Decision Support Systems 144 (2021) 113497

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Would you please like my tweet?! An artificially intelligent, generative probabilistic, and econometric based system design for popularity-driven tweet content generation

Myles D. Garvey [a,*], Jim Samuel [b], Alexander Pelaez [c]

[a] D'Amore Mc-Kim School of Business, Northeastern University. 360 Huntington Ave, Boston, MA 02115, United States of America
[b] Analytics & AI, School of Business, University of Charleston, Charleston, WV 25304, United States of America
[c] Zarb School of Business, Hofstra University, Hempstead, NY 11549, United States of America

ARTICLE INFO

Keywords:
Tweet suggestion
Tweet generation
Marketing analytics
Bayesian statistics
Business analytics
Natural language generation

ABSTRACT

An understudied area in the field of social media research is the design of decision support systems that can aid the manager by way of automated message component generation. Recent advances in this form of artificial intelligence has been suggested to allow content creators and managers to transcend their tasks from creation towards editing, thus overcoming a common problem: the tyranny of the blank screen. In this research, we address this topic by proposing a novel system design that will suggest engagement-driven message features as well as automatically generate critical and fully written unique Tweet message components for the goal of maximizing the probability of relatively high engagement levels. Our multi-methods design relies on the use of econometrics, machine learning, and Bayesian statistics, all of which are widely used in the emerging fields of Business and Marketing Analytics. Our system design is intended to analyze Tweet messages for the purpose of generating the most critical components and structure of Tweets. We propose econometric models to judge the quality of written Tweets by way of engagement-level prediction, as well as a generative probability model for the auto-generation of Tweet messages. Testing of our design demonstrates the need to take into account the contextual, semantic, and syntactic features of messages, while controlling for individual user characteristics, so that generated Tweet components and structure maximizes the potential engagement levels.

1. Introduction

Can an artificially-intelligent system aid social media managers and content creators to overcome the tyranny of the blank page? Not only do content creators often face this problem, but of recent times there has been much discussion as to how managers should leverage social media within their firms. Ranging from branding [1] to detection of product defects [2] and even predicting stock market returns [3], the use of social media in the firm has been proven to not only be an important driver of operational performance, but also that of strategic performance. Put simply, social media engagement has been shown to be a driver of sales [4] as well as an indicator of consumer purchase intention [5]. Thus, it is logical to assume that if the manager can design their content in a manner to increase overall engagement levels of their social media messages, this in turn can lead to increase firm performance.

A consequential question is thus, how should firms interact and navigate throughout social media to further build their engagement levels? When firms leverage social media to interact with their customers, business intelligence becomes valuable so as to better understand the landscape within which they are making decisions. With the recent transcendence of business intelligence (BI) towards business analytics (BA) [6], firms need system designs that will not only report using BI, but also suggest and predict useful message information for the manager to leverage in their strategy using BA. The strategy that a social media manager commonly employs so as to increase their presences and engagement levels rests upon two fundamental categories of decision making. More specifically, the manager or decision maker must first understand with whom shall they connect and subsequently what shall they communicate.

The first part of this strategy involves various network-structure related actions. Firms are not only part of large supply networks [7], but also social networks, and hence they need to undertake strategies

* Corresponding author.
E-mail address: m.garvey@northeastern.edu (M.D. Garvey).

https://doi.org/10.1016/j.dss.2021.113497
Received 26 April 2020; Received in revised form 27 November 2020; Accepted 11 January 2021
Available online 28 January 2021

that will embed the firm within these communities. This is an incredibly important consideration, since network structure and member characteristics influence not only operational considerations of the firm [8], but also the dissemination of information [9–11]. The second dimension of this strategy, which is the focus of our research, involves the articulation of particular content that the firm intends to disseminate. The firm must carefully think about the various sub-dimensions that define this content such as emotions [12], readability [13], information category [14], informative appeals [15], sentiment polarity [16], opinion polarity [17], among many others. Most of these sub-dimensions have been shown to be associated with the levels of engagement, virality, and more generally, popularity of the message [18].

Given that prior research has shown that various aspects of a social media message impact the engagement levels of said message, one must then inquire how should the manager construct such messages? One answer lies in the more recent advances in artificial intelligence, more specifically in the branch of natural language generation (NLG). This subbranch of artificial intelligence is concerned with the development of various methods that seek to generate human-readable language in the scope of highly specific tasks [19,20]. It has recently been argued that natural language generation systems will help reduce"the biggest barrier to getting work done, the tyranny of the blank paper or the blank screen" [21]. Indeed, writers such as social media content designers"will likely shift their activities from drafting to editing" [21]. These systems have already shown promise in aiding decision makers in tasks ranging from generating legal briefs to converting a human-readable description of a computer program into the code of said program [22]. It has even been suggested that 90% of all modern business intelligence systems will have some component that engages in natural language generation [23].

Despite the importance of the implications of implementing and proposing new natural language generation systems, especially to aid various tasks within social media management so as to increase consumer engagement, much of the related extant literature has primarily focused on the behavioral and theoretical attributes of message popularity, engagement, and virality [12,24–27]. Secondarily, this research has also investigated predictive and descriptive analytical methods [28–30], application of textual analytics [31–33], the development of machine learning algorithms [34,35], recommendation system designs [36], and topic modeling [37,38]. To the best of our knowledge, one branch of literature that seems to have lacked attention is within the design of social media language suggestion and generation. A preliminary effort, however, has been made in this limited body of literature, primarily within the scope of hashtag suggestion [39–41].

Our research seeks to address this gap by proposing a system design framework for not only applying traditional natural language processing methodologies to social media messages, but also leveraging the resulting information to aid the decision maker in determining optimal language structure so as to increase the chances of higher engagement levels. First, we move"beyond the hashtag", and propose a design framework that will suggest the important structural language characteristics of the content that the designer should leverage so as to increase the chances of heightened message popularity. In addition, our design framework leverages various empirical popularity models which take into account specification and estimation concerns that the extant literature has overlooked. Our framework leverages the empirical popularity models and other Tweet-related information in order to suggest features, structure, and specific *n*-grams, among other characteristics of"good" Tweets. In order to achieve this objective, both an intricate descriptive analysis procedure and a novel generative probability model and algorithm are proposed in order to auto-generate optimal social media language structure and the various descriptive features that make such messages"attractive". These procedures all operate under the objective of maximizing potential engagement. Thus, our research seeks to address the following questions:

- Which characteristics should be considered in the construction of a popularity-driven Tweet?
- How can popularity-driven Tweets and their features be auto-generated and suggested using data?
- How can a decision support system aid managers in the construction of a popularity-driven Tweet?

2. Related literature

2.1. Concepts in social media analytics

Three constructs that are quite prevalent in the social media related literature are engagement, popularity, and virality. Two of these constructs, namely virality and popularity, are often not well-defined in the extant literature, and tends to be defined within the context of the research [9]. Indeed,"popularity of an online content is not a well-defined, but a highly subjective term" [1]. Engagement on the other hand is often unambiguously"defined as Likes, comments, shares, and click-throughs-with the messages" [42]. Despite the lack of well-defined constructs, our research will not be concerned with this issue as we are not delving into the theoretical properties of these constructs. For the purposes of our research we will consider these constructs to simply be varying levels or derivatives of information sharing on the social media network itself, and heretofore we refer to them as simply "popularity".

Prior research that involve these constructs primarily rests within the behavioral and theoretical drivers and consequences of popularity. For example, some have studied the effects that role distributions and reputations have on the diffusion of knowledge within social networks [43]. Others have studied the effects that network structure had on structural virality from an analytical lens [9]. In addition, the content of posts such as combining brand personality with informative posts was shown to be a driver of engagement levels [42]. These studies are samples of three respective areas in the literature stream, only one of which we consider in this paper. The first stream of literature pertains to the effects that overall network structure has on the popularity of posts. Some have studied the spread of disease awareness within varying degrees of scale-free networks [44], the network propagation effects of popularity of YouTube videos [25], and the association between influential individuals, and the virality of information [45].

The second stream of literature pertains to the user characteristics of those whom are actively engaged within the network. User characteristics of social media accounts such as Twitter include the number of followers, following, lists, pages, or groups to which a user respectively possesses and belongs, posts, and their account age [27]. Much of the popularity research often relies on the use of these variables to partially explain the levels of popularity. For example, it was determined as part of a larger study that follower and following count were associated with structural virtality and overall engagement levels [46]. Likewise, others found that these same variables were associated with reTweet counts [47]. Other studies have also found similar results [12,38]. The third stream of literature, and the one that our research is primarily grounded within, pertains to the drivers of popularity in relation to the content characteristics. Content can include anything from videos [25], memes [48], pictures [49], and most importantly, text [27]. Prior studies have suggested that various elements of textual content have an influence on the popularity of messages. Some had investigated the impact that emotional sentiment has on reTweet behavior [12]. In addition to sentiment, others have studied the impact that topics [37] and journalistic frames [50] have on levels of engagement.

Of course, in order for one to strategically leverage the drivers of engagement within the construction of their own social media messages, they must first analyze social media textual information. A problem that one often faces when analyzing social media data is the ability to do so within a particular context, as well as the ability to identify it in the first place. Context is a difficult concept to define. In fact, most of the literature pertaining to social media either do not offer a precise definition,

or only implicitly do so. For example, context"… can be provided by all kinds of information, and it can portray the circumstances of any entity" [51]. Likewise, others have merely hinted at a definition by providing an example. Indeed, the"word, red, implies a pessimistic opinion about the price movements of $AAPL and $CAT because price decreases are usually quoted in red in the US. However, this information cannot be captured without knowing the context." [52]. Recent attempts have tried to clarify the definition of "context" in the scope of social media research. For example, context has been implicitly defined as a collection of features that describe a post and its replies, such as the number of words or sentences, or the sentiment of the message. Others have opted to rephrase the term "context" as either a "domain" [52] or an "aspect" [35]. Despite not having a clear and unambiguous definition of context within the area of social media research, there has been some research in the area of social media analytics that attempts to analyze or involve a context in some fashion. Such studies have endeavored to understand customer behavior in luxury contexts [53], advertising in communication contexts [54], context-dependent sentiments of words [52], and the design of recommendation systems [51,55].

Even if the analyst were to identify the context of a social media message, the next problem pertains to how one shall go about such an analysis. The field of textual analytics is very large and has made great strides over the prior few years. Textual analytics entails the analysis of text to identify trends. The commonplace tasks within such an analysis include but are not limited to summarizing, classifying, clustering, and predicting [56]. In order to achieve these objectives, a typical framework of preprocessing, text representation, and knowledge discovery is often undertaken [56]. Furthermore, these tasks are usually considered to be either syntactical or semantical. Syntactical tasks comprises extracting a message's"structure" by way of lemmatization and stemming [57], Parts-of-Speech Tagging [31], Grammatical Parsing and Analysis [58], and Character/Word Parsing [59].

Generally speaking, semantics comprises various tasks that are used to understand the deeper meaning behind text. In the technical sense, this mainly entails the grouping of words into categories or topics. In a broader sense, this could range from tasks including understanding the sentiment or emotions behind words to understanding the various meanings behind symbols. In other words, semantical tasks seek to understand the deeper meaning behind text, possibly through the use of Named Entity Recognition [60], Sentiment and Opinion Polarity [12], and Topic Modeling [52]. Such methods have been leveraged to help explain social media message popularity. For example, the effects of text readability [13] and the sentiment of Tweets [38,50] have been demonstrated to be related to engagement. Put generally, language conveyed in social media has been shown to be related to the overall engagement levels of said messages [61]. The behavioral effects of varying information structure and formats have been studied using machine learning and textual analytics in the context of performance in electronic markets [62]. It has also been applied to study the impact of different brand messages on information sharing [63]. Others had leveraged textual analysis to discover that topic, emotional measures and categories, relevance, and mentions were drivers of popularity [12,46,47,64].

2.2. Message suggestion and automated generation in social media

Despite the plethora of literature that aims to understand the drivers and consequences of popularity, both from analytical and empirical perspectives, little attention has been given to the objective of message suggestion frameworks. Despite this, there is some precedent and research that have focused on tangential topics within this area. The literature is currently split into two types of suggestion-based systems: hashtags and Tweets. In the case of hashtag suggestion systems, Latent Dirichlet Allocation (LDA) has been used to take into account text, hashtag, and visual information to recommend hashtags [65]. Similar system designs have been proposed to suggest hashtags based on the interest of users and live streaming topics. It has even been suggested that hashtags can be suggested based on pre-written Tweets that leverage a user's perceptions [41]. On the other hand, Tweet suggestion has experienced less attention in the literature. Frameworks for generating the text of Tweets based on news article headlines and text primarily do so by extracting sentences from said text [66]. Similar approaches have been suggested based on the use of openly available government documents [67].

There are multiple dimensions of automation in the Twitter domain, and most of these automation dimensions are primarily exogenous to the textual content of the Tweet itself. There has been a fair amount of interest in automated Tweet posting, wherein the Tweets are posted using algorithms, but are largely manually developed or created by rules rather than models [68]. There has also been significant interest in automated identification of sentiment and real-time topic extraction due to its relevance to corporate applications and marketing in particular [35,69]. More customized approaches have been used in textual analytics for automating the identification of traits and sentiments using tailored definitions, such as for expression of dominance, which has not been mapped in commonly used sentiment analysis packages [70]. Most analysis methods tend to focus on the content of the message itself and its general readability rather than on the generation of the message itself. However, outside social media, advances in the subbranch of artificial intelligence known as natural language generation has made great strides.

2.3. Natural language generation

In order to understand automated generation of social media messages, we turn our attention to a branch of study known as Natural Language Generation (NLG). In the current rush towards using artificial intelligence for aiding businesses in a variety of tasks, business managers have been turning to natural language processing (NLP) for a variety of organizational analysis, communications processing and textual analytics tasks [37,71]. There has also been a significant emphasis on using NLP for translations, summarization, simplification, paraphrasing and dialogue, among a host of other uses such as textual analytics for sentiment analysis [72,73]. NLP has been used for a wide range of business and organizational goals such as social media analytics and engagement, customer satisfaction, dialogue and feedback, public sentiment driven policy formation, text generation and a myriad of business purposes [74,75]. Though there has been a fair amount of research progress and industry emphasis upon the area of natural language generation (NLG), it is still largely a nascent scientific discipline and much work remains to be done [75]. NLG has been explored under a variety of constraints such as"stylistic /syntactic, their type …hard versus soft, …units to which the constraints are applied …words, sentences, rhetorical structure", and a broad range of artificial intelligence methods and techniques such as artificial neural networks, machine learning, deep learning, classification and clustering, and adversarial rankings, among others [76,77].In spite of many advances, NLP and NLG applications are often unable to understand simple linguistic differences and textual nuances, leading to initiatives mining big data for common sense knowledge [78]. Paraphrasing and textual consequences, discourse mechanisms and grammar rules, focus constraints and task specific extracts, and have been used in traditional approaches to NLG [79–81].

Genetic algorithms have been used to generate poetic text, field-gating encoders and associated description generators have helped describe tables in text, machine learning algorithms have addressed text generation for Q and A structures and customer feedback and recurrent neural networks have successfully helped improve speech synthesis generation, through all of which, the significant emphasis for NLG remains on value creation for businesses and organizations [75,82,83]. In spite of these advances, given the complexities of natural language and the vast range of subjective applications, there are many areas which

remain to be addressed. One such insufficiently addressed area is that of high popularity text generation for social media, and our focus is particularly on viral tweets, their characteristics and potential ways to generate suggestions to help managers create viral tweets, which could have significantly positive impact on customer engagement, and brand visibility.

More generally, the existing models that have addressed natural language generation primarily rest on the process of using partially written text as input and expecting NLG algorithms to render text as output. Put simply, most NLG tasks involves the use of text as input and text as output processes. A gap in this existing area of literature is that of providing other features different from text as input to generate specific types of output text that may possess the provided numerical features. For example, our current paper is focused on the general output of automated text that possess favorable features that could lead to a specific engagement level. In this sense, our approach is distinct from the existing NLG approaches in that we are concerned with tasks that have text and numerical input to generate text output. Put simply, we are using text and numerical features to suggest new text. While our focus in this research is not on text that"makes sense", we are rather focused on text that captures the"essence" of the textual and numerical input.

3. A framework for popularity-driven tweet suggestion and message auto-generation

Our system design seeks to address the gap in the extant literature of designing Tweet message structure, content, and context suggestions, auto-generation, and expected popularity measurement. An outline of the design is shown in Fig. 1. In this section we will detail each module in our design. Our design is influenced in part by the following hypothetical situation in which a social media manager or other decision maker may face. Ideally, a decision maker would like to know how to determine the optimal design-related decisions of their Tweet so as to maximize the expected popularity of their new Tweet. Such decisions involve message context, semantics, and syntax.

For contextual decisions, the manager would like to know the topic or topics that should be reflective in their post. Syntactical Decisions will involve (1) the respective total number of words and characters, (2) the respective relative frequency of verbs, nouns, adjectives, adverbs, determiners, and conjunctions to total word count, (3) the number of hashtags, and (4) the ratio of punctuation to total character count. On the other hand, semantical decisions will primarily involve the overall sentiment of the Tweet itself. Last, the manager would like to know which specific verbs, nouns, adjectives, adverbs, conjunctions, and determiners to use, as well as specific sequences of *n*-grams which are most common among those Tweets that have enjoyed high levels of popularity so as to replicate them, and those of low levels so as to avoid them. Once the manager is provided with this analytical"live snapshot" of message context, content, and structure of popular Tweets, they can design a few potential messages. Once each message is constructed by the manager as a result of the suggestion system, they can then be passed through a final evaluation module in the system to compute an expected popularity level so that the manager can wisely choose which alternative to post.

3.1. Data streaming, searching, downloading, and filtering module

Various social media platforms allow back-end access to messages that are posted and streamed via Application Programming Interfaces (API). In the case of our design, we focus on the Twitter API. The system is intended to download Tweets which are related to trending topics, or, are enjoying recently high reTweet counts. The general process we propose for downloading a set of"trending" Tweets to analyze would be to (1) request a list of streaming topic keywords and store these, (2) search for Tweets using the list of keywords obtained in the previous step, and (3) searching through Twitter with the keywords"RT","reTweet" and filtering the Tweets that are relevant to the trending topics. Once Tweets are stored, the system must then filter them for the relevant information to analyze. In the case of Twitter, the information that the API provides is vast, with over 80 variables returned. Furthermore, we want to filter our data in such a manner that allows for the system to more easily analyze the data. As such, our filtering process for this phase in the module would entail extracting and refining Tweets based on the (1) variables (2) text-salience, (3) English-only, (4) non-reTweets/quotes, (5) source device as"Desktop","Mobile" or"Other", (6) creation date, (7) account age relative to the Tweet creation date, (8) days within the current day, and (9) Tweets from accounts that have positive followers. More specifically for the device, Twitter reports hundreds of different yet specific devices from which the tweet originated, ranging from the type of operating system to the type of device such as a gaming system. We thus recommend categorizing the source of the tweet into it being from a desktop, mobile, or other device so that we can allow for statistical analysis without the need to have hundreds of classifications of source operating systems. It should be emphasized in this section that when downloading data, the design should filter our retweets. This can be accomplished by using the unique

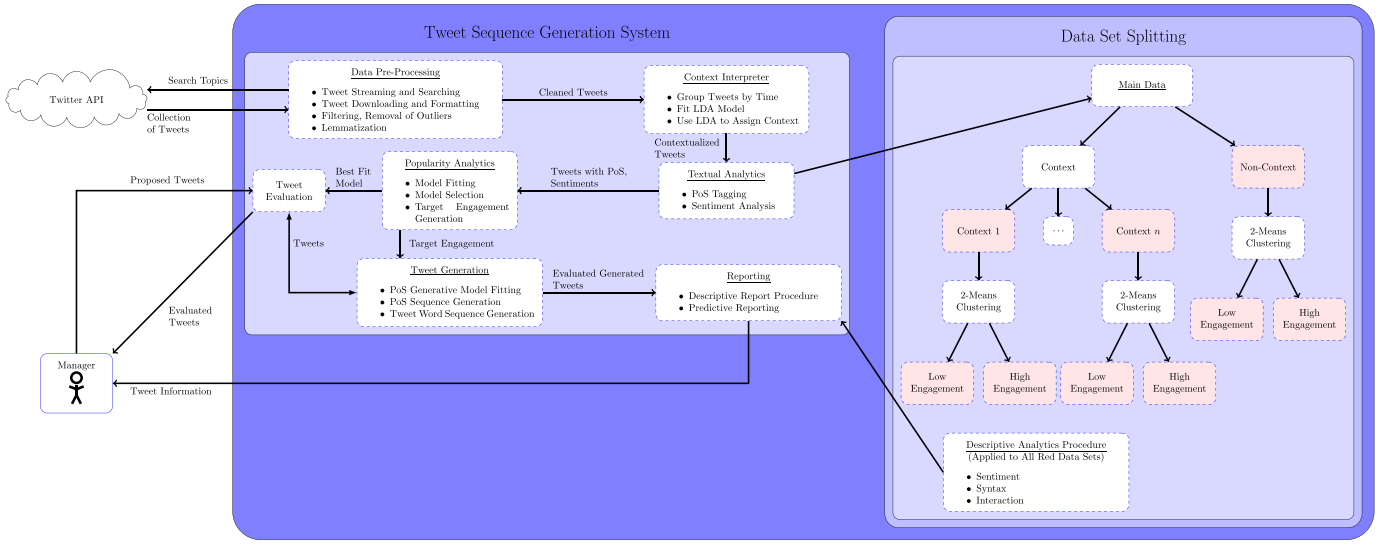

Fig. 1. The popularity-driven tweet suggestion system

identification number on each tweet. Doing so will remove duplicate tweets from the data.

3.2. Context interpreter module

The purpose of this module is to extract, for each downloaded and filtered Tweet, the context of the Tweet's text. Despite context having an ambiguous definition in the literature, it is often characterized by"topic", which can be extracted via the LDA model [84]. Using this model, we can define a"context" as a probability distribution of topics [84]. However, a standard LDA model necessitates large sized documents with a greater quantity of text in each document. Some have shown that the application of LDA to short text messages such as Tweets results in poor predictive performance [85]. Hence, it is often recommended to aggregate Tweets into a"single big Tweet" so that better LDA performance can be achieved [86]. We recommend aggregating Tweets into"single documents" based on the creation date of the Tweet, since trending Tweets tend to follow a well-defined collection of topics. In other words, if we were to group Tweets by user accounts, as others in the literature do, we would risk too much heterogeneity in a single"-document", and it would thus be challenging for the LDA model to identify well defined topics and the words that define said topics. If we group the Tweets by creation date, then there is a good chance that Tweets will share similarity. If Tweets were created around the same time frame, it could be argued that if they are trending then they follow similar patterns of topics. This allows for the LDA model to more clearly identify topics with more related keywords. Therefore, the process of contextualization using LDA would be to (1) aggregate Tweets which fall within the same time frame, (2) estimate the LDA model, and (3) use the LDA model to assign a topic to a new Tweet using the Maximum A Posterior(MAP) [87]. It should be noted that prior to running the LDA model, a pre-processing phase of lemmatization and spell-checking should be conducted so that the LDA model can pick up on the same words, rather than different forms of the words.

3.3. Textual analytics module

Prior to some of our textual analytics tasks, we need to generate a"pre-processed" version of our Tweet data. This process typically involves correction and lemmatization [57,88]. Correction entails spell-checking any words in the message, while lemmatization entails the conversion of words to their"base form" [57]. For example,"dogs" has a base form of"dog","waiting" has"wait", and"joyed" has"joy". This is different from stemming. For example, the word"running" would be stemmed to"runn", which lemmatization will change this to"run". Hence, the general process for the pre-processing phase is to (1) auto-correct each Tweet and to (2) iterate each Tweet through a lemmatizer. This will result in a new data set of"preprocessed" Tweets. We should emphasize that the original text data is kept in addition to this data. As we mentioned in the overview of the framework, the decision maker needs to not only understand the context within which to Tweet, but also the semantic and syntactic characteristics they should implement. We describe these details below.

3.3.1. Syntactical analysis

In this part of this module, two tasks are undertaken, scilicet to find (1) the total number of words and characters, (2) the frequencies of words and characters relative to the total, (3) the ratio of hashtags to total word count, (4) the ratio of various Parts of Speech to total word count (such as noun, verb, etc), and to find most frequent *n*-grams, with *n* chosen by the manager. These tasks often belong to a broader methodology known as Parts-of-Speech (POS) tagging [31].

3.3.2. Semantic analysis

Semantic analysis in Natural Language Processing typically entails relating various concepts together, often times grouping these concepts into larger categories. Generally speaking, semantics concerns itself with"what well-formed expressions are supposed to mean." ([89], pg. 49). Part of this general objective is to understand the relationship between concepts and sentiments [90]. Put simply, while semantic analysis in natural language processing concerns itself with relationships between concepts, general semantics also involves sentiment analysis. In our framework, we use the phrase"semantic analysis" to refer to"sentiment analysis", since this is the primary method through which we choose to study the meaning behind the words. However, we would like to be clear that many in the Natural Language Processing literature use the phrase"semantic analysis" to strictly refer to the study of the relationships between concepts, and our use of the phrase is borrowed from more general linguistics rather than from NLP. We are not attempting to redefine"semantic analysis" in the body of knowledge of NLP, but rather bring the concept of general semantic analysis into our framework.

Sentiment analysis can be carried out in a variety of ways [12,34], primarily through the use of a numerical score or a category assignment [56]. An overall numerical sentiment score is based on the use of a context free or context dependent lexicon [60], which is a collection of words that have been marked with a quantity between −1 and 1 [17]. Sentiments of each word using a chosen lexicon can be combined together to form a sentiment of the entire message. Some common methods for computing the message-level sentiment score involve summation [91], classifiers [92], positive/negative word count [93], and various other adjustments [94]. We suggest that sentiments only be computed on the pre-processed data, since the words within various lexicons are typically in base-form. The output of this module will thus compose of two data sets. The first will consist of the frequency counts of each word and character, each type of word, as well as all of the *n*-grams and frequency of the *n*-grams. The second data set will compose of the Tweet-level data appended with the syntactic and semantic information computed for each text.

3.4. Popularity analytics module

The purpose of this module is to take the non-pre-processed data and the pre-processed data to construct various empirical popularity models. The possible empirical specifications are endless given the amount of information that Twitter provides us. However, we suggest a general specification for system implementers to follow based on what the extant literature has (1) used and (2) the possible shortcomings.

3.4.1. A general empirical specification for tweet popularity

In order for the system to recommend Tweet structure that tends to favor higher probabilities of increased popularity, it will need an empirical model to determine the probability distributions of the popularity levels of a given message. There are many modeling choices since we are mostly leveraging the machine learning philosophy. On the one hand, we can choose our model design by adhering to strict econometric principles of specification and estimation as well as be guided by theory. The disadvantage of doing so, however, leads us towards the problem of conducting more explaining of theory rather than undertaking the task of prediction. In the machine learning context, inner-sample estimation is less of a concern, and often not a concern at all, compared to out of sample prediction. The philosophy is simple: who cares if the model fit is bad, if it predicts well, it's a good model to use. We have many choices ranging from neural networks to random forests. However, while we want to undertake a machine learning philosophy, we want some of our modeling decisions to be partially rooted in econometric theory, although not entirely. Hence, our approach to empirical specification will be primarily to leverage the econometric discipline in spirit, but not as rigid, since our goal here is not to explain but rather to predict. We will not venture off from here, however. While moving to other types of models and estimations may be of interesting topics in future research and others that some may wish to explore, our

scope of specification lies mostly within the econometric spirit while undertaking it from a machine learning philosophy.

First, to measure"popularity", we decided to use the total sum of reTweet and favorite counts since these metrics are easily accessible via the Twitter API and have been leveraged in past studies [12]. A Tweet lends itself to a natural categorization of independent variables to consider. Generally, the set of independent variables that our system will use will comprise the categories of user characteristics and Tweet characteristics, with the latter being our main focus and former being used as control. The extant literature has found support of the hypothesis that user and Tweet characteristics play a role in the level of popularity of messages generated by users [9,12,13,38,47]. Hence, any empirical specification should resemble the following structure:

$$engagement_i = \alpha_0 + \boldsymbol{\beta_1}^T \mathbf{user}\_char_i + \boldsymbol{\beta_2}^T \mathbf{Tweet}\_char_i + \varepsilon_i$$

User characteristics constitute information that the Twitter API returns regarding the user's background information and their relation to the social media network. We specify these characteristics using the vector of parameters $\boldsymbol{\beta_1}$ and the vector of user characteristics $\mathbf{user\_char_i}$:

$$\begin{aligned}\boldsymbol{\beta_1}^T \mathbf{user}\_char_i = &\alpha_1 listed_i + \alpha_2 follower_i + \alpha_3 favorite_i + \alpha_4 following_i \\ &+ \alpha_5 n\_Tweets_i + \alpha_6 verified_i\end{aligned}$$

Given our discussion above regarding the semantic, syntactic, and contextual features of the Tweet, we also would like to include these into the empirical estimation for the Tweet characteristics. However, Tweets have other characteristics that are associated with them that have nothing to do with the message itself. These characteristics are meta-information regarding the Tweet (such as its creation time, etc). We will heretofore refer to such information as the *properties* of the Tweet . Hence, we have the following breakdown of the Tweet characteristic portion in the specification:

$$\boldsymbol{\beta_2}^T \mathbf{Tweet}\_char_i = \boldsymbol{\gamma_1}^T \mathbf{context_i} + \boldsymbol{\gamma_2}^T \mathbf{properties_i} + \boldsymbol{\gamma_3}^T \mathbf{syntax_i} + \boldsymbol{\gamma_4}^T \mathbf{semantics_i}$$

At this point, there are many variable choices that the implementer of this system design can choose to include in the respective categories of the context, properties, syntax, and semantic variables. In our design, we leave this portion general and unspecified, and consider some possibilities in the next subsection.

3.4.2. Empirical model specification considerations

The system will involve the use of a base model, which will include the context, semantic, syntactic, and Tweet properties. However, there are a few estimation and specification concerns that we must take into account that some of the extant literature have not considered. First, the dependent variable is a count variable, which tends to naturally exhibit higher levels of heteroscedasticity [95,96]. Hence, it would be wise to consider a discrete count model such as the Poisson and Negative Binomial (NB) regressions [95]. Such approaches have been used with Twitter data in past studies [47,63,64]. The distribution of the variables, however, is not the only consideration we must keep in mind. Many in the extant literature have also seemed to ignore the possibility of an endogenous relationship between some of the user characteristics (namely the following count) and some of the Tweet characteristics.

Twitter does not return time-dependent count data upon inquiry, and hence, the endogeneity cannot inherently be controlled by controlling time. Put simply, when one queries Twitter for data, it returns the most up to date engagement, follower, and listed counts. The individual querying the API does not have access to a time series of these metrics, and hence, cannot control for the dynamics between engagement, follower, and listed counts. This creates a data estimation problem for those looking to conduct statistical regressions, since they do not know which occurred first in the counts: the followers or the engagement levels of the Tweet. It would be reasonable to presume that higher engagement counts could lead to increased following counts due to increased user account exposure via increased levels of sharing (i.e. reTweeting). Likewise, the Tweet can easily gain higher reTweets as a result of increased exposure due to larger following and hence increased exposure to the Tweet.

In order to overcome this potential problem, the specification can be modeled and estimated using an instrumental variable (IV) approach [96]. One such approach entails a two-stage process whereby we first choose an IV, regress the endogenous variable on the predictors with the instrumental variable, and compute the residuals. This is then followed by a regression of the dependent variable on a model that excludes the IV, but includes the residuals from the first stage as a predictor. One potential candidate for the IV is the listed count variable. This is a tally of the number of other users that have added a user to one of their lists of people to follow. It is a more dedicated, yet laborious, way of following another individual. We posit that the listed count is not an endogenous variable. The reason is as follows. While a user's Tweets will appear on the timeline of other users lists that contain the tweeting user, the opposite increase in list count as a result of a high retweeted item would be less likely than simply just following a new user. Put differently, the listed count changes more slowly than the follower count, and the previous time frames in the error term would have less of an impact on the listed count. This is contrasted with the follower count, where its previous values in time in the error term would be correlated more highly with the current reported value. Taking these into account, the endogenous-based models would resemble the following two-stage specifications. For stage 1, one would regress the followers count on the other variables included in the engagement-based regression, categorized by the user characteristics (without the follower count and the listed count), context, properties, syntax, and semantic variables chosen by the user (we let $\boldsymbol{\phi^T_2 user\_char\_o_i}$ represent the effects that the other user characteristics, excluding follower count and listed count, have on follower count):

$$follower_i = \phi_1 listed\_count_i + \boldsymbol{\phi_2^T user}\_char\_\boldsymbol{o_i} + u_i$$

Once this estimation is conducted, the residuals $\widehat{u}_i$ are computed and are subsequently included in the second specification, which is used to predict the engagement levels:

$$engagement_i = \alpha_0 + \alpha_2 follower_i + \phi_3 \widehat{u}_i + \boldsymbol{\phi^T_2 user}\_char\_\boldsymbol{o_i} + \boldsymbol{\beta_2^T Tweet}\_char_i + \varepsilon_i$$

3.4.3. Module design

Taking into account the considerations of the prior sections, we will now explain the models which we propose to include in a standard Tweet analysis and suggestion system design. These models are specified in Table 1. Models 1–3 will estimate the full model specification using

Table 1
The eight empirical specifications that the system module will run and estimate on the data.

Model number	DV	IVs	Distribution	Estimation
1	Total Engagement	All Tweet and User Char	Normal	OLS
2	Total Engagement	All Tweet and User Char	Poisson	MLE
3	Total Engagement	All Tweet and User Char	Negative Binomial	MLE
4	Total Engagement	All Tweet, User, and Syntactic Interaction Char	Normal	OLS
5	Total Engagement	All Tweet, User, and Semantic Interaction Char	Normal	OLS
6	Total Engagement	All Tweet and User Char	Normal	Instrumental Variable & 2SLS
7	Total Engagement	All Tweet and User Char	Poisson	Instrumental Variable & 2SLS
8	Total Engagement	All Tweet and User Char	Negative Binomial	Instrumental Variable & 2SLS

first OLS, then MLE with the distribution assumption changed to Poisson and Negative Binomial, respectively. Models 4 and 5 will respectively add an interaction term between each context and the semantic and syntactic independent variables. Last, models 6–8 will apply the procedure outlined in the prior section so as to handle the possible endogenous relationship but also take into account the distribution to be leveraged in the MLE.

Estimation will work as follows. The data set will be ordered according to time of Tweet posted. Next, the user will specify a starting value to use to split the data. The system will then conduct estimations on training and testing on a rolling basis. Once the models are estimated, the next Tweet in the data set will be sent to the models for prediction. This will then be compared against the actual total engagement levels, and a percentage error will be computed. The data point will then be appended to the training data set, and the process will repeat again until all data points have been consumed. This will allow the user to obtain predictability metrics. Based on these, the system will pick the"best" model to use based on past predication percentage error and will serve as the primary model to use for suggestion and evaluation.

3.5. Tweet generation module

The purpose of the Tweet generation module is to utilize the descriptive and predictive analysis to generate Tweet Sequences. In order to auto-generate Tweet Sequences, we propose a multi-step process. First, the best prediction model is passed to a sub-module which returns the features with positive coefficients. The sub-module will determine the highest allowable values for each feature in the model, plug these into the model with the negative features set to 0 and thus compute a maximum bound on the total engagement level. This value will be treated as a"target" which will be used as input for a subsequent set of generative probability models. The general process for suggesting Tweet Sequences will then comprise first generating a general sequence of Parts of Speech and then sampling specific words for each specific PoS in the sequence. In order to generate a single message, we first propose a generative model which will sample a set of counts of each PoS. Using the counts, we then propose a process for sampling the specific sequence of PoS by using the sampled counts. After the sequence is generated, we then replace each PoS in the sequence with a specific word by sampling from a distribution of words conditional on the PoS to replace and the previous word sampled in the sequence. The details of these models follow in each subsection below.

3.5.1. Parts of speech generative count model

In order for the system to optimally select different syntax structures which result in the estimated total engagement count, we propose that the system leverage a generative probability model to understand the actual counts of different types of words based on the targeted total engagement. For our model, we consider seven categories for parts of speech, namely nouns, adjectives, verbs, conjunctions, determiners, adverbs, and punctuation. The generative model will first assume that a total count of words n as well as the total expected engagement level y is known. Based on this, the generative model can be leveraged to sample counts of each parts of speech which sum to n but whose samples are indirectly dependent on the target total engagement level y. Each PoS category $j \in \{1, \ldots, 7\}$ present in the Tweet has a count which is modeled as a random variable X_j. A single Tweet i can therefore be characterized syntactically by PoS counts $(X_1, \ldots, X_7)_i$, where $\sum X_j = n_i$ for Tweet i. This is a random vector which is assumed to be distributed as *Multinomial* $((p_1, \ldots, p_7)_i, n_i)$. We further assume that $(p_1, \ldots, p_7)_i \sim Dir((\alpha_1, \ldots, \alpha_7)_i)$, a Dirichlet Distribution, where $\alpha_{j,\,i}$ is dependent on y_i. More specifically, we assume that $\alpha_j = e^{\beta_{0,\,j} + \beta_{1,\,j} y_i}$. Last, we assume that the parameters which define each $\alpha_{j,\,i}$ are themselves random, and that $\beta_{0,\,j} \sim N(\mu_{0,\,j}, \sigma_{0,\,j})$ and $\beta_{1,\,j} \sim N(\mu_{1,\,j}, \sigma_{1,\,j})$. Summarizing our proposed generative model, we have:

$$\beta_{0,j} \sim \mathrm{N}\left(\mu_{0,j}, \sigma_{0,j}\right)$$
$$\beta_{1,j} \sim \mathrm{N}\left(\mu_{1,j}, \sigma_{1,j}\right)$$
$$(p_1, \ldots, p_7)_i \sim \mathrm{Dirichlet}\left(e^{\beta_{0,1} + \beta_{1,1} y_i}, \ldots, e^{\beta_{0,7} + \beta_{1,7} y_i}\right)$$
$$(X_1, \ldots, X_7)_i \sim \mathrm{Multinomial}\left((p_1, \ldots, p_7)_i, n_i\right)$$

This model can be further simplified to allow for more tractable parameter estimation. First, we can create an 8th PoS category which we indicate as"empty". If we set a single number $n = \max\{n_i\}$, then we can set $(X_8)_i = n - n_i$. By doing this, it can be assumed that each multinomial observation will sum to the same number of counts. Next, we can collapse the two bottom layers of our model into a single distribution, which is known as the Dirichlet-Multinomial Distribution [87], whose pdf is defined as $f(X_1, \ldots, X_8) = \frac{\Gamma(n)\Gamma\sum \alpha_k}{\Gamma\left(n + \sum \alpha_k\right)} \prod_{i=1}^{8} \frac{\Gamma(X_k + \alpha_k)}{\Gamma(X_k)\Gamma(\alpha_k)}$, where $\Gamma(x)$ is the Gamma function, and we can write $(X_1, \ldots, X_8)_i \sim DirMulti((\alpha_1, \ldots, \alpha_8)_i, n)$. With this model, we will be able to generate samples of count vectors where a single sample would represent the counts of each PoS in our generated Tweet. The collapsed version of our model would thus be:

$$\beta_{0,j} \sim \mathrm{N}\left(\mu_{0,j}, \sigma_{0,j}\right)$$
$$\beta_{1,j} \sim \mathrm{N}\left(\mu_{1,j}, \sigma_{1,j}\right)$$
$$(X_1, \ldots, X_8)_i \sim \mathrm{DirMulti}\left(e^{\beta_{0,1} + \beta_{1,1} y_i}, \ldots, e^{\beta_{0,8} + \beta_{1,8} y_i}, n\right)$$

3.5.2. Generative model estimation

In order to sample syntax structures to recommend using the number of words and the total engagement count, one needs to fit the generative probability model. The joint probability distribution for the generative model described above is (we use"DM" to represent"DirichletMultinomial"):

$$f\left(\beta_{0,1}, \ldots, \beta_{0,8}, \beta_{1,1}, \ldots, \beta_{1,8} | X, y\right) \propto f\left(\boldsymbol{X} | \beta_{0,1}, \ldots, \beta_{0,8}, \beta_{1,1}, \ldots, \beta_{1,8}, y\right) f\left(\beta_{0,1}, \ldots, \beta_{0,8}, \beta_{1,1}, \ldots, \beta_{1,8}\right) \propto \prod_{i=1}^{N} f\left(\boldsymbol{x_i} | \beta_{0,1}, \ldots, \beta_{0,8}, \beta_{1,1}, \ldots, \beta_{1,8}, y_i\right) \prod_{m=1}^{8} f\left(\beta_{0,m}\right) f\left(\beta_{1,m}\right)$$
$$= \prod_{i=1}^{N} DM\left(\boldsymbol{x_i} | e^{\beta_{0,1} + \beta_{1,1} y_i}, \ldots, e^{\beta_{0,8} + \beta_{1,8} y_i}\right) \prod_{m=1}^{8} N\left(\beta_{0,m} | \mu_{0,m}, \sigma_{0,m}\right) N\left(\beta_{1,m} | \mu_{1,m}, \sigma_{1,m}\right)$$

The parameter distribution can be estimated using a Random Walk Metropolis-Hastings (RW-MH) Algorithm [87]. Let $x_t = (\beta_{0,\,1}, \beta_{1,\,1}, \ldots, \beta_{0,\,8}, \beta_{1,\,8})_t$. Generally, the algorithm operates by choosing a *proposal distribution* $q(x_t | x_{t-1})$ to sample new points x_t conditional on the current point x_{t-1} in the parameter space and then computes an acceptance probability to determine if the newly sampled point will be"accepted" or"rejected". Usually, once the ratio is computed, another number is randomly selected from a uniform distribution on 0 and 1. The new sample is then accepted if the sampled number is less than the ratio, otherwise the new sample is set equal to the previous sample. Typically this probability is computed by $\alpha = min\left\{1, \frac{f(x_t) q(x_t | x_{t-1})}{f(x_{t-1}) q(x_{t-1} | x_t)}\right\}$. When the proposal distribution is symmetric, we have $q(x_t | x_{t-1}) = q(x_{t-1} | x_t)$, which simplifies our computations. After enough iterations, the samples

from the proposal distribution should collectively converge to the posterior distribution of the parameters, for which either the MAP or the expected value of the distribution can be used as an estimate for the parameters [87]. We chose to use the expected value of the sampled posterior distribution. Specific to the RW-MH, q is often chosen to be a normal distribution, and each proposal is computed as $x_t = x_{t-1} + N(0, \sigma)$. For our research, we chose to use a non-correlated Multivariate Normal distribution with all means set to 0 and with the standard deviations set equal to those for each respective beta parameter found during the initialization step (more on this below). Doing so led to a total acceptance rate of 48%, which is within recommended calibration for the MH algorithm [97]. Given the complexity of our model, we reworked this algorithm in logs. That is, we have:

$$log(f(x_t)) = \sum_{i=1}^{N} log(DM(\boldsymbol{x}_i | e^{\beta_{0,1}+\beta_{1,1}y_i}, \ldots, e^{\beta_{0,8}+\beta_{1,8}y_i})) + \sum_{m=1}^{8} log\left(N\left(\beta_{0,m} | \mu_{0,m}, \sigma_{0,m}\right)\right) + log\left(N\left(\beta_{1,m} | \mu_{1,m}, \sigma_{1,m}\right)\right)$$

We then were able to easily compute the ratio:

$$log(\alpha) = min\{0, log(f(x_t)) - log(f(x_{t-1}))\}$$

Once we have the ratio, we then can sample $r \sim Unif(0,1)$, and if $log(r) < log(\alpha)$ we accept the newly sampled x_t from the proposal distribution, otherwise we set $x_t = x_{t-1}$. To initialize our parameters, we recommend to first discretize the log of the total engagement of each Tweet into equal sized intervals. For each interval, once can group the Tweets together and fit a Dirichlet Multinomial model to obtain the parameter estimate of $(\alpha_1, \ldots, \alpha_8)$, which can be assigned to each Tweet i in the interval. Next, for a certain number of iterations, one can randomly sampled a smaller number of Tweets and estimate the model $log(\alpha_{j,i}) = \beta_{0,j} + \beta_{1,j} y_i$ using OLS for each iteration. Doing so allows one to generate a data set which could be used to initialize each $\beta_{m,j}$. Once can subsequently compute the average $\mu_{m,j}$ and standard deviation $\sigma_{m,j}$ for each parameter and initialized the parameters by setting $\beta_{m,j} = \mu_{m,j}$.

In summary, our estimation approach works as follows:

- Initialize Parameters
  - Compute the log of the total engagement variable for all tweets.
  - Discretize each engagement and assign it to a an interval in one of equal sized intervals.
  - For each group, fit a Dirichlet Multinomial Model to obtain the $(\alpha_1, \ldots, \alpha_8)$ estimates. Assign to each Tweet in the group these values (hence, 8 new variables, the alphas, appended to the data).
  - For a certain number of iterations:
    - Sample a smaller number of tweets. Estimate the model $log(\alpha_{j,i}) = \beta_{0,j} + \beta_{1,j} y_i$ using OLS. A single iteration creates a single estimated value for each beta coefficient.
- For each beta parameter, find the average. This initializes the vector $x_0 = (\beta_{0,1}, \beta_{1,1}, \ldots, \beta_{0,8}, \beta_{1,8})$
- For n iterations, storing each x_t in a vector:
  - Sample $r \sim Unif(0,1)$. Sample $z \sim N((0, \ldots, 0), (\sigma_{0,1}, \sigma_{1,1}, \ldots, \sigma_{0,8}, \sigma_{1,8})$. Compute $x_t = x_{t-1} + z$. Compute $f(x_t)$ and $f(x_{t-1})$. Compute $log(\alpha) = min\{0, log(f(x_t)) - log(f(x_{t-1}))\}$. If $log(r) \geq log(\alpha)$ then set $x_t = x_{t-1}$. Else, keep new assignment.
- Compute average of betas sampled.

3.5.3. Parts of speech sequence generation

The system can leverage the generative model in the previous section to auto-generate a count-vector $\boldsymbol{X_1} = (X_1, \ldots, X_8)_1$, which will need to be converted to a specific sequence $(Z_1, \ldots, Z_n)$ of PoS, where $Z_k \in S = \{Noun, Adj, Verb, Con, Det, Adv, Punc, Empt\}$. In order to do so, we can think of the generation of the specific sequence as a finite-space stochastic process, where the state space is S and the transition probabilities update after each iteration. First, we need a set of base transition probabilities to know how to move from Z_t to Z_{t+1}. However, we also need to take into account the $\boldsymbol{X_t}$, since once we move into a state, we need to update $\boldsymbol{X_t}$, and subsequently update the transition probabilities. The reason for doing so is that it may be impossible to sample given our counts, despite our base probabilities being positive. For example, we may have $P(Z_t = \text{Verb} | Z_{t-1} = \text{Verb}) > 0$, but we may have $\boldsymbol{X_{t-1}} = (4,1,1,3,4,2,0,4)$ and $\boldsymbol{X_t} = (4,1,0,3,4,2,0,4)$, where the number of verbs left to sample is 0. Hence, in this case we need to ensure that $P(Z_t = \text{Verb} | Z_{t-1} = \text{Verb}, \boldsymbol{X_t}) = 0$. The way we define the transition probabilities of the stochastic process is as such. We start with estimates of $r_k = P(Z_1 = k)$ and $r_k^{(t)} = P(Z_t = k | Z_{t-1})$ for each PoS $k \in S$ based on existing Twitter data. Next, we update these probabilities by taking into account $\boldsymbol{X_1}$. To do so, we essentially want to keep the other probabilities proportionally the same, but set the probability to 0 for PoS which have count 0. Hence, we first set $t_k^{(1)} = \min\{r_k, (X_k)_1\}$ for each PoS $k \in \{1, \ldots, 8\}$. If the count is 0, then the $t_k^{(1)}$ will be set to 0. Next, we update the probabilities of the distribution by setting $p_k^{(1)} = \frac{t_k^{(1)}}{\sum_{m=1}^{8} t_m^{(1)}}$. After sampling $Z_1 \sim Cat(p_1^{(1)}, \ldots, p_8^{(1)})$, we create a vector $\boldsymbol{X_2}$ by updating the count $(X_{Z1})_2 = (X_{Z1})_1 - 1$. Next, we repeat the process above, but this time leverage the base conditional probabilities $r_k^{(t)}$. We continue the process while $\sum_{k=1}^{8} (X_k)_t > 0$. For sake of parsimony, we can ignore the"empty" category.

3.5.4. Tweet message generation

Once a sequence $(Z_1, \ldots, Z_n)$ has been generated, the system must then replace each PoS with a word to create a sequence of words $(W_1, \ldots, W_n)$, and hence the final Tweet message. However, in an attempt to make the fully written sentences make more sense, we seek to sample words for each PoS that we can expect to appear next to other words. On the other hand, if we base the conditional probabilities of each word given that a previous word in the sequence has been chosen, we may be essentially overfitting our data too much, and our process will not allow for the generation of new messages unless they have appeared in old messages. To balance between these two interests, we propose that each word be sampled from a mixture distribution, where part of the distribution reflects prior bigrams of words in Tweets, and the other part reflects the diversity of words to choose for a given PoS.

To conduct the sampling from the mixture distribution, we first need a base distribution from an existing Twitter data set. We assume that $d_{w_k} = P(W_t = w_k)$ and that $d_{w_k}^{(v_j)} = P(W_t = w_k | W_{t-1} = v_j)$. Put simply, we are given the probability of a given word in a given PoS, and we are also given the probability of a word given the prior word in the sequence and the current PoS. We further assume that this dictionary is built in a way to set $P(W_t = W_{t-1} | W_{t_1}) = 0$. That is, we assume it is impossible for the same word to be repeated consecutively. These probabilities can be estimated from a given set of words that have been extracted from Tweets. The probabilities are then mixed to form a mixture distribution. Therefore, we can create a sampling procedure to create the sequence of words (and hence, Tweet Sequences). To create the sequence, we start by sampling a word for W_1. For the PoS in $\boldsymbol{Z}$ at location 1, we sample from $W_1 \sim Cat(d_{w_k})$, where the categories are the words in the collection of PoS k. Next, for each Z_t that is not empty, we sequentially sample W_t from $W_t \sim Cat(0.5(d_{w_k} + d_{w_k}^{W_{t-1}}))$. Once this process is complete, we have a valuable and artificially intelligent Tweet Sequence $(W_1, \ldots, W_n)$. The goal from here is to generate many of these Tweets based on the target total engagement.

3.6. Reporting module

Once the models from the Tweet Generation Module are estimated, the system will then conduct a full descriptive analysis on all of the data gathered as well as analyzed thus far. It will report all of this information along with others in an organized format to the user for them to be able to construct Tweets with desirable features. Upon reporting, we suggest that the system organize the information into two major categories:

descriptive and predictive information. It will first apply the a descriptive analysis procedure, shown in more detail below, to different versions of the data set. For the predictive reporting, the system will report on the generated tweets from the generative model, as well as report on the factors that act as"Dampers" and"Enablers" of engagement. After it is complete with its descriptive analysis, it will organize the information from the predictive models and descriptive analysis and report them to the user.

For descriptive information, the system will execute a descriptive analysis procedure relating to the context, syntax, and semantics of Tweets. We suggest that the system apply the same descriptive analysis on two types of data sets. First, it should apply the descriptive analysis procedure to the entire set of Tweets it downloaded and prepared for analysis. Next, the module will run a clustering algorithm on the entire set of Tweets using the engagement variable to categorize Tweets as being"highly engaged" and"lowly engaged". Each of these respective data sets will also have the descriptive analysis procedure applied to it. Next, the system will split the main data into sub-data sets based on the context. Within each context, it will re-run the clustering algorithm to split the data again into two data sets for each context. On each of these split data sets for each context, it will again apply the descriptive analysis procedure. As for the procedure itself, we propose that the system descriptively analyze the syntactic, semantic (i.e. sentiment), and the interaction of these features of the Tweets in each set of data described above. The type of descriptive analysis it will undertake will be split into three general categories: syntax, semantics, and interaction of the two.

For the semantic analysis, we propose that the system report on the most frequent words used in the data, the most positive sentiment and negative sentiment words used, as well as how often the most positive and negative sentiment words are used by finding the product between the ratio and sentiment. The system will also report the most positive and negative Tweets and conduct an *n*-gram analysis on the words in the Tweets. For syntax, the system will find a few types of information pertaining to the Parts of Speech. First, it will convert each Tweet to a collection of ratios for each Parts of Speech the system considered in the previous modules. For each PoS, the system will compute the summary statistics of these ratios. Next, the system will conduct an *n*-gram analysis on the PoS sequences. For each Tweet, the message is converted to its PoS sequence, and the same procedure described above to conduct an *n*-gram analysis will be applied to the PoS sequences. When this is complete, the system will report on the specific PoS *n*-gram as well as the frequency of that *n*-gram in the data. Last, the system will combine the PoS and sentiment analysis to understand the common words often employed within each PoS, the word sentiments associated with each PoS, the various Tweet sentiments associated with each PoS, and the Tweet sentiments of the various PoS *n*-grams.

3.7. Tweet evaluation module

The very last module of the system will act as an evaluation module for the decision maker. After they receive the suggestions and descriptive information in the previous module, the decision maker is responsible for crafting multiple versions of a Tweet. After each Tweet is constructed, the collection of Tweets will be inputted into the evaluation module. This module will use the best empirical model to evaluate the crafted Tweet, and will return to the manager a probability distribution of the engagement levels, as well as other descriptive statistics, for each Tweet. The user can then use these metrics to determine which Tweet is the best one to publish with respect to expected popularity.

4. Demonstration of system design

While the purpose of our reserach is to present a general process of analyzing trending tweets as well as generating engagement-favorable tweets, we wanted to demonstrate the capability of this design by implementing it in an ad-hoc manner. Many can choose to implement our design as they see fit, which is why we left some of the modules in a more general description. As for our demonstration, we implemented some of the various modules of our system in an ad-hoc manner using R Studio Server running on two different Amazon EC2 Instances with one comprising of 8 CPUs and 32GiB of memory and the other comprising of 32 CPUs and 128 GiB of memory. All code was written in R and leveraged multiple packages. The goal of our evaluation was primarily to study the system's ability to report the desired information as well as to test the overall predictive ability of the other modules in our system design. Our evaluation entailed of a few tasks. First, we designed out the data gathering and filtering module to download a data set against which the system's performance could be evaluated. In addition, we developed the context interpreter and textual analytics modules, respectively, to be able to compute the relevant information which were needed as input into the popularity analytics and suggestion modules. Next, we implemented the 8 specified models outlined above. Last, we demonstrated the use of the reporting and suggestion modules as well as conducted numerical experiments to test the predictive performance of the system. Each of these tasks are explained in detail in the following subsections.

4.1. Data gathering, filtering, contextualization, and textual analytics

Custom R code was written leveraging the `rTweet` package to download a dataset according to the design. We decided to download a single yet extended run of data over the course of a month (Oct - Nov 2018) so that all of our analysis can be completed in one sitting. After collection, we filtered through the ids to extract only the unique and original Tweets. Last, we ran a final script to download the full 80+ variables for all the status ids. When the download was complete, we were left with 63,742 unique Tweets generated in this time frame. Our next step was to pass the gathered data through the filtering and cleaning module as described in the system design. Filtering led to us having 25,000 distinct Tweets. Additional pre-processing of the tweets involved spell checking leveraging the `hunspell` library in R and lemmatizing leveraging the `hash_lemmas` lexicon as specified by [98]. The tweets were then passed to the contextualization module where Latent Dirichlet Allocation was conducted. Our analysis led us to conclude that there were 6 major contexts in our data. Since a majority of our data came during the 2018 midterm elections, a good amount of the contexts were specific to the election and politics. Context 1 was pertaining to the general presidency (keywords included"military","president","american"), context 2 pertaining to the election ("vote","follow","election"), context 3 pertaining to political parties ("republican","democrat","fight"), context 4 pertaining to Trump presidency ("trump","florida","family"), context 5 pertaining to Obama presidency ("obama","democratic","health"), and context 6 pertaining the Mueller investigation ("mueller","congress","law"). The tweets were then sent to the textual analytics module where it underwent the syntactic and semantic computations, parts of speech analysis with Nouns and pronouns were both counted as nouns, numericals and adjectives were both counted as adjectives, verbs and auxiliaries were both counted as verbs, subordinating and coordinating conjunctions were counted as conjunctions, while determiners, adverbs, and punctuation were counted as their own categories. We next conducted the semantic analysis on our Tweet data by computing sentiments using the Sentiword Lexicon, which is a commonly used system.

4.2. Numerical experiments for popularity prediction

In order to test the prediction performance of our proposed models, we leveraged R's basic lm and glm functions to estimate the linear and Poisson models. The MASS package was used to leverage the glm.nb function to estimate the negative binomial model. The models which handled the endogenous variable were coded in two phases according to

[99]. Each model was coded into its own R function for ease of calling on different portions of our data set. In order to test the predictive ability of our model, we ran a numerical experiment by estimating the model for various time frames over our data. Before doing so, we ensured that our data was ordered in ascending order based on time. First, we sequentially estimated each model with data starting at the first Tweet generated in our data set and ending at the index of iteration. We started the index at 200 (to allow for a sufficient initial sample size for estimation) and iterated it up to the Tweet before the last Tweet in the data set. In each iteration, the model was estimated on the training data and subsequently used to predict the total engagement level of the next data point. For example, we first estimated our model on data points 1 to 200 and used the models to predict the total engagement for data point 201. We then estimated the model again with training data points 1 to 201 and used the models to predict the total engagement for data point 202. We continued in this fashion until the end of our dataset. The result was a set of predictions where the training data comprised of a cumulative set of Tweets.

4.3. Tweet generation module demonstration

First, we ran fit the generative probability model by following the estimation procedure outlined earlier. We ran the simulation for 50,000 iterations using custom written code in R. As for the reporting module demonstration, we used the most recent Tweets in our data set that comprised of one entire week prior to the most recent Tweet in our dataset, which comprised of 3341 Tweets. We removed the data points which had more than 25 words, which was approximately the mean number of words in this smaller data. This reduced our data down to a size of 2018 Tweets. Upon this data we ran the 8 model estimations using a cumulative rolling window beginning with the first 80% of data (1614 Tweets). Each data window was trained with the 8 models, predicted the next data point's engagement levels, computed the percentage error, and subsequently appended the next data point to the training set. This process continued until the 2017th data point was used in the training. We subsequently selected the model which was the highest percentage of times that it had the smallest percentage absolute error, which was model 7. We estimated a final version of model 7 with all 2018 data points and selected the context that led to the highest total engagement according to the coefficients, which was context 4. Next, we fit our generative model with the 2018 data points. Last, we used model 7 to set a target total engagement level and used the generative model to generate sequences of general PoS *n*-grams and specific Tweet messages. Furthermore, we used the the model 7 fit to judge the quality of the generated Tweets.

After the parameters were estimated, they were used in our generative model to generate PoS count vectors. We subsequently sampled 1000 random count vectors from the model. For each count vector, we carried out the algorithm mentioned in 3.5.2 with custom written code in R which used each count vector to sample PoS sequences. For each count vector, we sampled a PoS sequence using the algorithm and subsequently converted it to a Tweet message using the mixed-distribution sampling process explained above. Each Tweet was sent to Model 7 with context 4. The user account information was assumed to be set equal to the average of each user characteristics in our context 4 data. When we were finished, we were left with a total of 1000 fully written Tweet suggestions, along with PoS sequences, count vectors, and predicted engagement levels.

4.4. Reporting module demonstration

We next wrote custom code in R to implement the reporting module. We first wrote a custom function that would carry out the procedure outlined above for conducting a descriptive analysis when given a data set. We next took the data that was generated for the prior week's set of Tweets and broke this into the datasets which appear in red in Fig. 1. Given the limited space in our manuscript, we decided to run the descriptive analysis procedure on only (1) the entire data set, (2) the context 4 data set (since this was the context of focus from the prior section), and the (3) high/(4)low engagement data sets derived from context 4. To obtain the high/low engagement data sets for context 4, we ran a *k*-means clustering algorithm to split the data into high/low engagement Tweets. The algorithm was ran strictly on the engagement variable and no other variable. Doing so allowed the data to"speak for itself' so that the natural boundaries of"high" and"low" are clear. Essentially, the clustering algorithm allowed for us to avoid the problem of choosing arbitrary cut-off thresholds with fixed-size intervals, and allowed for the data to essentially tell us based on one-dimensional distance where the threshold should be based on how the engagement level data was clustering into two groups. To measure the quality of the resulting clusters, we computed the silhouette index on cluster sizes ranging from 2 to 10. The results were, respectively for cluster sizes 2 to 10, 0.8483, 0.8009, 0.7380, 0.6620, 0.6324, 0.5776, 0.5521, 0.5544,0.5543.

The output from the descriptive procedure only for the entire dataset is shown in Table 2. In each of these respective data sets, we applied the descriptive procedure. We gathered all of our results into a single output table for organization and reported the top 5 in each category of description indicated in Table 2. As for the predictive reporting, we took the model estimates for context 4 and separated the Tweet features based on the sign of their respective coefficients. We then ordered them based on the level of importance, which is indicated by the coefficient. We also reported the beta parameters that the generative model fit and organized these under each PoS. We last reported the highest-predicted engagement levels for the auto-generated Tweets, as well as computed their text sentiments, which are shown in Table 3.

5. Results of evaluation

5.1. Prediction performance of the Total engagement models

After receiving the predictions from the 8 models across multiple data windows, we computed the percentage error of each Tweet. For each time frame and test Tweet, we had 8 prediction errors. We can see that models 1, 4, 5, and 6 led to the most stable predictions with respect to the variance and the lowest percentage error. In addition to this analysis, we computed the absolute percentage error of the predictions. In this data and for each Tweet predicted, we determined which model led to the most accurate guess, which was defined as the model that led to the lowest absolute percentage error for the Tweet. Fig. 2 indicates the percentages of the"best models" based on the model characteristics. Looking at the distribution of best guesses based on the distribution used, we can see that the normal distribution was dominant. In addition, we can see that model 4 performed the best in most instances, followed by models 6,5,1,3,8,2, and 7, respectively in that order from best to worst. We can see the non-endogenous models outperformed the endogenous models. The models that did not control for an interaction between context and content were far superior than the models that include these interaction terms.

In addition to this analysis, we also wanted to determine how well the system performed over a long period of time. One way of doing so is by computing the percentage errors relative to the actual tweet's engagement level. Each model can be used to form a prediction at a particular point in time (since the tweets are ordered by time). The predictions for each time frame can then be cumulatively averaged so that we can get a picture of how much the percentage errors are stabilizing. We can accomplish this performance by observing the cumulative moving average of the absolute percentage errors. Essentially, for each point in time, each model was estimated on a portion of the chronologically ordered data and the next data point was inserted into the model to obtain a prediction for the engagement. This allowed us to have for each time frame an actual engagement level (a_i) of the tweet i and a

Table 2
The output of the descriptive procedure ran on the entire data set. The results show the top 5 for each category.

Semantics															
Word frequency		Word sentiments				Word impacts				Tweet sentiments				N-Gram Frequency	
Word	Freq	Positive		Negative		Highest		Lowest		Positive		Negative			
		Word	Sent	Word	Sent	Word	Impact	Word	Impact	Tweet	Sentiment	Tweet	Sentiment	N-Gram	Count
trump	95	excellent	1	afflict	−0.875	all	68	have	−90.5	we make each other happy, we keeping around whatever make us happy & why it make us happy is our business	0.810133918	Minor Threat to Senior Threat.	−0.670820393	in the	78
win	85	character	0.875	bulls***	−0.875	time	38.75	by	−25	energy	0.625	Bravery in indie film is rewarded.	−0.510310363	for the	67
thanksgiving	82	greatest	0.875	deplorable	−0.875	good	37.03125	off	−16.95833333	JUICE REALLY DO TASTE BETTER WHEN YOU DRINK IT WITH THE REFRIGERATOR STILL OPEN	0.61835979	Bruh	−0.5	of the	65
don	81	beautiful	0.75	guilt	−0.875	happy	36.875	keep	−16.875	lots of masked character design sketches âœ	0.612372436	Imagine not wanting Kirk Cousins on your team but wanting to have Alex Smith.	−0.492762915	this is	54
rt	69	bonnie	0.75	head	−0.875	like	31.5	s***	−13	14,056 immigrant children are in U. S. custody, an all-time high.	0.610560473	Our president has more harsh words for Michelle Wolf than MBS.	−0.490317749	to win	51
Syntax															
PoS Ratio Summary Statistics										Most Frequent PoS N-Grams					
	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	SD	VAR		N-Gram	Count				
n_punct	0	0	0.136363636	0.156199732	0.227272727	1	0.147988863	0.021900703		[NOUN][NOUN]	2654				
n_verb	0	0.111111111	0.181818182	0.189575473	0.260869565	1	0.122983336	0.015124901		[NOUN][VERB]	2169				
n_noun	0	0.294117647	0.380952381	0.395042645	0.5	1	0.164194991	0.026959995		[NOUN][PUNCT]	1688				
n_adv	0	0	0	0.052708235	0.090909091	0.666666667	0.07638132	0.005834106		[VERB][NOUN]	1612				
n_adj	0	0	0.083333333	0.095978165	0.142857143	1	0.106948868	0.01143806		[ADJ][NOUN]	1137				
n_con	0	0	0	0.030459861	0.058823529	0.333333333	0.049634787	0.002463612							
n_det	0	0	0.071428571	0.080035888	0.133333333	0.5	0.083340476	0.006945635							
Interaction															
Descriptive	NOUN		VERB		ADJ		ADV		CON		DET		PUNCT		
	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	
Frequency	you	0.041531121	is	0.079839051	one	0.027681661	when	0.081538462	and	0.440599769	the	0.44666002	.	0.410467588	
	i	0.035241813	be	0.02901313	happy	0.023356401	just	0.071538462	that	0.175317186	a	0.261714855	,	0.137353879	
	me	0.022446324	are	0.018636171	good	0.01816609	so	0.061538462	if	0.106113033	this	0.107676969	!	0.119022317	
	my	0.018651052	have	0.018000847	best	0.014705882	how	0.048461538	but	0.103806228	all	0.050847458	:	0.089266738	
	it	0.01810887	do	0.017789072	exclusive	0.014705882	now	0.046923077	or	0.044982699	no	0.02891326	”	0.077045696	
	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	
	bulls***	−0.875	ashamed	−0.75	small	−0.875	poorly	−0.6875	as	−0.125	a	0	off**	−0.458333333	

(*continued on next page*)

Table 2 (*continued*)

Semantics															
Word frequency		Word sentiments				Word impacts				Tweet sentiments				N-Gram Frequency	
Word	Freq	Positive		Negative		Highest		Lowest		Positive		Negative			
		Word	Sent	Word	Sent	Word	Impact	Word	Impact	Tweet	Sentiment	Tweet	Sentiment	N-Gram	Count
Most	guilt	−0.875	hurting	−0.75	worried	−0.875	cowardly	−0.5	#endgunviolencetogether	0	an	0	–	0	
Negative	head	−0.875	regret	−0.6875	miserable	−0.8125	loud	−0.5	although	0	another	0	–	0	
	cross	−0.75	apologize	−0.625	worst	−0.8125	blindly	−0.375	and	0	any	0	–	0	
	darkness	−0.75	burning	−0.625	cheesy	−0.75	outrageously	−0.375	because	0	both	0	,	0	
	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	
Most	character	0.875	cheat	0.625	excellent	1	comically	0.625	plus	0.75	all	0.5	–	0	
Positive	worship	0.75	cheating	0.625	fantabulous	1	thoroughly	0.625	#endgunviolencetogether	0	some	0.25	–	0	
	choice	0.625	flirting	0.625	greatest	0.875	basically	0.5	although	0	a	0	–	0	
	corruption	0.625	guarded	0.625	#beautiful	0.75	entirely	0.5	and	0	an	0	,	0	
	energy	0.625	keeping	0.625	beautiful	0.75	everyday	0.5	because	0	another	0	;	0	
	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	
Negative	s***	−12.5	have	−42.5	happy	33.75	away	−7.1875	as	−0.75	a	0	off**	−0.458333333	
Impact	head	−8.75	keep	−13.125	good	19.6875	still	−3.857142857	#endgunviolencetogether	0	an	0	–	0	
	b****	−8.25	going	−11.25	best	19.125	now	−3.8125	although	0	another	0	–	0	
	support	−8	need	−7.8125	better	14.4375	too	−2.75	and	0	any	0	–	0	
	game	−5.25	stop	−7.5	first	10.125	always	−2.25	because	0	both	0	,	0	
	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	Word	Ratio	
Positive	time	31.875	love	20.25	happy	33.75	just	14.53125	plus	1.5	all	51	–	0	
Impact	chance	15	follow	15.75	good	19.6875	then	6.75	#endgunviolencetogether	0	some	7.5	–	0	
	&amp	13.5	do	10.5	best	19.125	back	6.5	although	0	a	0	–	0	
	y'all	11.3137085	make	10.5	better	14.4375	right	6.1875	and	0	an	0	,	0	
	season	6	win	9.375	first	10.125	well	5.2866	because	0	another	0	;	0	
Twitter Sentiment Summary Statistics for each PoS															
pos	min	first	median	mean	third	max	var	sd							
ADJ	−0.670820393	−0.063331211	0.010206207	0.030348498	0.115617105	0.810133918	0.027338876	0.16534472							
NOUN	−0.670820393	−0.0625	0.01259884	0.023675123	0.108654325	0.810133918	0.022817279	0.151053896							
PUNCT	−0.670820393	−0.053204558	0.010790608	0.026197469	0.102326489	0.810133918	0.021173511	0.145511206							
DET	−0.459279327	−0.054829843	0.023457872	0.032680244	0.118341158	0.810133918	0.023251461	0.152484297							
CON	−0.492762915	−0.077525386	0.005991201	0.016847922	0.110750125	0.506715321	0.022402629	0.149675079							
VERB	−0.510310363	−0.073471884	0.010105037	0.018620968	0.108253175	0.810133918	0.023684674	0.153898259							
ADV	−0.477027835	−0.072168784	0.018263556	0.029046183	0.112152784	0.810133918	0.024988634	0.158077938							
Twitter Sentiment Summary Statistics for each N-GRAM															
	n_gram	min	first	median	mean	third	max								
Most	[ADJ][ADJ]	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835								
Positive	[NOUN][ADJ] [ADV]														
	[ADJ][ADV] [ADJ][NOUN]	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835								
	[ADJ][ADV] [ADJ][NOUN] [ADJ]	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835								
	[ADJ][NOUN] [ADJ][ADV] [ADJ]	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835								
	[ADV][ADJ] [NOUN][ADJ] [ADJ]	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835	−0.477027835								
	n_gram	min	first	median	mean	third	max								

(*continued on next page*)

Table 2 (*continued*)

Semantics																					
Word frequency			Word sentiments						Word impacts						Tweet sentiments						N-Gram Frequency	
			Positive			Negative			Highest			Lowest			Positive			Negative				
Word	Freq		Word	Sent		Word	Sent		Word	Impact		Word	Impact		Tweet	Sentiment		Tweet	Sentiment		N-Gram	Count
Most Negative	[DET][VERB]	0.810133918	0.810133918			0.810133918			0.810133918	0.810133918												
	[NOUN][ADJ][NOUN]	0.61835979	0.61835979			0.61835979			0.61835979	0.61835979												
	[ADV][VERB][NOUN][ADJ][ADV]	0.5625	0.5625			0.5625			0.5625	0.5625												
	[ADJ][ADV][NOUN]	0.530330086	0.530330086			0.530330086			0.530330086	0.530330086												
	[VERB][ADJ][DET][VERB]	0.520031434	0.520031434			0.520031434			0.520031434	0.520031434												
	[DET][ADV][ADJ]																					
	[ADJ][DET][ADJ][PUNCT][ADV]																					

Table 3
The generated Tweets, organized by the highest predicted engagement.

Generated tweet	Tweet sentiment	Estimated engagement
you decisions video is a while tonight the a fact check it a jared and this	0.04	18,953.66
asked which means you cannot the most of thank wheeler climate this she ties dog me midnight airline this mnf game	−0.05	18,949.21
them brother nature go now are the a nineloko how	−0.02	18,948.32
his first rest a your lips i come the a person these	0.17	18,937.96
all of his relatives year a them this	0.08	18,932.61
&gt me the get away deals a**i time? .! .	0.02	18,922.35
a the that we imagined is under was u!: - . the moment	−0.01	18,905.82
everything in invaded &amp you hurt violated the new vid is most stunning attempted the still gon all polar	−0.26	18,902.92
this happened to win a 2016 all heard some fortnite a l t got it the	0.17	18,879.16
reTweet if ur why the united states you is did cousins try it food! a an	0.07	18,879.02
call this: anyone who's a embrace sexy a now when forever when	0.08	18,869.95

predicted engagement level ($p_{i,j}$) for the tweet i using model j. We were able to them compute a cumulative mean absolute percentage error at tweet i (where i can be thought of as a time since the data was ordered chronologically) for model j as:

$$CMAPE_{j,i} = \frac{1}{(i-200)+1}\sum_{k=200}^{i}\frac{|a_i - p_{i,j}|}{a_i}$$

In Fig. 3, we notice the cumulative mean absolute percentage error of each of the 8 models. We notice that models 1, 4, and 6 respectively are the most stable and least biased. As it turns out, these models are the OLS models, and their long term percentage errors appear to stabilize near 14% error. In addition, we wanted to understand the stability of the prediction errors over time. Hence, we found the best prediction error for each Tweet using the model that had the lowest such error. We created a new time series that was the lowest percent error across all models for each tweet. Put differently, for example, for tweet 1, we found the lowest percent error across the 8 models and recorded it. We moved to tweet 2, found the lowest percent error and recorded it. We continued in this fashion.

5.2. Results of demonstration of tweet generation and reporting module

Our code found that Context 4 was the most preferred context according to the popularity analytics module. The output of a single descriptive analysis comprised of the three types discussed in an earlier section, scilicet the semantics, syntax, and interaction of the two. An example output of this analysis for the entire data set for the top 5 in each data category reported is shown in Table 2. For brevity, we excluded the tables of output on the other data sets. However, there were some notable differences in the same data categories for the different data sets. Notable differences include the word frequency, where the top 5 words used from most frequent to least frequent in the entire data set were *trump, win, thanksgiving, don,* and *rt*. In the highest engagement context 4 data set these words were *dat, trump, friend, president, b*****, while in the lowest engagement context 4 data these were *don, tbs, love, s***, trump*. Other differences were also found in the frequency of top specific PoS words. In addition to the descriptive information provided by the procedure outlined earlier, we also had the results from the Tweet generation and predictive model. The Tweets were generated using the generative model with a target engagement level of 41,252. This value was used as input into the Tweet generation

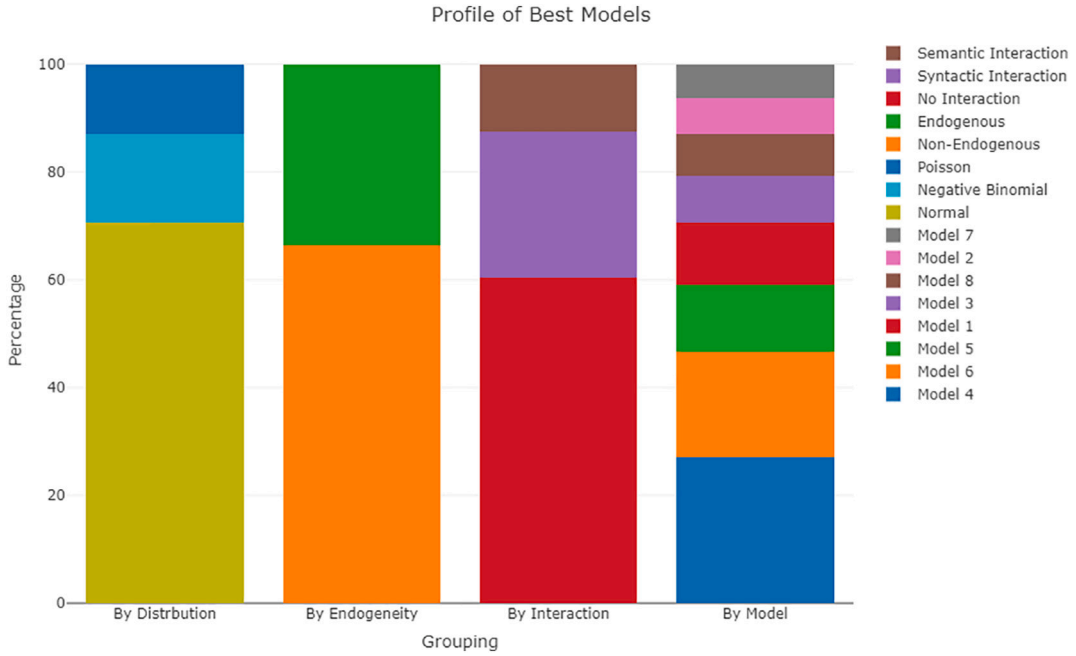

Fig. 2. Distributions of the Best Model broken down by primary characteristic of the model.

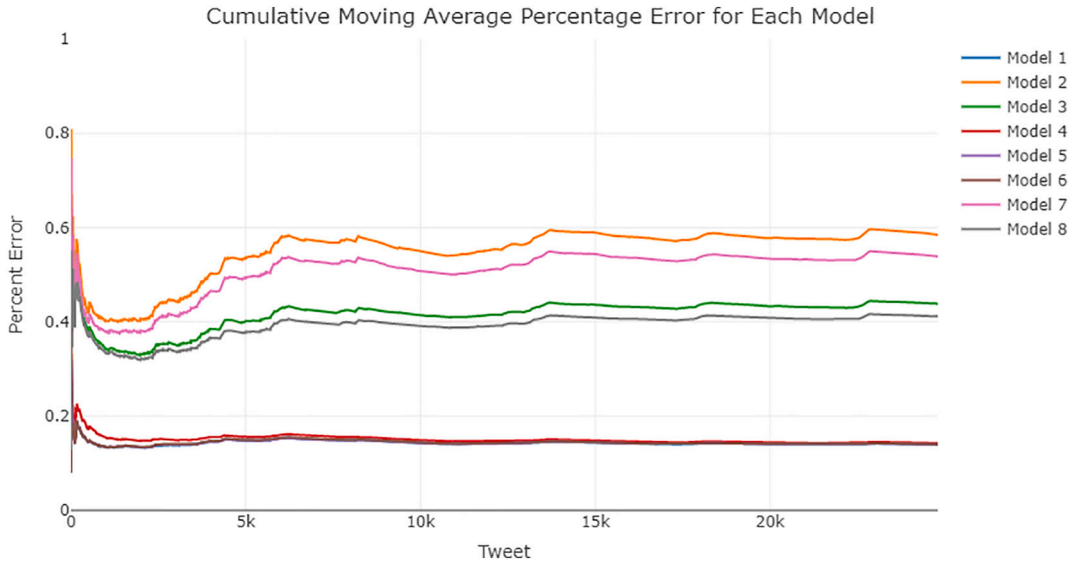

Fig. 3. Cumulative mean absolute percentage error for models 1 to 8.

generative model. As mentioned earlier, the generative model first generates PoS counts. We furthermore have a summary of the generated Tweets shown in Table 3. The Tweets are organized by the highest to lowest predicted engagement levels.

5.3. Results of tweet evaluation demonstration

Upon observing the output from the descriptive and predictive analysis, one could leverage this information to create hand-crafted Tweets, and subsequently score them using our model. We choose not to do this here. However, we will reflect upon some of the characteristics that are salient in the results of our reporting module. We can observe some of the descriptive output through the lens of our predictive output. As we previously illustrated, the"best" model to leverage for our specific test data was Model 7. The factors that contribute to the highest engagement levels include higher ratios of determiners, punctuation, nouns, adverbs, verbs, and total word count. Factors we would like to avoid are using too many hashtags, positive text, conjunctions,adjectives, as well as too many characters. We can thus infer from the model that a"good" Tweet is one that uses concise language, with not many characters, but more words. This would indicate that we should use words which are on average fewer characters per word.

Table 2, under"Twitter Sentiment Summary Statistics for each N-GRAM", indicates that we may seek to use PoS N-Grams such as"[DET][VERB][NOUN][ADJ][NOUN]" and"[ADV][VERB][NOUN][ADJ][ADV]". Since we would like to leverage more negative Tweets as well as more nouns and verbs, Table 2 also indicates we may want to consider using words such as"bulls***","guilt","head","cross", and"darkness" as well as"ashamed","hurting","regret","apologize", and"burning".

Last, we can observer from Table 3 that our system, while not perfect, does generate Tweet Word Sequences that conveys an overall"essence" which should be present in potentially high-engaged Tweets. While

many of these Tweets appears to be non-sense, some meaning can be extracted from some of them. In summary, the manager can essentially take the information from the results of the descriptive analysis as well as the auto-generated Tweet word sequences and use these to generate human-readable messages which will contain the same"essence" that our system is suggesting. Therefore, from these characteristics, a manager could design various candidates of messages, and subsequently pass them through the Tweet evaluation module.

6. Discussion and Implications

Given our proposed system design, we argue that managers can be better equipped at designing messages for use on social media, and more specifically Twitter, by understanding the"essence" of a Tweet. This"essence" is captured by way of two different mechanisms as illustrated in this research. First, our system is designed to analyze and report back a variety of different descriptive statistics in 2 that provide background into the various facets of a"good" social media message. Second, our system leverages information about the past to estimate a generative model that can be used to auto-generate text-suggestions for the user. Samples of these auto-generated tweets are shown in Table 3.

The type of information, we argue, that can aid managers and content creators in the design of potentially high-engagement tweets is demonstrated in part in Table 2. Such a table of descriptive statistics enables the manager to observe the semantic and syntactic features that are present in high and low engagement tweets. For example, suppose a manager of a sporting goods store would like to post a Tweet that will be a very high-sentiment-based tweet. Table 2 indicates that [ADJ][ADJ][NOUN][ADJ][ADV] is a very positive-sentiment-based structure. The manager can then"fill in the words" leveraging information from the most negative and positive sentiment, impact, as well as frequency within each part of speech. So, for example, the manager can fill in the parts of speech sequence specified above with perhaps the following:"One excellent camper worried thoroughly".

The auto-generated raw Tweet-sequence words that we generated above can be further developed to articulate a ready-to-post Tweet. As is obvious from Table 3, most of the messages cannot be directly used as Tweets. This is anticipated given the parsimonious structure of our generative model, which simply aims to identify the core components which posses the maximum popularity-potential. The purpose of these auto-generated messages is to reflect a high potential"essence" in each sequence, and to develop an array of such essence carrying sequences. Such an artificial intelligence generated array of Tweet-sequences would be an extremely valuable support mechanism for managers and decision makers, as they would now have an opportunity to consider sequences primarily driven by context, semantic and syntactic considerations, in their articulations of final ready-to-post Tweets. For example, the Tweet"you decisions video is a while tonight the a fact check it a jared and this" conveys the"essence" regarding a video and fact checking, as well as a time frame. This may convey the idea that videos which are posted online should be done so at night, and should also be fact checked. One could, for example, rework this sentence to something along the lines of"If you decide to post a video at night, please ensure that you fact check its content". This near-ready Tweet can be further fine-tuned using the descriptive output of the system, and further aligned with corporate strategy and branding by human experts to finalize a valuable final Tweet.

Our research illustrates the need for managers of social media accounts to be aware of the various strategic implications of their decision making in content design and network formation. In our research, we addressed the former of these two strategies. More specifically, our research contributes to an underdeveloped area in the extant information systems and marketing science literature, scilicet, the area of semantic and syntactic structure driven auto-generated Tweet-Sequence words suggestion for maximizing popularity. While the extant literature has addressed methodologies to do so for hashtags, we demonstrated in our research that full message suggestion analysis has not been previously explored to its fullest extent. As such, our system design is the first of its kind in this literature. Not only do we synthesize much of the extant literature on the topic of social media analytics and the various behavioral and theoretical findings with respect to drivers and consequences of popularity, but we also have shown how to take these elements and combine them into a system that will aid the decision maker in textual content development. Furthermore, we have also contributed to the literature of drivers of message popularity across different contexts. While our design has shown to be useful and the first of its kind, it is not without its limitations.

First, our design rests on a context-free sentiment analysis. Greater strides can be made upon using different Lexicons and building additional models using more context-specific sentiments. Second, our data collection method may have been skewed due to the timing of the data gathering effort. The excessive number of posts on politics, given that the time frame of data collection occurred during the 2018 Midterm Elections, may have misrepresented the general population of Tweets that often occur on off-election years. Further empirical research may be needed to further evaluate the performance of our system.

In addition, our context determination module was implemented by a simple use of LDA and combining Tweets by time Tweeted. Future studies will need to determine how context can be further refined automatically other than through message aggregation with respect to time. Furthermore, many of the tweets that our system generates do not"make sense". Further research is needed so as to address how a system can not only identify messages that do not"make sense", but also correct them so that they do. Last, our engagement prediction models were based on sentiment analysis. While many methods exist to compute the sentiments, this method of analysis is far from perfect. Our choice of using a parsimonious general-context method of sentiment computation may have led us to improper conclusions regarding our model selection and their performance. Future research may want to consider the impact of different sentiment analysis on the prediction performance of these popularity engagement prediction models.

7. Conclusion

We argue that our research will help aid social media managers in trying to"ride the wave" when they attempt to connect with others and post messages with high popularity potential. The system will allow for the manager to more intelligently design their content strategy. Furthermore, our research can also aid public policy makers and advocacy groups who seek to disperse important information and"piggyback" on information that is trending. Our research also raises new and interesting questions. First, our system is statically designed, where a new model is estimated off new data every day. A potential extension to the design could be to determine how the implementation of reinforcement learning of the context and informal grammar impacts performance. Likewise, our research introduces character properties in the analysis of popularity prediction, namely by looking at the proportion of punctuation to total character count. Further research will need to explore how different types of unigrams impact sentiment polarity within the learned contexts. Last, our research serves as a novel base framework within which to study social media text across varying contexts. However, our definition of "context" is merely a topic. A more ontological investigation can be undertaken in future research to help formalize, mathematically, a"social-media context" to allow for higher-performing textual analytics. Nevertheless, we argue that our research has made great strides in better understanding how managers can design their Tweets to be highly popular so as to achieve strategic and operational excellence.

Credit author statement

Myles Garvey – Literature Review, Illustrations, Data Analysis,

Model Development, Project Management, Coding.

Jim Samuel – Literature Review, Manuscript Design, Data Analysis, Model Development, Project Management, Coding.

Alex Pelaez – Manuscript Design, Review, Model Development, Project Management.

References

[1] A.H. Zadeh, R. Sharda, Modeling brand post popularity dynamics in online social networks, Decis. Support. Syst. 65 (2014) 59–68.

[2] A.S. Abrahams, W. Fan, G.A. Wang, Z. Zhang, J. Jiao, An integrated text analytic framework for product defect discovery, Prod. Oper. Manag. 24 (6) (2015) 975–990.

[3] R.P. Schumaker, H. Chen, A quantitative stock prediction system based on financial news, Inf. Process. Manag. 45 (5) (2009) 571–583.

[4] C.K. Coursaris, W. van Osch, B.A. Balogh, Do facebook likes lead to shares or sales? exploring the empirical links between social media content, brand equity, purchase intention, and engagement, in: 2016 49th Hawaii international conference on system sciences (HICSS), IEEE, 2016, pp. 3546–3555.

[5] U. Manzoor, S.A. Baig, M. Hashim, A. Sami, Impact of social media marketing on consumer's purchase intentions: the mediating role of customer trust, Int. J. Entrepren. Res. 3 (2) (2020) 41–48.

[6] D. Larson, V. Chang, A review and future direction of agile, business intelligence, analytics and data science, Int. J. Inf. Manag. 36 (5) (2016) 700–710.

[7] M.D. Garvey, S. Carnovale, S. Yeniyurt, An analytical framework for supply network risk propagation: a bayesian network approach, Eur. J. Oper. Res. 243 (2) (2015) 618–627.

[8] M.D. Garvey, S. Carnovale, The rippled newsvendor: a new inventory framework for modelling supply chain risk severity in the presence of risk propagation, Int. J. Prod. Econ. 107752 (2020).

[9] S. Goel, A. Anderson, J. Hofman, D.J. Watts, The structural virality of online diffusion, Manag. Sci. 62 (1) (2015) 180–196.

[10] S.A. Delre, W. Jager, T.H. Bijmolt, M.A. Janssen, Will it spread or not? The effects of social influences and network topology on innovation diffusion, J. Prod. Innov. Manag. 27 (2) (2010) 267–282.

[11] K. De Valck, G.H. Van Bruggen, B. Wierenga, Virtual communities: a marketing perspective, Decis. Support. Syst. 47 (3) (2009) 185–203.

[12] S. Stieglitz, L. Dang-Xuan, Emotions and information diffusion in social media—sentiment of microblogs and sharing behavior, J. Manag. Inf. Syst. 29 (4) (2013) 217–248.

[13] E. Pancer, V. Chandler, M. Poole, T.J. Noseworthy, How readability shapes social media engagement, J. Consum. Psychol. 29 (2) (2019) 262–270.

[14] J. Samuel, R. Holowczak, A. Pelaez, The effects of technology driven information categories on performance in electronic trading markets, J. Inform. Technol. Manag.

[15] E. Akpinar, J. Berger, Valuable virality, J. Mark. Res. 54 (2) (2017) 318–330.

[16] J. Shore, J. Baek, C. Dellarocas, Network structure and patterns of information diversity on twitter, Manag. Inf. Syst. Q. 42 (3) (2018) 849–972.

[17] A.S. Abrahams, J. Jiao, G.A. Wang, W. Fan, Vehicle defect discovery from social media, Decis. Support. Syst. 54 (1) (2012) 87–97.

[18] K.K. Kapoor, K. Tamilmani, N.P. Rana, P. Patil, Y.K. Dwivedi, S. Nerur, Advances in social media research: past, present and future, Inf. Syst. Front. 20 (3) (2018) 531–558.

[19] Z. Chi, L. Dong, F. Wei, W. Wang, X.-L. Mao, H. Huang, Cross-lingual natural language generation via pre-training, in: AAAI, 2020, pp. 7570–7577.

[20] O. Dušek, J. Novikova, V. Rieser, Evaluating the state-of-the-art of end-to-end natural language generation: the e2e nlg challenge, Comput. Speech Lang. 59 (2020) 123–156.

[21] D. Price, An AI Breaks the Writing Barrier, URL, https://www.wsj.com/articles/an-ai-breaks-the-writing-barrier-11598068862, 2020.

[22] T. Simonite, Did a Person Write this Headline, or a Machine?, URL, https://www.wired.com/story/ai-text-generator-gpt-3-learning-language-fitfully/, 2020.

[23] K. Panetta, Deep Learning and Natural-Language Generation will Become Standards in Analytics, URL, https://www.gartner.com/smarterwithgartner/neural-networks-and-modern-bi-platforms-will-evolve-data-and-analytics/, 2020.

[24] S. Bapna, M.J. Benner, L. Qiu, Nurturing online communities: An empirical investigation, MIS Quarterly 43 (2) (2019). Chicago.

[25] A. Susarla, J.-H. Oh, Y. Tan, Influentials, imitables, or susceptibles? Virality and word-of-mouth conversations in online social networks, J. Manag. Inf. Syst. 33 (1) (2016) 139–170.

[26] B. Wu, H. Shen, Analyzing and predicting news popularity on twitter, Int. J. Inf. Manag. 35 (6) (2015) 702–711.

[27] J. Samuel, M. Garvey, R. Kashyap, That message went viral?! Exploratory analytics and sentiment analysis into the propagation of tweets, in: Annual Proceedings of Northeast Decision Sciences Institute (NEDSI), 2019.

[28] R.P. Schumaker, H. Chen, Textual analysis of stock market prediction using breaking financial news: The azfin text system, ACM Trans. Inform. Syst. 27 (2) (2009) 12.

[29] X. Guo, Q. Wei, G. Chen, J. Zhang, D. Qiao, Extracting representative information on intra-organizational blogging platforms, MIS Q. 41 (4) (2017) 1105–1127.

[30] L. Dessart, Social media engagement: a model of antecedents and relational outcomes, J. Mark. Manag. 33 (5–6) (2017) 375–399.

[31] H.-S. Lee, H.-R. Lee, J.-U. Park, Y.-S. Han, An abusive text detection system based on enhanced abusive and non-abusive word lists, Decis. Support. Syst. 113 (2018) 22–31.

[32] A.S. Abrahams, J. Jiao, W. Fan, G.A. Wang, Z. Zhang, What's buzzing in the blizzard of buzz? Automotive component isolation in social media postings, Decis. Support. Syst. 55 (4) (2013) 871–882.

[33] W. Li, H. Chen, J.F. Nunamaker Jr., Identifying and profiling key sellers in cyber carding community: Azsecure text mining system, J. Manag. Inf. Syst. 33 (4) (2016) 1059–1086.

[34] W. Dong, S. Liao, Z. Zhang, Leveraging financial social media data for corporate fraud detection, J. Manag. Inf. Syst. 35 (2) (2018) 461–487.

[35] M. Ghiassi, D. Zimbra, S. Lee, Targeted twitter sentiment analysis for brands using supervised feature engineering and the dynamic architecture for artificial neural networks, J. Manag. Inf. Syst. 33 (4) (2016) 1034–1058.

[36] X. Li, M. Wang, T.-P. Liang, A multi-theoretical kernel-based approach to social network-based recommendation, Decis. Support. Syst. 65 (2014) 95–104.

[37] X. Liu, H. Shin, A. C. Burns, Examining the impact of luxury brand's social media marketing on customer engagement: using big data analytics and natural language processing, J. Bus. Res.

[38] S. Tsugawa, H. Ohsaki, On the relation between message sentiment and its virality on social media, Soc. Netw. Anal. Min. 7 (1) (2017) 19.

[39] Z. Ding, X. Qiu, Q. Zhang, X. Huang, Learning topical translation model for microblog hashtag suggestion, in: Twenty-Third International Joint Conference on Artificial Intelligence, 2013.

[40] J. Li, H. Xu, Suggest what to tag: recommending more precise hashtags based on users' dynamic interests and streaming tweet content, Knowl.-Based Syst. 106 (2016) 196–205.

[41] A. Tariq, A. Karim, F. Gomez, H. Foroosh, Exploiting topical perceptions over multi-lingual text for hashtag suggestion on twitter, in: The Twenty-Sixth International FLAIRS Conference, 2013.

[42] D. Lee, K. Hosanagar, H.S. Nair, Advertising content and consumer engagement on social media: evidence from facebook, Manag. Sci. 64 (11) (2018) 5105–5131.

[43] T. Havakhor, A.A. Soror, R. Sabherwal, Diffusion of knowledge in social media networks: effects of reputation mechanisms and distribution of knowledge roles, Inf. Syst. J. 28 (1) (2018) 104–141.

[44] C. Xia, Z. Wang, C. Zheng, Q. Guo, Y. Shi, M. Dehmer, Z. Chen, A new coupled disease-awareness spreading model with mass media on multiplex networks, Inf. Sci. 471 (2019) 185–200.

[45] T. Chesney, Networked individuals predict a community wide outcome from their local information, Decis. Support. Syst. 57 (2014) 11–21.

[46] X. Wang, L. Chen, J. Shi, T.-Q. Peng, What makes cancer information viral on social media? Comput. Hum. Behav. 93 (2019) 149–156.

[47] R. Syed, M. Rahafrooz, J.M. Keisler, What it takes to get retweeted: an analysis of software vulnerability messages, Comput. Hum. Behav. 80 (2018) 207–215.

[48] S. He, X. Zheng, D. Zeng, A model-free scheme for meme ranking in social media, Decis. Support. Syst. 81 (2016) 1–11.

[49] X. Zeng, L. Wei, Social ties and user content generation: evidence from flickr, Inf. Syst. Res. 24 (1) (2013) 71–87.

[50] S. Valenzuela, M. Piña, J. Ramírez, Behavioral effects of framing on social media users: how conflict, economic, human interest, and morality frames drive news sharing, J. Commun. 67 (5) (2017) 803–826.

[51] Y.-M. Li, L.-F. Lin, C.-C. Ho, A social route recommender mechanism for store shopping support, Decis. Support. Syst. 94 (2017) 97–108.

[52] S. Deng, A.P. Sinha, H. Zhao, Adapting sentiment lexicons to domain-specific social media texts, Decis. Support. Syst. 94 (2017) 65–76.

[53] I. Pentina, V. Guilloux, A.C. Micu, Exploring social media engagement behaviors in the context of luxury brands, J. Advert. 47 (1) (2018) 55–69.

[54] O. Sabri, Does viral communication context increase the harmfulness of controversial taboo advertising? J. Bus. Ethics 141 (2) (2017) 235–247.

[55] Y. Liu, C. Jiang, H. Zhao, Using contextual features and multi-view ensemble learning in product defect identification from online discussion forums, Decis. Support. Syst. 105 (2018) 1–12.

[56] C.C. Aggarwal, C. Zhai, Mining Text Data, Springer Science & Business Media, 2012.

[57] F.H. Khan, S. Bashir, U. Qamar, Tom: twitter opinion mining framework using hybrid classification scheme, Decis. Support. Syst. 57 (2014) 245–257.

[58] H. Dutta, K.H. Kwon, H.R. Rao, A system for intergroup prejudice detection: the case of microblogging under terrorist attacks, Decis. Support. Syst. 113 (2018) 11–21.

[59] S. Jiang, H. Chen, J.F. Nunamaker, D. Zimbra, Analyzing firm-specific social media and market: a stakeholder-based event analysis framework, Decis. Support. Syst. 67 (2014) 30–39.

[60] R.Y. Lau, C. Li, S.S. Liao, Social analytics: learning fuzzy product ontologies for aspect-oriented sentiment analysis, Decis. Support. Syst. 65 (2014) 80–94.

[61] R. Gruss, E. Kim, A. Abrahams, Engaging restaurant customers on facebook: the power of belongingness appeals on social media, J. Hosp. Tour. Res. 44 (2) (2020) 201–228.

[62] J. Samuel, Information token driven machine learning for electronic markets: performance effects in behavioral financial big data analytics, J. Inform. Syst. Technol. Manag. 14 (3) (2017) 371–383.

[63] F. Villarroel Ordenes, D. Grewal, S. Ludwig, K.D. Ruyter, D. Mahr, M. Wetzels, Cutting through content clutter: how speech and image acts drive consumer sharing of social media brand messages, J. Consum. Res. 45 (5) (2018) 988–1012.

[64] W.W. Xu, C. Zhang, Sentiment, richness, authority, and relevance model of information sharing during social crises—the case of# mh370 tweets, Comput. Hum. Behav. 89 (2018) 199–206.

[65] Y. Gong, Q. Zhang, X. Huang, Hashtag recommendation for multimodal microblog posts, Neurocomputing 272 (2018) 170–177.
[66] E. Lloret, M. Palomar, Towards automatic tweet generation: a comparative study from the text summarization perspective in the journalism genre, Expert Syst. Appl. 40 (16) (2013) 6624–6630.
[67] C. Lofi, R. Krestel, iparticipate: automatic tweet generation from local government data, in: International Conference on Database Systems for Advanced Applications, Springer, 2012, pp. 295–298.
[68] S. Haustein, T.D. Bowman, K. Holmberg, A. Tsou, C.R. Sugimoto, V. Larivière, Tweets as impact indicators: examining the implications of automated "bot" accounts on t witter, J. Assoc. Inf. Sci. Technol. 67 (1) (2016) 232–238.
[69] W. Xie, F. Zhu, J. Jiang, E.-P. Lim, K. Wang, Topicsketch: real-time bursty topic detection from twitter, IEEE Trans. Knowl. Data Eng. 28 (8) (2016) 2216–2229.
[70] J. Samuel, R. Holowczak, R. Benbunan-Fich, I. Levine, Automating discovery of dominance in synchronous computer-mediated communication, in: 2014 47th Hawaii International Conference on System Sciences, IEEE, 2014, pp. 1804–1812.
[71] S. Pandey, S.K. Pandey, Applying natural language processing capabilities in computerized textual analysis to measure organizational culture, Organ. Res. Methods 22 (3) (2019) 765–797.
[72] A. Gatt, E. Krahmer, Survey of the state of the art in natural language generation: Core tasks, applications and evaluation, J. Artif. Intell. Res. 61 (2018) 65–170.
[73] J. Samuel, G. Ali, M. Rahman, E. Esawi, Y. Samuel, et al., Covid-19 public sentiment insights and machine learning for tweets classification, Information 11 (6) (2020) 314.
[74] J. Samuel, M.M. Rahman, G.G.M.N. Ali, Y. Samuel, A. Pelaez, P.H.J. Chong, M. Yakubov, Feeling positive about reopening? New normal scenarios from covid-19 us reopen sentiment analytics, IEEE Access 8 (2020) 142173–142190.
[75] R. Dale, Natural language generation: the commercial state of the art in 2020, Nat. Lang. Eng. 26 (4) (2020) 481–487.
[76] R. Perera, P. Nand, Recent advances in natural language generation: a survey and classification of the empirical literature, Comput. Inform. 36 (1) (2017) 1–32.
[77] K. Lin, D. Li, X. He, Z. Zhang, M.-T. Sun, Adversarial ranking for language generation, in: Advances in Neural Information Processing Systems, 2017, pp. 3155–3165.
[78] N. Tandon, A.S. Varde, G. de Melo, Commonsense knowledge in machine intelligence, ACM SIGMOD Rec. 46 (4) (2018) 49–52.
[79] I. Androutsopoulos, P. Malakasiotis, A survey of paraphrasing and textual entailment methods, J. Artif. Intell. Res. 38 (2010) 135–187.
[80] K. McKeown, Text Generation, Cambridge University Press, 1992.
[81] R. Soricut, D. Marcu, Towards developing generation algorithms for text-to-text applications, in: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), 2005, pp. 66–74.
[82] R. Manurung, G. Ritchie, H. Thompson, Using genetic algorithms to create meaningful poetic text, J. Exp. Theoret. Artific. Intellig. 24 (1) (2012) 43–64.
[83] V. Klimkov, A. Moinet, A. Nadolski, T. Drugman, Parameter generation algorithms for text-to-speech synthesis with recurrent neural networks, in: 2018 IEEE Spoken Language Technology Workshop (SLT), IEEE, 2018, pp. 626–631.
[84] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (Jan) (2003) 993–1022.
[85] X. Cheng, X. Yan, Y. Lan, J. Guo, Btm: topic modeling over short texts, IEEE Trans. Knowl. Data Eng. 26 (12) (2014) 2928–2941.
[86] Y. Zuo, J. Wu, H. Zhang, H. Lin, F. Wang, K. Xu, H. Xiong, Topic modeling of short texts: A pseudo-document view, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 2105–2114.
[87] K.P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.
[88] M. Meire, M. Ballings, D. Van den Poel, The added value of auxiliary data in sentiment analysis of facebook posts, Decis. Support. Syst. 89 (2016) 98–112.
[89] E. Cambria, B. White, Jumping nlp curves: a review of natural language processing research, IEEE Comput. Intell. Mag. 9 (2) (2014) 48–57.
[90] T. Nasukawa, J. Yi, Sentiment analysis: Capturing favorability using natural language processing, in: Proceedings of the 2nd International Conference on Knowledge Capture, 2003, pp. 70–77.
[91] S. Stieglitz, L. Dang-Xuan, A. Bruns, C. Neuberger, Social media analytics, Bus. Inf. Syst. Eng. 6 (2) (2014) 89–96.
[92] N.F. Da Silva, E.R. Hruschka, E.R. Hruschka Jr., Tweet sentiment analysis with classifier ensembles, Decis. Support. Syst. 66 (2014) 170–179.
[93] Y. Yu, W. Duan, Q. Cao, The impact of social and conventional media on firm equity value: a sentiment analysis approach, Decis. Support. Syst. 55 (4) (2013) 919–926.
[94] D. Wu, Y. Cui, Disaster early warning and damage assessment analysis using social media data and geo-location information, Decis. Support. Syst. 111 (2018) 48–59.
[95] A.C. Cameron, P.K. Trivedi, Regression Analysis of Count Data 53, Cambridge University Press, 2013.
[96] W.H. Greene, Econometric Analysis, Pearson Education India, 2003.
[97] S. Chib, E. Greenberg, Understanding the metropolis-Hastings algorithm, Am. Stat. 49 (4) (1995) 327–335.
[98] M. Mechura, Lemmatization list: English (en) [data file], Retrived from, http://www.lexiconista.com, 2016.
[99] J.M. Wooldridge, Econometric Analysis of Cross Section and Panel Data, MIT press, 2010.