the contextual source of the quote. Due to the qualitative nature of *context*, we've used an LDA topic modeler (which we will discuss in a later section) to compare articles to our defined source of quote context (transcripts), and to identify matched pairs of articles and transcripts that *may* contain misrepresentations, but the actual analysis of *how* these quotes are misrepresented are carried out quantitatively via our sentiment analyzers and careful reading.

1b. What is sentiment & emotion analysis?

Sentiment analysis is a subset of natural language processing centered around measuring the "sentiment" in a given text. A sentiment analyzer can be used to group documents based on the *emotions* they exhibit, and from these groupings we can obtain some insights on *how* documents talk about their subjects. For example, if a transcript is considered to be overall *positive* while the article that is referencing it is overall *negative*, it could indicate a level of context relating to the intentions of the information being conveyed. We further expand on this hypothesis by using what is known as emotion analysis in this paper. Instead of outputting simply *positive, negative, and neutral*, emotion analysis outputs distributions for a predefined selection of emotions like *sad, happy, joyful,* and *resentment*. We will be using a library called NRC-Lex (which we will discuss in a later section) in order to capture the emotions associated with our transcripts and articles to see whether the emotions in topically different transcripts and articles exhibit different emotions as well.

[1] NLP is a field of computer science combining the rule-based model of human language with statistical and machine learning models to draw potentially unseen linguistic information from text [19]

2. Data Collection & Preparation

In this section, we will describe the kind of data we're interested in studying, along with some additional information about how they are collected. Data collection can be divided into three parts: articles, transcripts, and quotes.

As a group, we collectively decided to pursue a news aggregator. With the help of Kevin Draper (Carleton College '11) and Michael Cupo (Disney) we ran into Perigon [20]. Perigon is a video/article news aggregator that provides access to news articles from various news outlets. Using their API, we obtained articles from ESPN, Fox Sports, NYT, and the Associated Press, along with their associated metadata[2]. After contacting Perigon regarding our research, we were supplied with three API academic trial keys for us to use, allowing us to extract roughly 130,000 unique articles that both contain quotes and contain "college sports" in the subject metadata. The only downside to this dataset, to our knowledge, is that it only includes articles written from January 1st, 2021 until January 18 2024[3] (our final day of data collection).

ASAP Sports [2] is a website that hosts data on primarily college sport interview press conference transcripts. Clients of ASAP Sports include the NBA, MLB, and NCAA. We contacted ASAP Sports regarding data access for use in our research, but have not received a reply to our initial correspondence. A Terms of Service couldn't be found anywhere on their website, so we proceeded with scraping the transcripts they had available. We built a custom scraper using the Selenium and BeautifulSoup Python libraries, and were able to extract around 65,000 transcripts for football, basketball, tennis, swimming, volleyball, track and field, baseball and soccer from the year 1998 to 2024[4]. Each transcript included the speaker, in addition to the transcribed text of what is being said.

To see how topic and sentiment values of the raw quotes compare to articles and transcripts, we extracted the raw quotes from each article along with the speaker's name using Journalism AI's Quote Extractor [15]. Of the three data collection parts, collecting quotes proved to be the least challenging.

In order to sanitize our datasets prior to training, we used NLTK's Stopword English corpus [13] and Gensim built in stopword removal preprocessing library [7] to remove words like "the", "is", "a", and other non-English words. We also lemmatized our corpora using NLTK's wordnet module [12], which would group words like "athletes" and "athlete" into the same training token.

3. Methodologies

In this section, we will discuss the kind of techniques we will be using to study our data and provide a brief overview on how each of the techniques works, along with the kind of information we can learn from each of them.

3a. Quote Matching

While some of our analysis focused on comparing a secondary-source and primary source set of documents, we also wanted to explore specific instances of quotes being used in or out of context on a case-by-case basis. To do so, we algorithmically matched articles to transcripts that contained the same quotations, and performed our analysis on this subset of data and used the result as a way to verify our findings from the larger corpus strategies. This process ideally would give us an idea of some micro trends that show whether sports quotes are used effectively. We will spend the rest of this section describing the processes we've used to identify matches.

[2] Metadata includes the source of the article, publishing date, length of article, author, etc…

[3] Goperigon.com was established in 2021, which is likely the reason why we were able to easily get help from them regarding API usage for research.

[4] This dataset does not include transcripts taken from the NFL.

3bi. Levenshtein Distance

Levenshtein distance is a way of measuring the similarity of two strings in a quantitative manner. The Levenshtein distance between two strings is the number of substitutions (letter for a different letter), deletions (removal of a letter), and/or additions (insertion of a letter) it takes to transform one string into the other [17]. Levenshtein distance is commutative; it does not matter which string you start with.

3bii. Fuzzy Matching

We used a technique called "fuzzy matching" to find quotes within transcripts. The tool for fuzzy matching incorporated the python library thefuzz, also known as fuzzywuzzy [6] in order to find the closest match for each quote in each transcript. This library has methods for comparing characters, tokens, and combinations of substrings between two strings. Our main use of this library centered around the partial_ratio() function, which takes in two strings and outputs a score from 0 to 100 based on the Levenshtein distance between the first of them and every substring in the second one using the formula in figure 1. The smaller string in this case would be the quote and the larger string was an entire transcript. Our process consisted of comparing each quote with each transcript to find the best match for that quote in the transcript using partial_ratio(), because the quote would be a substring within the transcript. The partial_ratio() method finds the Levenshtein distance between the quote and each substring of the transcript. Our method compared each quote with each transcript, and for each quote, matched the quote and article with the transcript that gave the highest output for partial_ratio(quote, transcript) with a few constraints.

$$\frac{(|a| + |b|) - lev_{a,b}(i, j)}{|a| + |b|}$$

Figure 1. This is the expression for the 'confidence' score between two strings, where the Levenshtein distance is subtracted from |a| + |b|, or the length of the two strings combined. Longer strings will have a higher confidence score for the same distance

We set the minimum confidence score to be 85 for any quote, where if no transcript comparison for that quote gave a score of 85 or greater we would not consider it 'matched'. We also only used quotes with a minimum of 8 tokens to minimize false positive results, where in some cases a quote would be so generic that it would have a score of 85 with a transcript that it was not actually taken from. We will analyze limitations of this strategy in the limitations section.

3c. Topic Modeling with Latent Dirichlet Allocation

One of the most common tools used to build topic models is latent dirichlet allocation (LDA)[11]. It is a probabilistic topic modeling technique that takes in a list of documents as input, and outputs a statistical model that will tell us the probability of a document containing certain topics (in the form of a dirichlet distribution [5]. We have found this technique to be very unstable when applied to our dataset (see Appendix about LDA instability), but the ease of implementation via the Gensim package [8] made this method really attractive for our purpose. We will spend the rest of this section discussing how we used the outputs of these LDA models to compare topical differences between sport articles and the transcripts they are quoting from.

3d. MTCC: A Multi-Corpus Comparison Technique

Our topic modeling analysis used a unique multi-corpus comparison methodology proposed by Lu et. al [1]. The "Multi-corpus Topic-based Corpus Comparison" (MTCC) takes in several parameters to note:

- *n* - number of corpora (articles vs transcripts vs quotes = 3 corpora)

- *[k]* - several values of *k*, where *k* is the number of topics a model is to be trained on (we arbitrarily decided on [5 10 20 50 100])
- *m* - divergence score metric used to rank the topics
- *g* - the global policy to be used for divergence score ranking (*sum*, *wsum*, *max*)

Process

Below is a high-level overview of the main five steps of MTCC:

1. **Combine corpora** - combine *n* corpora into one single master corpus
2. **Train LDA Models -** for each value *k*, train an LDA model the master corpus
3. **Compute Semantic Coherence -** for each of the *k* models, compute a coherency score to determine which model to progress with. The coherency score is computed using FastText [18] (see Appendix on Coherency Scores)
4. **Compute Discriminatory Topics -** find the most discriminatory topics using *g* and *m* using Jensen-Shannon divergence in 1 vs all comparisons (see Appendix on Jensen Shannon Divergence)
5. **Visualization -** create a 2d representation of all document-topic vectors with a mapping of most discriminatory topic (see Appendix on Data Visualization)

Output

There are two notable outputs from the MTCC code. First, the results of the topic comparison (step 4) are the top 10 most discriminatory topics calculated from all the topics in the most coherent LDA model. These are the results of performing one vs all Jenson-Shannon computations between all of the corpora for each topic, and ranking the topics by the maximum JSD value found (the corpus that differs the most from the others for that particular topic, see Appendix on Jensen Shannon Divergence).

The second output is an interactive 2d scatter plot generated with the Bokeh Python module, where each dot is a document's topic vector shrunken down from *k*-dimensions to 2 dimensions via t-SNE computations (see Appendix for t-SNE computation). Additionally, the top discriminatory topics are also mapped on the plot based on where those topic vectors would appear. The closer two dots are together, the more similar their original *k*-dimensional topic vectors are.

Each document in the visualization is either colored or gray. If a document is gray, this means that its most probable topic (out of all *k* topics in the vector) is *not* one of the top 10 discriminatory topics. If a dot is colored, then that document's most probable topic is one of the discriminatory topics, and it is colored based on which corpus it originates from. Visualizations are shown in section 4a.The ways in which we will be utilizing MTCC can be found in the following Black Box Testing section.

Black Box Testing

Due to time limitations, we opted for a "black box" testing methodology to explore the outputs of MTCC and infer the results as they relate to topical distributions of our various corpora. We formulated different combinations of corpus subsets (from the original three corpora of articles, transcripts and quotes), and analyzed the output of MTCC in order to quantify contextual differences between the various sub corpora. Below are some combinations of corpora we tried along with the motivation behind each one:

Combination	**Motivation**

Articles vs **transcripts** vs **quotes** (only unigrams, limited stopwording)	First run of MTCC
Articles vs **transcripts** vs **quotes** (unigrams and bigrams, limited stopwording)	How do bigrams affect the most discriminatory topics?
Articles vs **transcripts** vs **quotes** (unigrams and bigrams, conversational language stopwording)	Do conversational prose/language differences between articles and transcripts influence topic differences?
Articles vs **transcripts** (football topic) vs **quotes** (extracted from football articles)	Does a variety of sports in both corpora influence topic differences?
Articles vs **quotes** (unigrams & bigrams)	Does the inclusion of another corpus affect the distribution of the other two corpora? How do the discriminatory topics change?
Articles separated by sport **Transcripts** separated by sport	Does the language used in articles vary more than transcripts depending on what sport the document is about?
Matched **articles** vs **transcripts** (unigrams/bigrams, using models trained on master corpora)	Are matched articles and transcripts different in terms of topic?

(See Appendix for additional MTCC Results)

3e. Sentiment & Emotion Analysis

In this section, we will discuss how we have used sentiment analysis to quantify *how* articles and transcripts might be different from one another. We will start by providing a detailed description of how we have extracted the sentiment of our various datasets.

For each of the articles, transcripts, and quotes corpora, we implemented this process:

1. Taking the CSV file for the corpus as input, run NRC-Lex to extract the sentiment-emotion data from each document in the corpus. See appendix for more information on NRC-Lex.
2. Using the sentiment-emotion data from each document in the corpus, output an aggregate based on the corpus it originates from.
3. Make a bar graph of the aggregated data using matplotlib.

Table 1: I/O flow of sentiment/emotion analysis. "<corpus>" is a placeholder representing any of the three corpora, as the process was the same for all.

Process/algorithm	**Input**	**Output**

Extracting sentiment-analysis data	`<corpus>.csv`	`<corpus>_AFFECT_FREQUENCIES.txt`
		`<corpus>_RAW_SCORES.txt`
		`<corpus>_TOP_EMOTIONS.txt`

Aggregating data of all documents in corpus	`<corpus>_RAW_SCORES.txt`	`<corpus>.AGGREGATE.txt`
Graphing aggregated data	`<corpus>.AGGREGATE.txt`	Pie charts for each corpus, one bar chart for all corpora

We chose to exclude the quotes corpus from sentiment analysis because they are a direct subset of the articles corpus and didn't yield any interesting results. Unlike topic modeling, we included a control group to give us an idea of typical sentiment values for news articles. We ran the same sentiment-emotion analysis on a group of approximately 4,000 non–sport news articles written for CNN in the year 2023.

4. Results & Analysis

In this section, we will discuss the results of our various computational experiments, alongside with brief explanations on why we think the results are the way they are.

4a. MTCC

Our first run of MTCC was comparing three corpora: all articles, all transcripts, and all quotes, where each document in the quote corpus contained all of the quotes from each article[5]. We did no further stopwording for this trial, instead relying only on our preprocessing step (see Data Collection and Preparation). The outputs are shown in Figure 1 and Table 2.

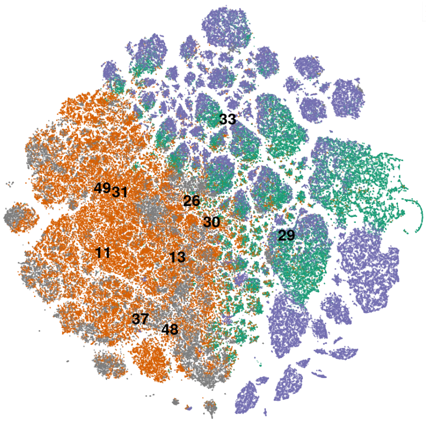


Figure 2. First visualization of MTCC using all articles (orange), all quotes (green), and all transcripts (purple)

Table 2. top 10 most discriminatory topics, in descending order by discriminatory power. Source corpus is the corpus responsible for the topic's discriminatory power when using the "maximum" global ranking policy

topic	words	source corpus
29	think , know , i'm , yeah , match , play , dont , good , playing	quotes
33	guy , game , he , got , think , going , team , good , play , lot	quotes
11	game , team , win , season , week , said , play , loss , Saturday , state	articles
31	alabama , georgia , sec , florida , lsu , class , texas , recruiting , miss , buckeye	articles
13	football , college , nick , sport , news , paul , nfl , saban , medium , according	articles
48	conference , committee , school , west , virginia , member , division , commissioner , decision , athletic	articles
49	season , year , player , freshman , injury , high , best , starting , williams , spring	articles
26	didn't , like , time , way , week , don't , day , play , good , game	quotes
37	big , 12 , texas , oklahoma , conference , state , washington , kansas , oregon , pac12	articles
30	year , player , want , going , think , thing , that's , don't , people , right	transcripts

[5] We constructed the quote corpus in this manner to keep the number of documents similar to the other twocorporas, and to ensure we don't have any documents that are too short (i.e. the quote being "he scored a goal").

We can see a couple of interesting findings in these initial results. In Figure 2, despite the quotes being a direct subset of the articles and the articles *still including* them, the quotes are clustering heavily with the transcript documents. This indicates a general trend that when quotes are used in articles, the language style of dialogue (between the interviewee and reporters) isn’t tampered.

We can also see in Table 2 that 7 of the 10 top discriminatory topics are caused by the divergence of the article corpus. One factor that could account for this is that press conferences and interviews aren’t very specific when it comes to discussing the sport itself - they would mostly be discussing player performance, thoughts on outcomes or personal events, and such, whereas journalists might take more time to explain sport-specific terminology for more concise communication. Additionally, this result could indicate that due to articles being heavy in sport terminology, the variety of sports present in our first iteration could have caused the most discriminatory topics to only appear in the articles.

Removing Conversational Words

Upon closer inspection of the top discriminatory topics, we noticed that words like “think”, “guy”, “didn’t”, “like”, “good” and such made up a lot of the top topics, which were closely associated with the quote and transcript corpora. An explanation for this is likely that written language of articles doesn’t typically include these kinds of conversational words, given that authors can take more time and use more resources to write and articulate using concise words, which explains why those general words from the quotes/transcript corpora would discriminate the most compared to articles. To remove this noise, we added some additional stopwords [14], which resulted in discriminatory topics that are much more aligned with what we perceive as topics.

Table 3. top 10 most discriminatory topics **after removing conversational language**, in descending order by discriminatory power. Source corpus is the corpus responsible for the topic’s discriminatory power when using the “maximum” global ranking policy

topic	words	source corpus
10	game , play , good , year , don’t , lot , week , kind , played, time	transcripts
8	match, yeah, court, tennis, tournament, play, set, playing, played, today	transcripts
18	'state', 'ohio', 'alabama', 'michigan', 'team', 'ohio_state', 'georgia', 'win', 'season', 'championship'	articles
66	'week', 'saturday', 'day', 'jim', 'game', 'night', 'friday', 'monday', 'paul', 'sunday'	articles
3	'tournament', 'team', 'ncaa', 'basketball', 'championship', 'year', 'ncaa_tournament', 'woman', 'season', 'final'	articles
0	'player', 'school', 'sport', 'athlete', 'rule', 'college', 'transfer', 'portal', 'deal', 'medium'	articles
13	'football', 'college', 'game', 'college_football', 'year', 'team', 'season', 'playoff', 'national', 'stadium'	articles
2	'season', 'pitch', 'year', 'freshman', 'player', 'receiver', 'inning', 'defensive', 'spring', 'pitcher'	articles
14	'big', 'texas', '12', 'clay', 'big_12', 'conference', 'oklahoma', 'kansa', 'state', 'baylor'	articles
16	'game', 'quarterback', 'defense', 'football', 'offense', 'week', 'yard', 'play', 'field', 'running'	articles

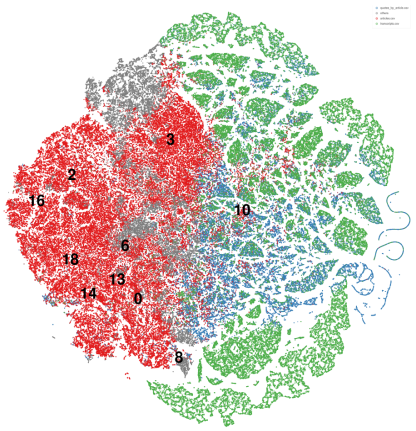


Figure 3. Visual of MTCC after removing conversational language (articles = red, quotes = blue, transcripts = green)

As noted in the data collection section, the proportions of sports are drastically different between articles and transcripts, and in Table 3 we can see that the new top discriminatory topics contain a lot more sport-specific words, especially from the two topics that were discriminatory from the transcript corpus. This is an indication that conversational language differences create more noise than sport variety, and that conversational language was indeed creating unwanted noise in our results.

Comparing Football

We also considered the variety in sports and discrepancy in proportions to be a confounding variable, as this could skew the clustering and discriminatory topics due to sport-specific terminology (ie. football articles mention “quarterback” and “season” a lot while tennis mentions “matches” and “hits”). An example of this bias can be seen upon closer inspection of Figure 3, where the bottom-left cluster’s most probable topic contains the words “pitch”, “series”, “inning” and “hit”, all words clearly associated with baseball. Another cluster toward the bottom contains documents whose most probable topic contains the words “race”, “ran”, “mile” and “time”.

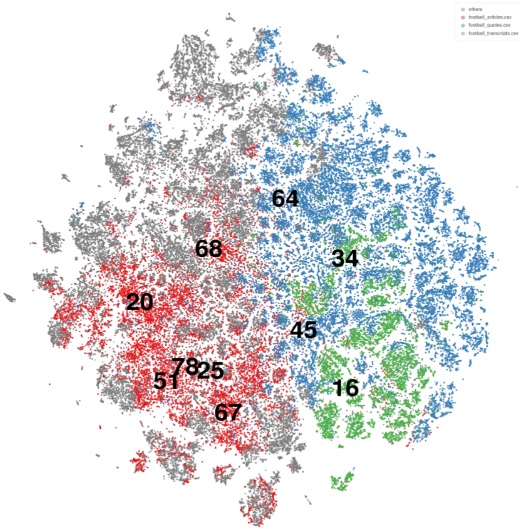


Figure 4. Third run of MTCC after removing all documents unrelated to football (articles = red, quotes = blue, transcripts = green)

For the next iteration of MTCC, we reduced all of the corpora down so that we only used articles, quotes and transcripts related to football. Ideally, this would push other sports out of the most discriminatory topics, and we will get a better glimpse of relevant topic differences between the corpora.

We can see that a lot more of Figure 4 is gray, which indicates a lot less documents are being considered discriminatory than before. Additionally, if we compare the results of the most discriminatory topics before and after reducing the corpora, we can see this occur. Table 3 shows the results when comparing the initial corpora together, where we can see topics with different sport terminologies (topics *8*, *3*, *13*, and , and we can see that 8 of the 10 topics diverge due to the article corpus. Looking at Table 4 below after reducing the corpora to football, we can see a much more even distribution of discriminatory topics across the corpora, and more football related words across the topics.

topic	words	Source corpus
12	'guy', 'game', 'play', 'good', 'team', 'thing', 'lot', 'week', 'kind', 'played'	Transcripts
4	'year', 'great', 'lot', 'guy', 'player', 'coach', 'thing', 'time', 'people', 'day'	Quotes
29	'yard', 'touchdown', 'game', 'rushing', 'pass', 'passing', 'carry', 'running', '10', 'season'	Articles
64	'season', '2021', '2022', 'year', '2020', 'game', '2023', '2019', 'time', 'career'	Articles
7	'season', 'win', 'game', 'team', 'year', 'fisher', 'loss', 'jimbo', '10', 'winning'	Articles
60	'time', 'game', 'team', 'thing', 'people', 'day', 'feel', 'win', 'bad', 'good'	Quotes
61	'dont', 'ill', 'hell', 'care', 'worry', 'kid', 'fortunate', 'people', 'league', 'problem'	Quotes
83	'defense', 'offense', 'ball', 'running', 'play', 'quarterback', 'game', 'offensive', 'throw'	Transcripts
52	'west', 'west_virginia', 'virginia', 'point', '12', 'big', 'game', 'team', 'big_12', 'tournament'	Articles
70	'quarterback', 'season', 'starting', 'starter', 'year', 'qb', 'starting_quarterback', 'offense', 'offensive', 'start'	Articles

Table 4. top 10 most discriminatory topics **after removing all non-football documents**

4b. Sentiment & Emotion Analysis

As was described in the Methodologies section, a few graphs were generated from the extracted sentiment-emotion data. Figure 5 provides a visual comparison of sentiment-emotion between the articles and transcripts corpora.

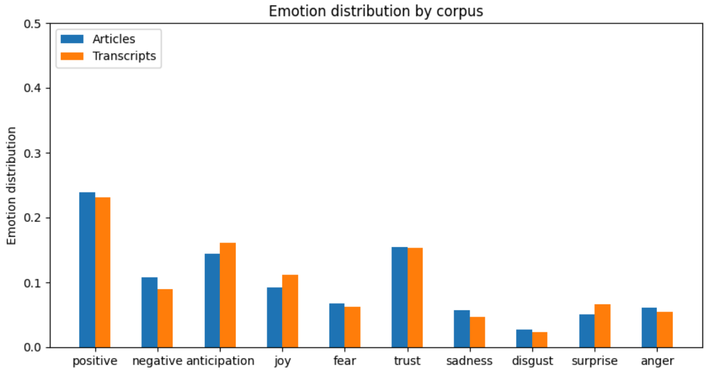


Figure 5: Multi-bar chart of sentiment-emotion distributions of the articles and transcripts corpora.

While both corpora have relatively similar distributions, the observable differences in distribution – while small – deserve emphasis. This is because the data was processed from a large dataset (~130,000 articles, ~77,000 transcripts). Also of note is that these distributions represent the entirety of independent corpora. That is to say, there is no concrete connection between the corpora themselves other than our choosing of data.

In Figure 5, the distribution of "positive" conflicts with that of "negative." For both polarities, the articles corpus has a higher reported distribution. This most likely happened due to the nature of analysis; by using a dictionary-based approach, each word within the documents of each corpus contributed to a "count" of a certain polarity and/or emotion. (See Appendix on Sentiment & Emotion Analysis w/ NRCLex for more details.) This peculiarity is exacerbated by the fact that for every specific negative emotion (fear, sadness, disgust, anger), the article corpus has a consistently higher distribution. When looking at the positive emotions (anticipation, surprise, joy, trust), transcripts rank as much or higher in its distribution.

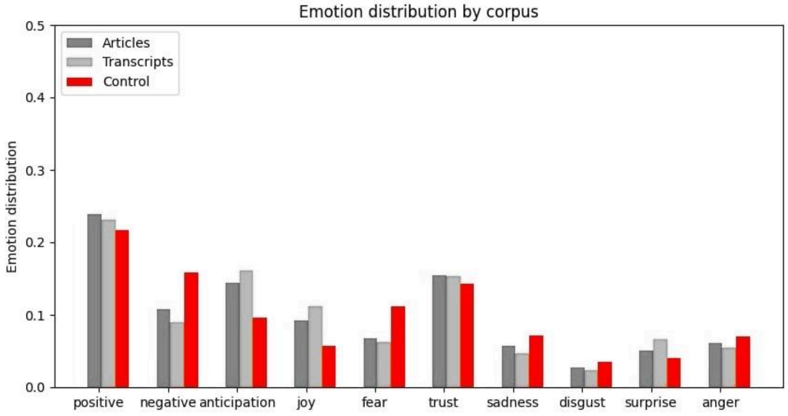


Figure 6: Multi-bar chart of sentiment-emotion distributions of articles, transcripts, and control group corpora. Control group is highlighted to demonstrate its differences from the strictly sports-related corpora.

4c. Matched Articles and Transcripts

We took time to calculate the indices in the combined corpus where matched articles and transcripts occur. Using these indices, we compared two different topic vectors together, and performed analysis on the vectors before and after t-SNE compression to 2 dimensions. The model used to compute these topic vectors are discussed in our Removing Conversation Words section.

For each matched article and transcript pair of vectors, we calculated both the euclidean distance and the cosine similarity between the two vectors[6]. The general trend between these metrics were as expected - that is to say, the larger the euclidean distance, the smaller the cosine similarity. After sorting these pairs by euclidean distance, what we have are pairs of matched articles and transcripts sorted by the difference in vector distance. A few of the top ones turned out not to be matches under manual inspection, but the pair with the largest euclidean distance between the topic vectors was a transcript from Greg Schiano's press conference in New Jersey on October 14 2023 following the Rutgers win against Michigan State (27-24), and an article written by the *New York Post* [16] on the same day. The video of the press conference can be found here: https://www.youtube.com/watch?v=dZl2uhHPQaA [10].

[6] We opted to compute both measures as a way to cross check the vector differences.

Qualitative Comparison of Article vs Transcript

New York Post article	*ASAP Sports* transcript
On a rainy, Homecoming afternoon, one of the greatest comebacks in Rutgers' football history went basically unnoticed… Kyle Monangai ran for 148 yards and had a go-ahead 21-yard touchdown run as the Scarlet Knights rallied from an 18-point fourth-quarter deficit to stun Michigan… Rutgers (5-2, 2-2 Big Ten) got two big plays on special teams and two scores… handing Michigan State (2-4, 0-3) its fourth straight loss under Barnett. Of the announced crowd of 52,879 fans, there might have been a couple of thousand fans on hand to see the turnaround, Rutgers' biggest since 2015. “Those who stuck around in the rain got to see something really exciting,” Rutgers coach **Greg Schiano** said. “Everything that could go wrong in the first three quarters pretty much did.” The final 15 minutes were remarkable, at least for the Scarlet Knights — who outgained Michigan State 120 to minus-20 in the fourth quarter. “Games take a life of their own," Schiano said. "Some people think momentum isn’t real. They're living in a dream. But watch this one today.” Aaron Young started the comeback with 13:09 to play,	Rutgers 27, Michigan State 24 **GREG SCHIANO:** For those that stuck in the round they got to see something exciting, that was really, really good. Really proud of our players. It was definitely, as we say, a 60-minute chop. Everything that could go wrong in the first three quarters pretty much did. And they kept going. It's a great lesson for them in life. It's a football game but as I've always said, I think this is one of the great teachers for young people, this game. So I'll open it up. **Q**. Special teams early this season has struggled in the fourth quarter had some big plays. Can you talk about the resilience and how big those plays were in winning the game? … **Q**. When the momentum changed, you could sense from particularly the defense, those guys were fired up. Were you ahead of the game or were you just letting it flow as the game went on? **GREG SCHIANO**: I think as a coach, you always have to be ahead of the game. Some people think you can make the game go the way you want. I've said to you many times, unfortunately, after losses, that games take on a life of their own. Well, this game took on a life of its own and many times it was in our favor and that happens. Momentum, anybody who doesn't think momentum is real, you know, they are kind of living in a dream world. Because you watch that one today, right, we're having trouble stopping them and all of the sudden they can't gain an inch. How does that happen? There has to be some force. I have my beliefs but I was really happy for our guys.

Fig 7. The highlights indicate the part of the transcript (right) that are being quoted in the article (left).

Looking at these two matched excerpts, we can see some notable differences in topic as well as sentiment. In the press conference Greg Schiano is celebrating a massive comeback by Rutgers by commending his players’ drives and resilience by stating how “everything that could have gone wrong in the first three quarters pretty much did”, yet “they kept going, it’s a great life lesson.”

The *New York Post* article adopts a more pessimistic view on Rutger's victory. The article starts by stating that the win went practically unnoticed, and diving into the low fan retention rate and using the quote “everything that could have gone wrong did” to exacerbate the low turnout, while the quote was initially used to provide context to signify the comeback’s shock.

This discrepancy is similar to a lot of other matched article/transcript pairs we observed, where it is clear that they are talking about the same topic, but the article simply focuses on a single part of the transcript's more broad analysis of the game or event. In this case however, the difference is exacerbated because the article and transcript talk about the event in completely different ways. This conclusion, if shown be a general trend[7]could indicate that topical differences is a valid methodology of identifying context. Although the subjects in which these quotes are referring to sound qualitatively similar, the language that is used when talking about them is different, and could convey a different meaning. Once shown at scale, this result could give us a way in which to quantifiably identify misconstrued context.

4ci. Distributions of Topical Differences

We attempted to show the above qualitative result at scale by graphically analyzing the distributions of these differences. The graphs below show the Euclidean distance and cosine similarity between an outputted topic vector of an article, and its matched transcript, grouped into a countable bar chart.

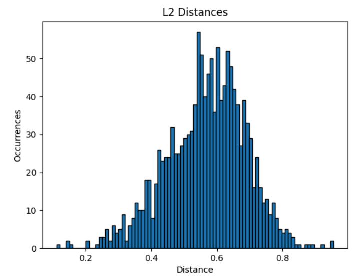
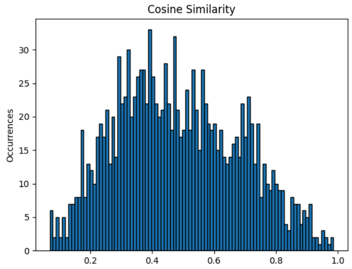
Legend	L2 Distance	Cos Similarity
Bar Graph **y-axis** : count **x-axis**: difference		

Figure 8. Bar graph of the count of topical distance and cosine similarity between matched transcripts and articles

By representing the differences in this manner, we are able to infer, graphically, how *different* article(s) are when compared to their respective transcript(s). If articles are, on average, topically close[8] to the transcripts that they’re quoting from, then we should see the L2 exhibiting a leftward skew (i.e. the distance between topic vector of an article and its matched transcript is small) while the Cosine Similarity graph should skew to the right (i.e. the cosine angle between the matched topic vectors is close to one). While we performed analysis mainly using the L2 distance, we used Cosine Similarity for qualitative assurance. Since we do not see this behavior in the distributions, we offer two hypothesis that would suggest these distributions:

1. **If *topic* is a good indicator for context, thus a large topical difference is a good indication for quote misrepresentation, then on average articles aren’t very likely to take quotes out of context.** That is, since the majority of our data exhibit *medium sized* topical differences, the qualitative result we’re able to gather (see Matched Articles and Transcript) of an *out-of-context* quote is very unlikely to happen. It also tells us that articles are very likely to diverge, at least in

[7] We are unfortunately unable to repeat these findings at scale due to time constraints.
[8] That is, the outputted topic vectors are closer together.

some parts[9], from the original transcript but not enough to the point of complete misrepresentation.

2. **If *topic* is a good indicator for context, thus a large topical difference is a good indication for quote misrepresentation, then misrepresentation, although rare, happens when the topical difference is large**. That is, when we see an outrageous topical difference (i.e. the small bars to the very right of the L2 graph, or to the left of the Cosine graph), then there is a good chance that there is some misrepresentation going on and we should proceed with a qualitative analysis of the matched pair.

The following four scatter plots show the same trend, but gives us a little bit more insights into what kind of topical differences are expected of our dataset. To construct these plots, we simply sorted the computed differences (once by the cosine similarity and the other by Euclidean distance) and simply plotted those indices in order (i.e. the leftmost index is the pairing of transcript and article with the smallest difference). Although the majority of our dataset exhibits ambiguous topical difference, the sharp inflection points of the highest difference (seen in L2 Distance graph sorted by L2) indicated that the most diverging topical pairings are *very* different. This was how we were able to identify the article and transcripts used in our qualitative analysis, in which one of them was talked about in Matched Article and Transcript, and also offer additional support to the second hypothesis we discussed in the previous paragraph.

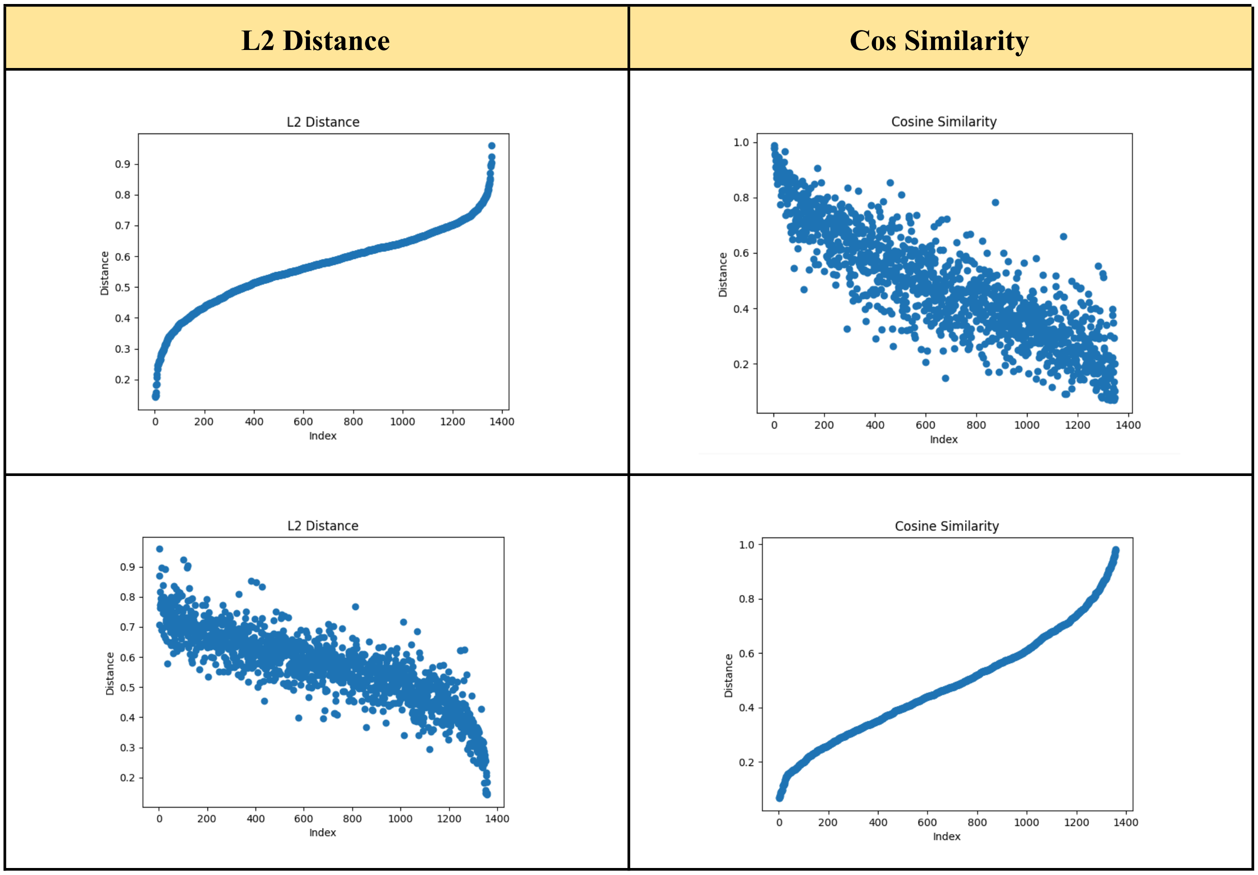


Fig 9. Scatter plot of topical difference and cosine similarity between matched transcripts and articles.

[9] If the articles and transcripts are talking about the exact same thing, then readers would simply skip the articles and only read interview transcripts.

4cii. Emotion & Sentiment Comparison

In the following section, we will analyze and visually describe any patterns between the sentiment-emotion distributions of matched articles and transcripts. This is different from our initial sentiment-emotion analyses in the sense that we're observing how the same event is represented emotionally depending on the document describing it.

Each scatter plot below (Figures 10 to 19) shows the distribution of a single polarity/emotion among matched article-transcript pairs. Each point on each plot represents such a pair, with the x-axis representing the distribution within the article and the y-axis representing that within the transcript.

Figure 10

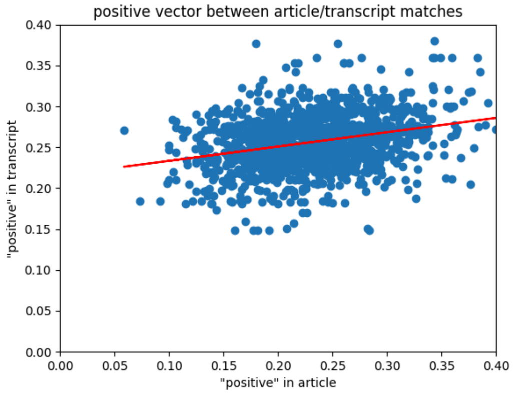


Figure 11

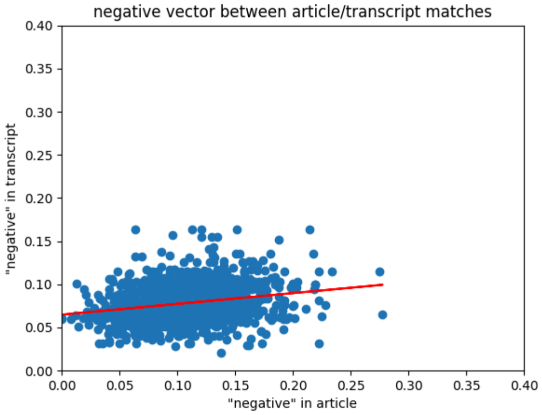


Figure 12

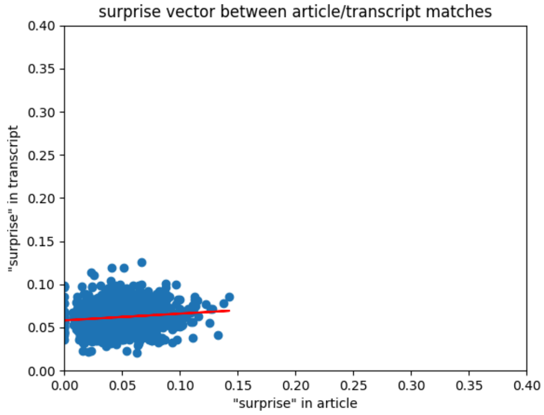


Figure 13

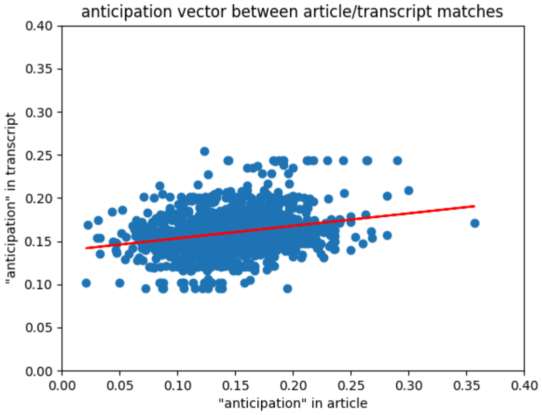


Figure 14

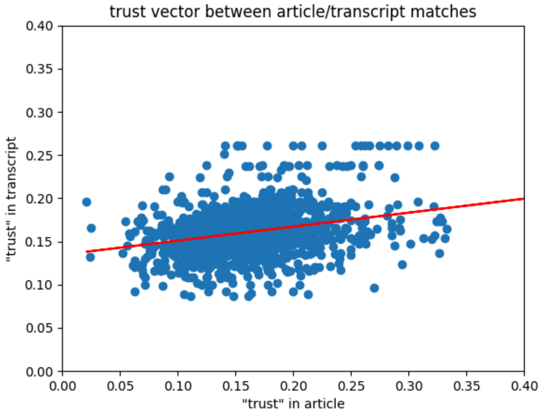


Figure 15

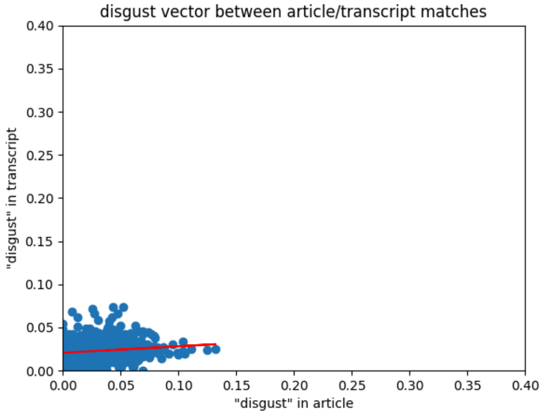


Figure 16

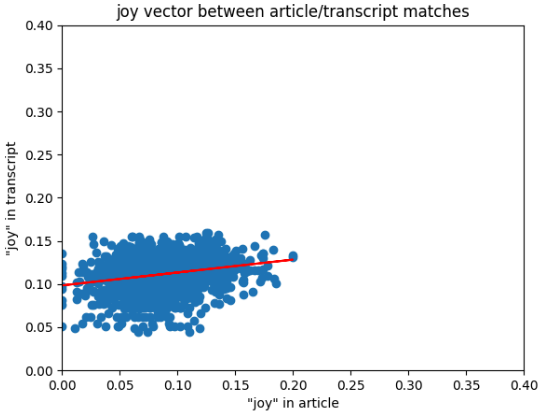


Figure 17

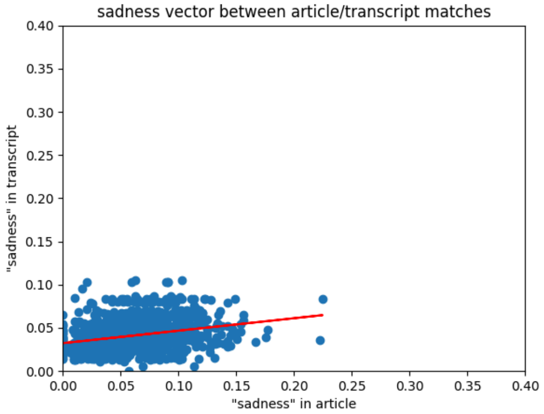


Figure 18

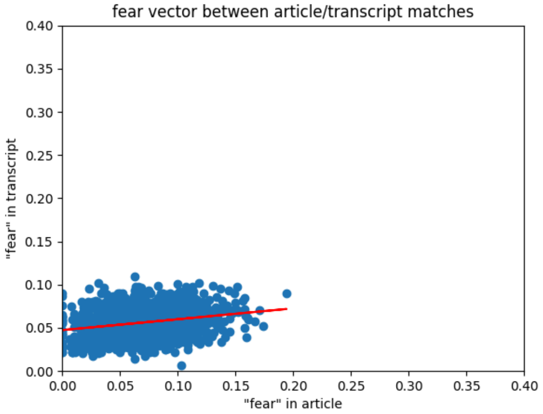
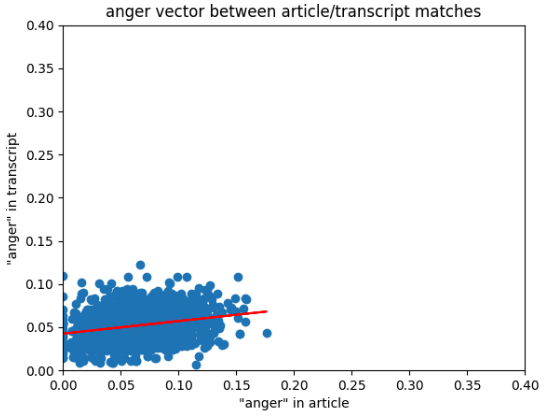


Figure 19



For every observed emotion (both positive and negative), we notice that the positive slope of our trendline (red) is small[10]. This would indicate that the articles express a larger spread of every emotion, which would lead us to the following conclusion about the sentiment across sports transcripts and articles:

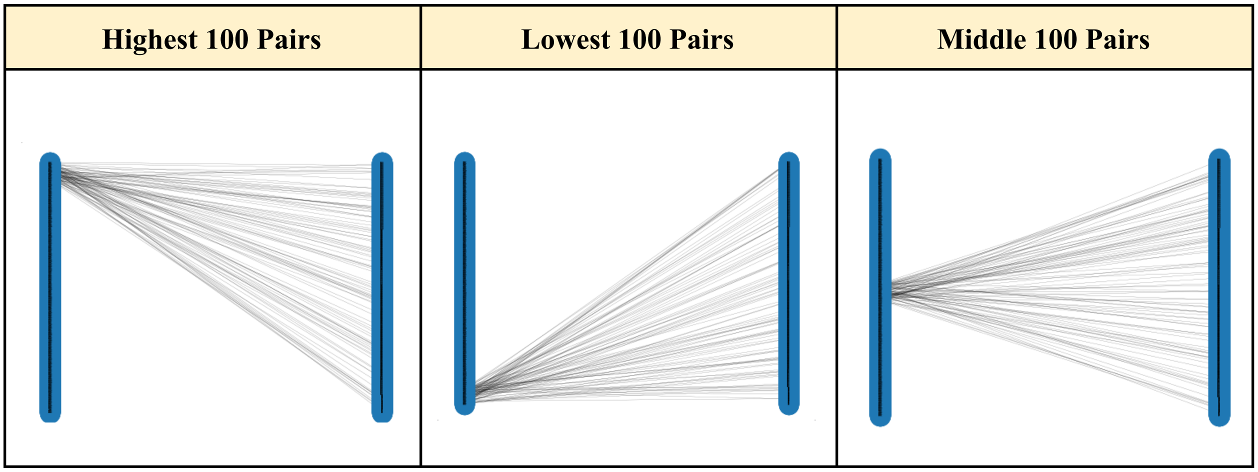
[10] It is closer to the horizontal, or *article* axis than it is to the vertical, *transcript* axis.

- **The small slope value would indicate that the spread of sentiment emotions across transcripts is smaller than the spread of sentiment across articles.** That is to say, transcripts are typically closer to each other in emotions than articles are. This would point to the possibility that articles could contain a bigger range of emotions[11] while transcripts does not exhibit the same behavior[12].
- **The positive correlation would indicate that sentiment between transcripts and articles are typically consistent.** That is, if an athlete or coach gives a generally negative interview, then the article that is quoting them is also likely to contain these negative emotions as well.

Our results would point to the possibility of there being a small difference between the sentiment of articles and the transcripts in which they are quoting from, and in the following section we will discuss the result observed when combining these sentiment values with our topical output of MTCC.

4ciii. Combining Topic and Sentiment Analysis

We concluded our research by attempting to find a correlation between topic difference and emotional difference in matched articles and transcripts using a bipartite graph. We sorted 1,350 matched pairs in ascending order by euclidean distance[13] (left side, descending top to bottom), and again by average emotion difference (right, descending top to bottom), and connected the corresponding matches from both sets. It was difficult to see trends with 1,350 lines in the visual, so we created three graphs in Figure 20 displaying lines for the highest, middle, and lowest 100 *L2 topical* differences (left).



Highest 100 Pairs

Lowest 100 Pairs

Middle 100 Pairs

Fig 20. Bipartite graphs connecting matched article/transcript pairs between topical L2 rankings and sentiment difference rankings

These graphs show no correlation between topic euclidean distance and average emotional distance, and that even if the topical differences between the articles and the transcripts in which they've quoted from are high, **it does not mean that we will see a high sentimental difference between them.** That is, even if both topics and sentiments are good indicators for context (i.e. these quantities capture misrepresentation when the differences are high, and accurate representation when the differences are low), there likely exists no visual correlation between one measure and the other.

[11] Different writers could write about the same event, but the emotions behind their writings can vary drastically

[12] Athletes are often media prepped, which often results in them being less expressive in their responses.

[13] This was decided arbitrarily since both measures (Cosine angle and Distance) exhibit the same general trend.

5. Discussion

In the following section, we will discuss our result with respect to our overall hypothesis. That is, we will answer the question of "does our analysis of this sport dataset answer our overarching question".

5a. Conclusions

We have a few notable characteristics of our findings as they relate to analyzing quotes in context. From topic modeling, there is a clear language style difference between transcripts and articles that creates irrelevant noise, but the quotes that articles use are far more similar to *spoken language* found in transcripts. This is important to note to consider whether articles are an accurate representation of what athletes have to say on certain matters. We cannot definitively say that articles keep quotes in or out of "context" by this criteria, and our findings indicate there are some differences between the topics and emotions in transcripts and articles.

There are a number of potential reasons for these differences, including a) the different language styles of articles vs transcripts, b) the sources of the text of articles and transcripts, and c) the ways in which computers are able to analyze bodies of text in a qualitative way. The matched results, which we analyzed qualitatively and quantitatively, produced some key insights into why our findings behaved the way they did. Most of the transcripts that were the farthest distance apart from their matched article covered a wide range of topics whereas the article only mentioned one or two, while pairs that were closer in distance were because the transcripts were much shorter and didn't talk about a large variety of topics. These results align with the quantitative findings from the general unmatched corpora; there are topical differences between the two corpora but these differences aren't due to quotes being taken out of context. Rather, they indicate differences in written versus conversational language as well as an inherent topical difference between articles and transcripts. This means that our qualitative analysis of the matched documents is likely a good sample of the overall relationship between articles and transcripts concerning quote usage.

Through sentiment analysis, we found that on average, articles are more emotionally charged than transcripts. Articles are both more negative and more positive than transcripts. However, when looking at specific positively and negatively connotated emotions like fear (negative) and joy (positive), articles have a higher percentage of negative emotions and a lower percentage of positive emotions. Additionally, articles have a larger range of emotions. Transcripts varied in most emotions by ten percent whereas articles varied by up to twenty percent. Again, these findings could be due to the inherent differences between articles and transcripts and do not show quotes being taken out of context.

5b. Why Do We Care?

Most casual sports fans have come across at least one example of a quote being taken out of context. If fans can recognize that an article is misleading, it isn't an issue. But there are also cases where fans may not *know* a quote is being taken out of context, which, when applied on a mass scale, creates a big issue. In an ideal world, every sports article would have all the necessary context needed to fully understand what the athlete or coach meant to say. There is more media and greater access to media in circulation in our society than ever before. As such, the need for truthful news is becoming increasingly important and sought after. Even in the lighthearted world of sports, information can be misconstrued. It is not hard to imagine a tool that provides additional context for misleading quotes and articles. This study is meant as a small building block towards achieving this goal.

5c. Did We Achieve Our Goals?

We wanted to determine if quotes from athletes and coaches are taken out of context by articles. To do so, we compared articles to transcripts using two NLP techniques: topic modeling and sentiment analysis. Results from these methods gave us some insight into the differences in *what* and *how* articles and transcripts talk about quotes. While we cannot give definitive answers as to if, when, and how articles take quotes out of context, we hope that this paper serves as a starting point for this emerging field of study.

6. Limitations

6a. Data Limitations

A big limitation we had was data we had access to. After many dead ends for finding large databases of articles and transcripts, we eventually stuck with Perigon and ASAP Sports. As we relied on two separate sources for our articles and transcripts, we didn't necessarily have transcripts that were referenced in the articles we had. While we could perform analysis on the two corpora as a whole, the most interesting analysis is done on an article and transcript that are talking about the same thing. We hope our analysis serves as a starting point for later iterations of this study, and that future research will have the luxury of more matched articles and transcripts.

The fact that we defined transcripts as the context to articles is a limitation in and of itself. Given our introductory definition of context, we know that events preceding quotes can be a large part of the true context, and gaining a complete knowledge of the preceding events for every single article and transcript would've been a near impossible feat given our time constraints. The transcript for the quote is a good starting point for getting context from the conversation itself, but context for an event doesn't just start there. We debated about using game results preceding interviews as additional context, however we noticed that typically the sentimental values of words in an article or transcript could typically paint a picture of the results of the event.

6b. Limitations of Matching and Match Verification

Our data collection was centered around finding a collection of transcripts that had a good chance of providing context to the quotes found in related articles. Using our dataset, *thefuzz* was able to find around 1,400 pairs of articles and transcripts with 95% accuracy. This accuracy only stems from the quote words itself, however. We found that even after limiting matched quotes to being greater than 8 words, some common quotes could be the same word for word, but from different sources. One way we could have reduced this rare inconsistency would have been to incorporate other metadata into the matching process (date and time, speakers/subjects using named entity recognition, locations, etc), but unfortunately we did not have enough time to explore this type of matching verification. Additionally, the vast majority of these matches came from college football articles/transcripts, and thus our matches were constrained to only this topic, limiting how effective our analysis of matched data could be.

7. Further Research

7a. MTCC Hyperparameters

Most of our analysis focused on how we can change the input for topic modeling and sentiment analysis in order to yield results related to quote misuse. We disregarded the tuning hyperparameters provided to us by LDA. Namely, the alpha value could have been tuned to a low number because sport related media are likely to only contain a few topics. We think that setting this parameter would give a much clearer difference between outputted topic vectors of matched quotes and transcripts, which could perhaps give our paper a more conclusive finding. Additionally, there were other probability distribution comparison methods and global ranking policies of JSD values we could have used for MTCC, but we chose not to explore those in depth due to the original paper noting a dissimilarity between the methods.

7b. Defining Context Even Further

Topics and sentiments do not seem quite as representative of context as we had initially hoped for. Though these are easily quantifiable measures, they don't seem to capture what we've hoped to be a quantity measuring *context*. In future iterations of this project, we suggest working with linguists and/or other humanities departments to properly define what it is we're trying to measure using these computational techniques.

7c. Multicorpus Sentiment Analysis

We recommend running at least Jensen-Shannon divergence on the various sentiment outputs of NRC-Lex. This will allow us to infer, more concretely, on the diverging sentiments between our various corpora (i.e. transcripts and articles). This process should be relatively simple, that is to say, simply replace the first step of training the LDA models within MTCC with a QRC-Lex instead. The rest of the processes can stay the same.

7d. Better Data Collection

We can not stress enough how important the data is to our project. Since the majority of our analysis was done on transcripts and articles taken from very niche sources, our results were subjected to a lot of noise due to the scarcity of matched transcripts and articles data. In future iterations of this project, we suggest finding a more "matched" corpus, which we believe would significantly change the result.

Citations

[1]

J. Lu, M. Henchion, and B. M. Namee, "A Topic-Based Approach to Multiple Corpus Comparison".

[2]

"ASAP Sport." Accessed: Mar. 11, 2024. [Online]. Available: https://www.asapsports.com/

[3]

S. M. Mohammad and P. D. Turney, "Crowdsourcing a Word-Emotion Association Lexicon." arXiv, Aug. 28, 2013. Accessed: Mar. 11, 2024. [Online]. Available: http://arxiv.org/abs/1308.6297

[4]

Laurens van der Maaten and Geoffrey Hinton, "Visualizing Data using t-SNE." Journal of Machine Learning Research, Nov. 11, 2008. Accessed: Mar. 11, 2024. [Online]. Available: https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf

[5]

"Dirichlet distribution," ***Wikipedia*****. Jan. 05, 2024. Accessed: Mar. 11, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Dirichlet_distribution&oldid=1193761832**

[6]

"Fuzzy String Matching in Python Tutorial." Accessed: Mar. 11, 2024. [Online]. Available: https://www.datacamp.com/tutorial/fuzzy-string-python

[7]

"Gensim: topic modelling for humans." Accessed: Mar. 11, 2024. [Online]. Available: https://radimrehurek.com/gensim/parsing/preprocessing.html

[8]

"Gensim: topic modelling for humans." Accessed: Mar. 11, 2024. [Online]. Available: https://radimrehurek.com/gensim/models/ldamodel.html

[9]

"Gensim: topic modelling for humans." Accessed: Mar. 11, 2024. [Online]. Available: https://radimrehurek.com/gensim/models/phrases.html

[10]

Greg Schiano Michigan State Post-Game Press Conference 10/14/23**, (Oct. 14, 2023). Accessed: Mar. 11, 2024. [Online Video]. Available: https://www.youtube.com/watch?v=dZl2uhHPQaA**

[11]

D. M. Blei, "Latent Dirichlet Allocation".

[12]

"NLTK :: nltk.stem.wordnet module." Accessed: Mar. 11, 2024. [Online]. Available: https://www.nltk.org/api/nltk.stem.wordnet.html?highlight=wordnetlemmatizer

[13]

"NLTK Data." Accessed: Mar. 11, 2024. [Online]. Available: https://www.nltk.org/nltk_data/

[14]

262588213843476, "NLTK's list of english stopwords," Gist. Accessed: Mar. 11, 2024. [Online]. Available: https://gist.github.com/sebleier/554280

[15]

"quote-extraction/regex_pipeline at main · JournalismAI-2021-Quotes/quote-extraction," GitHub. Accessed: Mar. 11, 2024. [Online]. Available: https://github.com/JournalismAI-2021-Quotes/quote-extraction/tree/main/regex_pipeline

[16]

"Rutgers rallies from 18-point deficit to stun Michigan State." Accessed: Mar. 11, 2024. [Online]. Available: https://nypost.com/2023/10/14/rutgers-rallies-from-18-point-deficit-to-stun-michigan-state/

[17] "The Levenshtein distance algorithm," Educative. Accessed: Mar. 11, 2024. [Online]. Available: https://www.educative.io/answers/the-levenshtein-distance-algorithm

[18] "Word representations · fastText." Accessed: Mar. 11, 2024. [Online]. Available: https://fasttext.cc/index.html

[19] "What is Natural Language Processing? | IBM." Accessed: Mar. 12, 2024. [Online]. Available: https://www.ibm.com/topics/natural-language-processing

[20] "Perigon" Accessed: Mar. 11, 2024. [Online]. Available:https://www.goperigon.com/

8. Appendix

Data Collection: Initial Attempt

Initial Data collection plan:

1. Scrape an article off of ESPN
2. Extract the quotes using Journalism AI's Quote Extractor
3. Use the quotes/speaker to produce Youtube search queries
4. Scrape the transcript from the correct Youtube video
5. Match this transcript to the scraped article

Our goal when making this pipeline was to make a generic article/transcript matcher, where when given *any* article (regardless of subject), we could extract the quotes and find the transcript that matches the quotes. Using Youtube was our first approach at making a transcript finder for universal topic articles

We had plans to scrape various news outlets[14], but we were advised against doing so without express permission from these outlets themselves. We reached out to Fox Sports, Disney, the New York Times, and The Associated Press, and of these four, we heard back from Disney and the New York Times, thanks to Kevin Draper (Carleton College '10). With his help, we were able to get access to an NYT internal API and a meeting with Aaron Laberge, The Walt Disney's Company acting CTO. We ended up speaking with Michael Cupo, a senior VP of Technology and Business Strategy at ESPN who promised us access to API content. The NYT API only supplied us with articles summary and abstract, and we never received any follow-up from Cupo.

Additional Limitations

Limitations to Quote Extractor

Main Limitations:

1. Regular expressions used to find quotes *relied* on limited cue verb list. We had to add new words to this given list to handle more cases:

   Regular expression example from extractor: (“[^“\n]+?[,?!]”)({cue_verbs})

   - (“[^“\n]+?[,?!]”): a quote has an open quotation mark, a closing quotation mark, and no other new lines or open quotation marks in between them,
   - ({cue_verbs}): a verb used to describe how a subject says the quote (“said” someone, “mentioned” someone)
2. Quotation marks don't always represent quotes (can designated titles, nicknames, definitions, etc)
3. Extractor relied on specific open/close Unicode quotation marks to perform extraction, was not initially compatible with keyboard quotation marks or other variations

[14] A non-exhaustive list includes ESPN, Fox Sports, Sports Illustrated, Bleacher Reports, Yahoo Sports, NBC Sports, The Athletics, New York Times, and Associated Press.

MTCC Limitations

Coherency score

MTCC uses semantic coherency to choose the optimal number of topics to build a model with. MTCC is implemented using a semantic coherency metric, which determines how semantically related two words are. We can think of each word having a vector in “semantic space.” For example, “football” and “interception” would have a high semantic coherency score while “football” and “think” would have a low semantic coherency score.

We implemented semantic coherency by considering all unique pairs of words in the top ten most probable words of each topic. For each pair, we computed the semantic coherency and averaged them out to get a single semantic coherency metric for each topic. The semantic coherency values are computed using cosine similarity based on pre–trained vectors of each word. These vectors are trained using Facebook data.

Our topic coherency scores are mostly in the range of 0.1 – 0.2. While it is not definitive whether this is a “good” or “bad” score because semantic coherence scores differ by subject, data, etc., this indicates a low overall topic coherency.

Instability of LDA

The instability of LDA is inherent to the probabilistic nature of the technique itself, and in this section we will demonstrate this instability with our dataset. In order to simulate this instability, we took roughly 15% of our total dataset, and ran 5 MTCC runs using the same parameters. Here are the resulting coherency scores of these various runs. The k values chosen were 5, 10, and 20.

Seed value 1500		Seed value 1912		Seed value 1969	
5	0.02652	5	0.02713	5	0.02700
10	0.01205	10	0.02188	10	0.02329
20	0.00863	20	0.01046	20	0.00874

Seed value 2001		Seed value 2024	
5	0.02374	5	0.02440
10	0.01221	10	0.02323
20	0.00514	20	0.01007

Although the most coherent topic values (i.e. most coherent *k* value) did not change due to chance, the change in differences between *k = 10* and *k = 5* for seed values 1912, 1969, and 2024 when compared to seed value 1500 and 2001 indicate that MTCC’s outlined method of identifying most probable *k* value is unstable.

We also repeated the same experiment, but this time increasing the pass count to 20 (default pass count of Gensim is 10 when training LDA models), and the same pattern persists.

Seed value 1500		Seed value 1912		Seed value 1969	
5	0.02652	5	0.02713	5	0.02700
10	0.01622	10	0.02024	10	0.02108
20	0.00734	20	0.01289	20	0.01434

Seed value 2001		Seed value 2024	
5	0.02374	5	0.02440
10	0.01409	10	0.02201
20	0.00634	20	0.01300

Comparing Topic Distributions using Jenson-Shannon

A common way of comparing two corpora We can compare topic distributions using Jenson Shannon divergence. To compare two corpora, we can sum up the document/topic vectors to create an aggregate topic distribution vector for both corpora.

Given two *k*-dimensional sum vectors X and Y , we can treat these as probability distributions in Jensen-Shannon to get a divergence score for each of the *k* topics. To get the JSD value for topic *i*, we use the following four values:

1. $A = X[i]$
2. $B = (\sum_{n=1}^{k} X[n]) - A$
3. $C = Y[i]$
4. $D = (\sum_{n=1}^{k} Y[n]) - C$s
5. $N = A + B + C + D$ (sum of both vector magnitudes)

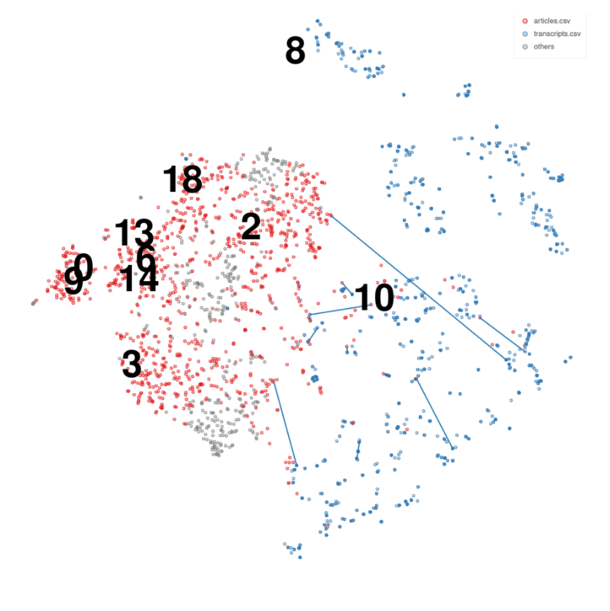
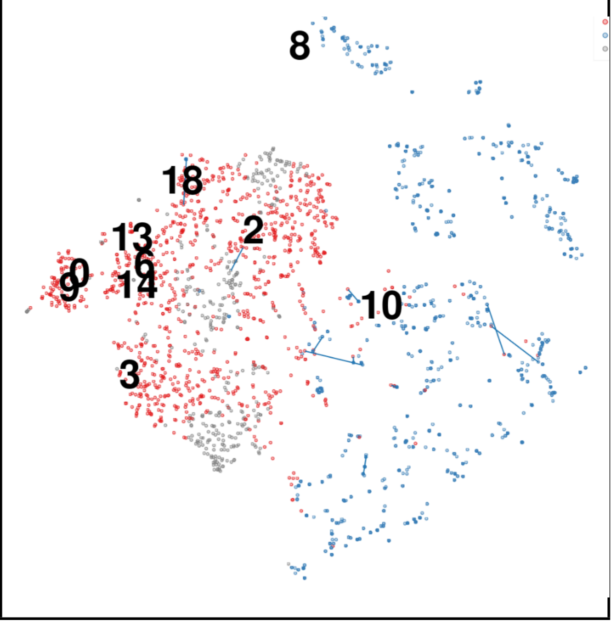
Using these values, we can compute the Jensen-Shannon divergence between vectors *X* and *Y* for topic *i* using the following formula:

$$JSD_i = -\frac{A+C}{N} log_2(\frac{A+C}{N}) + \frac{A}{N} log_2(\frac{A}{A+B}) + \frac{C}{N} log_2(\frac{C}{C+D})$$

MTCC utilizes a “one-vs-all” approach in a similar fashion to compare each corpus $C \in S$ to corpus $S' = S - C$. Using this method, for each topic t, we obtain a JSD value for topic t in *terms* of each singled out corpus. Given *n* JSD values for each topic, to get a single score for each topic, we can settle on a divergence score metric to rank each topic. The default is using the *maximum* JSD value calculated out of all of the *n* values available for a given topic (other aggregation options included getting a sum of the JSD values, as well as a weighted sum, which takes into account the number of documents of each corpus). If maximum is used, then if a particular topic is considered discriminatory, then there is one corpus out of the *n* corpora whose aggregate topic vector is *very* different from the rest of the corpora in terms of that particular topic. We can then say that the highest valued topics out of the *k* are considered the most *discriminatory* topics.

PCA and t-SNE Visualization

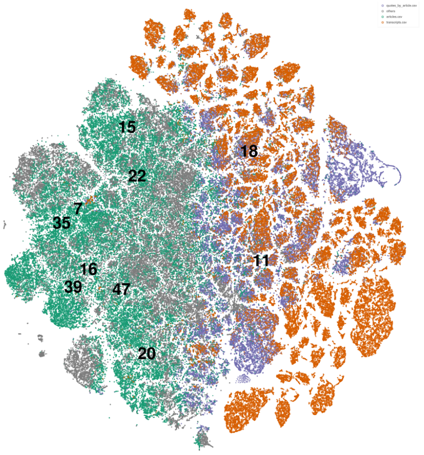
Using t-SNE [4], we were able to compress our *k-dimensional* topic distribution vectors into a 2D representation for easy visualization. Though the t-SNE algorithm is good at preserving pairwise similarities between our compressed vectors, the compression will exaggerate the topical differences between the transcripts and articles. The figures below are meant to aid in visualizing the discrepancies between compressed and uncompressed topic vectors.

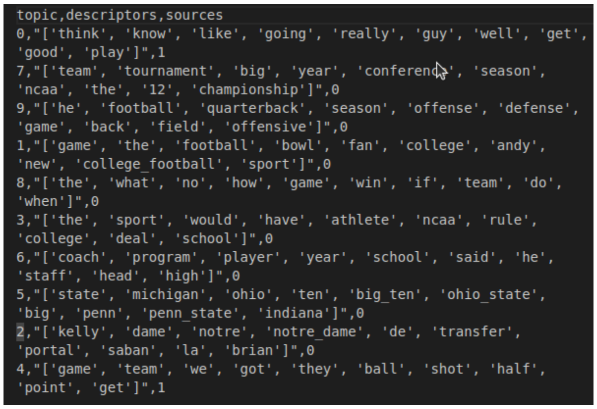
Shortest distances Before tSNE computation (20d)	**Shortest distances After tSNE computation (2d)**
	

These figures are the shortest topical difference between a pair of transcript and article (computed using Euclidean distance). The figure on the left features multiple long, distinct lines that connect a red (article) to blue (transcript) dot whereas the lines on the right figure are noticeably shorter. This indicates that after t-SNE, articles and transcripts may be clustered away from one another even if they share similar topic probabilities.

Additional MTCC Results

Articles vs transcripts vs quotes (bigrams)





```
topic,descriptors,sources
0,"['think', 'know', 'like', 'going', 'really', 'guy', 'well', 'get',
'good', 'play']",1
7,"['team', 'tournament', 'big', 'year', 'conference', 'season',
'ncaa', 'the', '12', 'championship']",0
9,"['he', 'football', 'quarterback', 'season', 'offense', 'defense',
'game', 'back', 'field', 'offensive']",0
1,"['game', 'the', 'football', 'bowl', 'fan', 'college', 'andy',
'new', 'college_football', 'sport']",0
8,"['the', 'what', 'no', 'how', 'game', 'win', 'if', 'team', 'do',
'when']",0
3,"['the', 'sport', 'would', 'have', 'athlete', 'ncaa', 'rule',
'college', 'deal', 'school']",0
6,"['coach', 'program', 'player', 'year', 'school', 'said', 'he',
'staff', 'head', 'high']",0
5,"['state', 'michigan', 'ohio', 'ten', 'big_ten', 'ohio_state',
'big', 'penn', 'penn_state', 'indiana']",0
2,"['kelly', 'dame', 'notre', 'notre_dame', 'de', 'transfer',
'portal', 'saban', 'la', 'brian']",0
4,"['game', 'team', 'we', 'got', 'they', 'ball', 'shot', 'half',
'point', 'get']",1
```

In order to see whether connecting words has an effect on our outputted topic vectors, we computed bigrams using the same corpus as outlined in our First Run section. The bigram computation process is as follows:

Bigram computation

We used Gensim's Phraser model to compute our Bigrams using the ENGLISH_CONNECTOR_WORDS list [9]. This Phraser model is an implementation of Mikolov et. al. Phrase Skip Gram model (https://proceedings.neurips.cc/paper_files/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf) . Discussing the mechanism behind Phrase Skip Gram is beyond the scope of this report, but the basic idea between bigram classification is behind this mathematical formula:

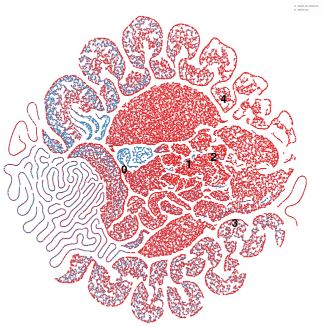
$$score(w_i, w_j) = \frac{count((w_i w_j) - \delta}{count((w_i) \times count(w_j)}$$

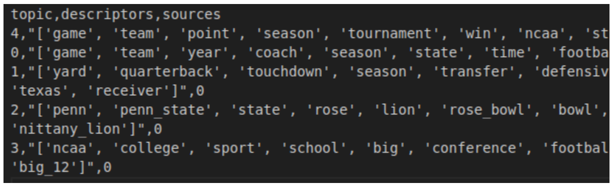
Where score(w_i,w_j) is just the number of occurrences of word w_i and w_j occurring together. The higher the score, the more likely that "w_i_w_j" is a bigram in our corpus.

Here is a summary of what we've found:

- Most coherent model was trained with *k = 10*, out of these values *k = 5, 10, 20, 50, 100* (see Stability of MTCC).
- The most divergent topics are very similar to those found in the unigram version of this model [LINK TO FIRST RUN]. I.e They are topics that contain words about a team, conference, or sport-specific terminologies.
- The grouping of the quotes (purple) is less prominent than it was with the unigram model. Which could indicate that a lot of the distinct clustering seen in the unigram model was due to common connector words (i.e. Texas University would be a bigram).

Articles vs quotes



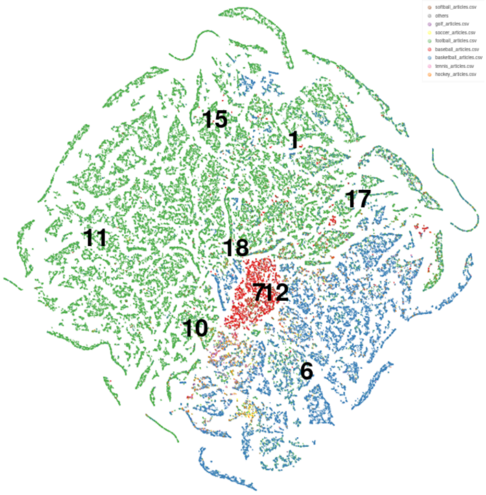


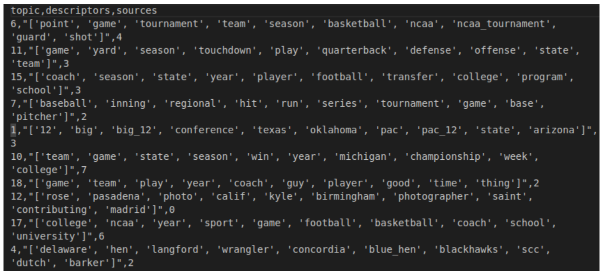
```
topic,descriptors,sources
4,"['game', 'team', 'point', 'season', 'tournament', 'win', 'ncaa', 'st
0,"['game', 'team', 'year', 'coach', 'season', 'state', 'time', 'footba
1,"['yard', 'quarterback', 'touchdown', 'season', 'transfer', 'defensiv
'texas', 'receiver']",0
2,"['penn', 'penn_state', 'state', 'rose', 'lion', 'rose_bowl', 'bowl',
'nittany_lion']",0
3,"['ncaa', 'college', 'sport', 'school', 'big', 'conference', 'footbal
'big_12']",0
```

To confirm whether the topic distributions of quotes are different from the articles in which they're seen in, we removed the transcript corpus and ran MTCC. Here is a list of our notable findings:

- Most coherent model was trained with *k* = *5*, out of these values *k* = *5, 10, 20, 50, 100* (see Stability of MTCC).
- The clustering of outputted topics are blended together, which could indicate that the quotes themselves are at least similar, topic-wise, to the articles in which they originate from. This is expected as the quotes are a direct subset of the articles themselves because we did not remove the quotes from our articles' text.
- The most divergent topics are found in the article corpus, which we hypothesize stemmed from the fact that articles often give commentaries on the quotes themselves, rather than just directly reporting on the quotes. This however does not give us conclusive evidence on what the articles are talking about, and even less so when dealing with *context*.

Articles separated by sport



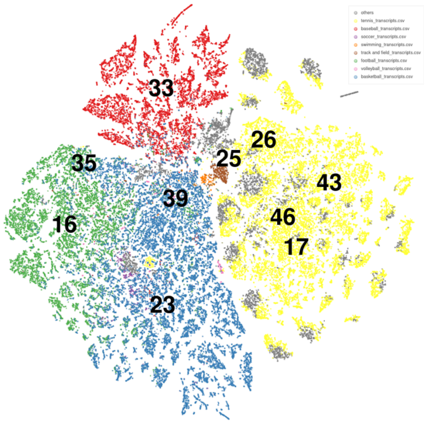


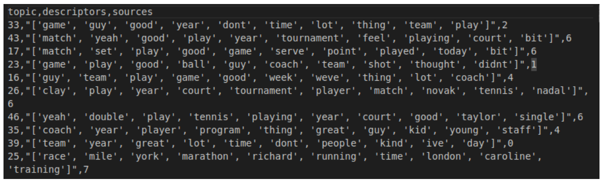
```
topic,descriptors,sources
6,"['point', 'game', 'tournament', 'team', 'season', 'basketball', 'ncaa', 'ncaa_tournament',
'guard', 'shot']",4
11,"['game', 'yard', 'season', 'touchdown', 'play', 'quarterback', 'defense', 'offense', 'state',
'team']",3
15,"['coach', 'season', 'state', 'year', 'player', 'football', 'transfer', 'college', 'program',
'school']",3
7,"['baseball', 'inning', 'regional', 'hit', 'run', 'series', 'tournament', 'game', 'base',
'pitcher']",2
1,"['12', 'big', 'big_12', 'conference', 'texas', 'oklahoma', 'pac', 'pac_12', 'state', 'arizona']",
3
10,"['team', 'game', 'state', 'season', 'win', 'year', 'michigan', 'championship', 'week',
'college']",7
18,"['game', 'team', 'play', 'year', 'coach', 'guy', 'player', 'good', 'time', 'thing']",2
12,"['rose', 'pasadena', 'photo', 'calif', 'kyle', 'birmingham', 'photographer', 'saint',
'contributing', 'madrid']",0
17,"['college', 'ncaa', 'year', 'sport', 'game', 'football', 'basketball', 'coach', 'school',
'university']",6
4,"['delaware', 'hen', 'langford', 'wrangler', 'concordia', 'blue_hen', 'blackhawks', 'scc',
'dutch', 'barker']",2
```

To investigate whether topical differences are related to the sport that articles are covering, we grouped our article corpus by sport and ran MTCC on the new groupings. Here is a summary of those findings:

- Most coherent model was trained with *k = 10*, out of these values *k = 5, 10, 20, 50, 100* (see Stability of MTCC)
- As expected, we see a clustering effect of these articles separated by the sport they're covering. We think that this happened because of the sport-specific words being used in each of these articles (e.g. "touchdown" is a football specific term and isn't used in other sports).

Transcripts separated by sport





```
topic,descriptors,sources
33,"['game', 'guy', 'good', 'year', 'dont', 'time', 'lot', 'thing', 'team', 'play']",2
43,"['match', 'yeah', 'good', 'play', 'year', 'tournament', 'feel', 'playing', 'court', 'bit']",6
17,"['match', 'set', 'play', 'good', 'game', 'serve', 'point', 'played', 'today', 'bit']",6
23,"['game', 'play', 'good', 'ball', 'guy', 'coach', 'team', 'shot', 'thought', 'didnt']",1
16,"['guy', 'team', 'play', 'game', 'good', 'week', 'weve', 'thing', 'lot', 'coach']",4
26,"['clay', 'play', 'year', 'court', 'tournament', 'player', 'match', 'novak', 'tennis', 'nadal']",
6
46,"['yeah', 'double', 'play', 'tennis', 'playing', 'year', 'court', 'good', 'taylor', 'single']",6
35,"['coach', 'year', 'player', 'program', 'thing', 'great', 'guy', 'kid', 'young', 'staff']",4
39,"['team', 'year', 'great', 'lot', 'time', 'dont', 'people', 'kind', 'ive', 'day']",0
25,"['race', 'mile', 'york', 'marathon', 'richard', 'running', 'time', 'london', 'caroline',
'training']",7
```

To investigate whether topical differences are related to the sport that transcripts originate from, we grouped our transcript corpus by sport and ran MTCC on the new groupings. Here is a summary of those findings:

- Most coherent model was trained with *k = 10*, out of these values *k = 5, 10, 20, 50, 100* (see Stability of MTCC).
- As expected, we see a clustering effect of these transcripts separated by the sport they're covering. We think that this happened because of the sport-specific words being used in each of these articles (e.g. "touchdown" is a football specific term and isn't used in other sports). This behavior is consistent with our article separation, which indicates that there are definitive vocabulary differences that are used when talking about specific sports.

Sentiment & Emotion Analysis w/ NRCLex

For sentiment-emotion analysis, a Python library called NRCLex was used. NRCLex is a library designed to process sentiment and emotion data from any valid English text and provide information on the polarity and emotion measures from the text [3]. These measures are calculated and stored in multiple ways.

Sentiment and emotion analysis was done using a **dictionary-based** approach. Dictionary-based methods are much simpler and require significantly fewer resources. As the name suggests, this method relies on a prepopulated dictionary. This is how the NRCLex library collects data from the input text. The effectiveness of such an approach depends on how the dictionary is configured. To appropriately configure the dictionary, you must first define the possible keys and values to be paired. The key set for the NRCLex dictionary is a collection of approximately 27,000 English words, which includes word "synonyms" from the Natural Language Toolkit (NLTK) Python library's WordNet dictionary.

The value set includes the two polarities "positive" and "negative," as well as eight emotions: joy, sadness, trust, disgust, anticipation, surprise, fear, and anger. These keywords were selected to represent the overall “range of emotion” to be found in any given text. They were selected based on psychologist Robert Plutchik’s theory of eight basic emotions. (Figure 1). It must be stated that emotions are variable and subjective, with no clear boundaries of true categorization among them [3].

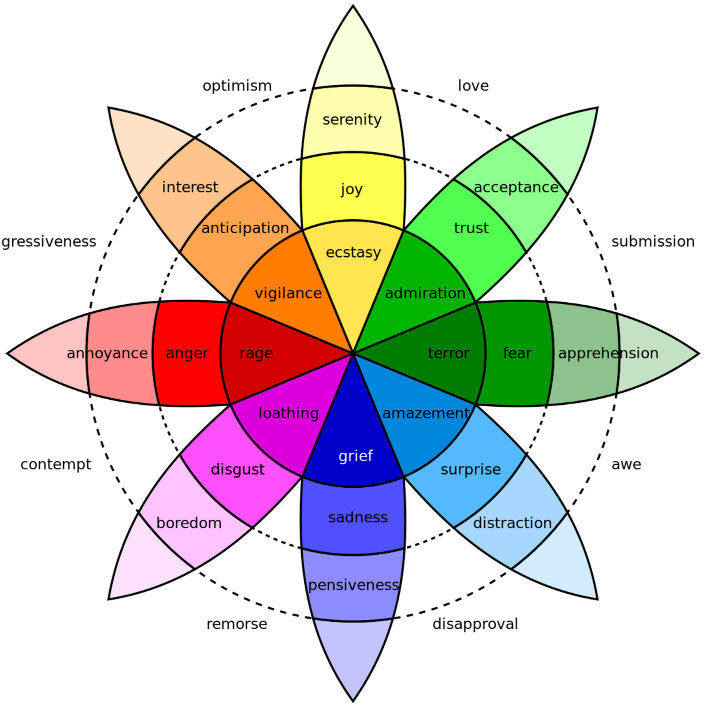


Figure 1. Plutchik’s Wheel of Emotions. The selected emotions are based on the second innermost ring of the wheel.

Within the NRCLex library is a function, `NRCLex()`, which takes a string as input. The function automatically tokenizes each word within the string to then be used as a dictionary lookup key to find the associated emotion(s) and polarity of that word. The function returns an `NRCLex` object as output. The object includes many attributes, but for our purposes we must only understand `raw_emotion_scores`, `affect_frequencies`, and `affect_dict`.

The first attribute to discuss is the `raw_emotion_scores` attribute. `raw_emotion_scores` is a dictionary containing the raw counts, as integers, of every polarity and emotion detected in the input text. Logically, this runs intuitively with the dictionary-based approach we’ve described. If there are no instances of a certain emotion in the text, it will not be included in the `raw_emotion_scores` dictionary. In other words, if the raw count for an emotion is 0, it will be ignored.

Next we’ll discuss the `affect_frequencies` attribute. `affect_frequencies` is a dictionary containing the percentage of each emotion and polarity found in the input text. (“Distribution,” “frequency,” and “percentage” are used interchangeably). The frequency of each emotion is calculated by taking the raw count of that emotion/polarity and dividing it by the raw counts of all emotions and polarity in total. What is interesting about this implementation is the decision to include polarities as a percentage of all of the emotions. It might have been more intuitive to calculate the percentages of the polarities only in relation to each other, and to calculate the emotion percentages separately. Instead, the polarity and emotion frequencies are calculated together despite being ontologically distinct.

Lastly, we have `affect_dict`, which is another dictionary. The key set is the words from the text that have at least one emotion and/or polarity mapped to it. This dictionary will not include words that give no relevant information. (i.e. “the,” “a,” “it,” etc).