Mining Effective Words For Climate Change Communication

Lazar Milikic
Section of Computer Science
EPFL

Yurui Zhu
Section of Digital Humanities
EPFL

Marko Lisicic
Section of Computer Science
EPFL

***Abstract*—In order to garner more effective attention on Twitter for the topic of climate change, our project aims to design and implement three interpretable models to predict the winning tweet that has better engagement inside each tweet pair, and then to identify words, phrases, and visual appeals (e.g., image, video, hashtag, etc.) that could increase engagement by interpreting the learned parameters. Our models' accuracy is approximately 60%, and we are able to identify patterns in the most engaging words and phrases, which we believe can be used as strategies when composing new tweets that aim to draw attention.**

I. Introduction

In recent decades, climate change has become a major concern. Global interest in this topic has grown in recent years as the issue becomes more prominent and its implications more apparent. Despite the scientific community's efforts to highlight the urgency and importance of this issue [1], [2], [3], the matter remains controversial. Many organizations and people have turned to social media to raise climate change awareness. Twitter has become a venue for sharing information, stories, and opinions about climate change impacts and solutions.

Therefore, in order to help the movement in raising global attention, and encouraging people to take action and even pursue finding a sustainable solution, we aim in this project to explore which expressions tend to drive more engagement, and further help a tweet achieve better outreach and interest the general population about climate change.

To do this, we form comparable tweet pairs with different engagement scores[1], and then train three variations of Bradley Terry model (BTM) to calculate the probability that one tweet beats another. These models are in essence all linear, as our main goal is not to predict the "winning" tweet, but to capture why one tweet wins over the others[2]. Thus, to keep learned parameters interpretable, we insist on using only linear models knowing that their accuracy inevitably must suffer.

Contributions. We identify what kind of language and expressions could help organizations and individuals raise awareness about the climate change issue more effectively, and words that are more audience-specific words for different kinds of authors.

[1]Twitter algorithm determines the engagement score of a tweet by summing its like, retweet, and reply counts [4]

[2]We will call a tweet with higher engagement within each pair the "winning" tweet or the winner.

II. Data

This project used a dataset collected using the Twitter API, based on the Climate Change Taxonomy created by the United Nations Global Pulse[7], including keywords and hashtags relevant to climate change topic.

The dataset comprises 2'041'921 tweets from 2021-01-01 to 2021-07-22, including all kinds of information such as the author id, text, attachment, and public metrics including the number of likes, retweets and replies. Only the tweets with English as language were kept, reducing the set to 1'931'515 tweets. As a large percentage (94.6%) of the tweets are in English, a similar analysis would have been difficult for other languages due to the lack of data. We also dropped attributes such as tweet source, context annotations and their geo-information since they are not related to this project.

III. Methods and Models

As discussed, the main goal is to find out the most engaging words, expressions, or even acronyms that could draw attention. We approach this problem through our secondary goal, predicting which of two comparable tweets turns out to be more successful and appealing. In other words, we wish to design a Bradley–Terry model (BTM) that can predict the winner of a paired tweets[6]. The idea behind is to train a model that assign larger scores to more engaging tweets, so we can use these scores to answer our primary question.

To keep the scores meaningful and interpretable, we need to design a Bradley–Terry model such that:

1) The output scores for each tweet's text can be deconstructed to the word level, i.e., tweet scores are a linear combination of word scores.
2) The model is linear, hence learned weights should directly correspond to the role the feature plays in predicting the correct label

A. Word and Tweet Embedding

To feed tweets' text to machine learning models and perform proper analyses, it is necessary to convert the text into a vector representation. We encode each word to a high-dimensional vector that captures its meaning and relationship to other words. Hence, for a tweet t with n words, we obtain n word embeddings denoted as $\{word_1, word_2, \ldots, word_n\}$. Finally,

the text of the whole tweet is embedded as an average of all embeddings of its words:

$$d(t) = \frac{1}{n}\sum_{i=1}^{n} word_i \tag{1}$$

where, $d(t)$ denotes the document emebdding of a tweet t.

Word embeddings require considerable preprocessing of tweets. First, we remove in-text hyperlinks, usernames, special characters, and punctuation. We translate emojis from texts using *emoji* and *emoji_translate* libraries. We remove the # symbol from hashtags since we consider them words. We conclude preparing tweets by tokenizing extracted text.

Now we can apply the Fasttext[5] word vector model, pre-trained on Common Crawl and Wikipedia[8] to embed our tweets. FastText is convenient as it represents words as combinations of their sub-word units, taking sub-word information into account. This is advantageous for dealing with out-of-vocabulary words and variances in spelling, which are typical in social media texts like tweets.

Furthermore, we generate bigrams from tokens and embed them as an average of two words that constitute it. We do this to further support the interpretation by providing context to individual words as they alone tend to contain limited information, e.g., abbreviations and words with multiple meanings.

B. Metadata Labels

In addition to the text of a tweet, we wish to understand the effects of using in-text hyperlinks, hashtags, Gifs, videos and images on the tweet's engagement. Thus, we encode five additional binary features that characterize whether tweets have the five aforementioned metadata. To distinguish these metadata labels from word embedding values and to stress their relevance in interaction, we label 10 instead of 1 to indicate that a tweet has that feature and 0 otherwise.

C. Bradley-Terry model

Once, we have all the features ready, we can build and train a Bradley-Terry model to achieve our secondary goal as described above. Originally, the Bradley-Terry model (BTM) was designed as a probability model for predicting the outcome of a paired comparison: what is the probability for each item of the pair to win when they are matched up together?

In this project, items of each pair are two tweets, the outcome is which tweet is more engaging, and the Bradley-Terry model is designed as the Logistic regression model predicting the winning tweet.

Let t_0 and t_1 be the two tweets that we wish to compare that have n and m words respectively. Next, let $d(t_0) \in \mathbf{R}^D$ and $d(t_1) \in \mathbf{R}^D$ be their respective document embeddings such that D represents dimension of embedding (See 3.A), then the **BTM-init** model is given as follows,

$$\begin{aligned} p(1|\, d(t_{0,1}), w) &= \sigma\left(w^T d(t_1) - w^T d(t_0)\right) \\ &= \sigma\left(w^T \frac{1}{m}\sum_{i=1}^{m} word_i^1 - w^T \frac{1}{n}\sum_{i=1}^{n} word_i^0\right) \end{aligned} \tag{2}$$

where $p(1|\, t_0, t_1, w)$ denotes the probability that t_1 wins (more engaging) given $d(t_0)$, $d(t_1)$ and learnable weights $w \in \mathbf{R}^D$. Also, σ denotes the Sigmoid function. If on top of the document embeddings for each tweet we add metadata as features of the model (See 3.B), we call that model **BTM-meta**. Word embeddings with metadata of a tweet t, we denote as $d(t)_m \in \mathbf{R}^{D+5}$, as there is 5 additional meta-features.

In this setting, we could train BTM-init and BTM-meta on any pair of tweets. However, to reduce noise and confounding effects, we limited which tweets may be paired. First, we don't include equal-engagement tweet pairs because we can't label the winner. Next, we only consider tweets by the same author sent within 7 days to reduce confounding effects. To further decrease noise and accelerate learning, we avoid pairing tweets with less than 10 counts or 10% difference (whichever is larger) in engagement scores.

D. Latent Author Vector

Both BTM-init and BTM-meta models want to generalize the problem and learn which terms (embeddings) offer higher success for all authors. However, we know that various authors have distinct audiences and that audience choices might vary greatly. To increase our BTM model's learning capacity, we wish to account for these variations by using the Latent Author vector, which reflects the average embedding of all author's tweets. In other words, the Latent Author vector captures the typical word type of an author and, inadvertently, indirectly captures the latent preferences of the author's audience because the document embedding of a tweet is simply the average embedding of all words in this.

Hence, for pair of tweets t_0 and t_1 posted by the same author a with the Latent Author vector denoted as $LV(a) \in \mathbf{R}^{D+5}$ (D is the dimension of embeddings and 5 is the number of meta-features) we design the BTM-latent model as follows,

$$s_0 = w^T d_m(t_0) + LV(a)^T W d_m(t_0) \tag{3}$$

$$s_1 = w^T d_m(t_1) + LV(a)^T W d_m(t_1) \tag{4}$$

$$p(1|\, d_m(t_0), d_m(t_1), w, W) = \sigma(s_1 - s_0) \tag{5}$$

where again $p(1|\, t_0, t_1, w)$ denotes the probability that t_1 wins given $d_m(t_0)$, $d_m(t_1)$ and learnable weights $w \in \mathbf{R}^{D+5}$ and $W \in \mathbf{R}^{(D+5)\times(D+5)}$. σ represents the Sigmoid function.

E. Model selection and evaluation

Since we work with a huge number of tweets, we can form a large number of pairs between them (over 5 million pairs), even under the constraints we set. Cross-validation isn't practical and advantageous because we can evaluate our models using conventional validation on the huge validation set (over 750 thousand pairs). Thus, for model selection, we divide our dataset of pairs into training, validation, and test(for evaluation) sets with 70%, 15%, and 15% ratios. To make results more realistic, we separate data by tweet posting time: train set comprises older tweets, test set contains newer tweets.

As the constructed pairs described in section 3.C are rather balanced with respect to the winning tweet label, so that

the classic categorical accuracy can be considered a reliable metrics to evaluate models.

F. Interpretation

Since all of our proposed models are linear and based on equations (1), (2), (3), and (4), the scores that our models produce for each tweet are the sum of all word scores. Thus, the BTM results are interpretable (See section 3).

In addition, BTM models learn to predict the winning tweet based on the difference in model weights given to each tweet embedding (see equations (2), (3), and (4)), therefore they assign higher scores to the winning tweet. A word that appeared in several winning tweets should have a high score, and visa-versa. Therefore, to see which words are more engaging, we just need to compute the scores for each word and look at those with the highest scores. To compute the score for each word, it is sufficient to replace document embeddings with word embeddings and apply formulas (2) (no subtraction, just the first part) or (3) depending on the model.

It is important to note that the corpus contains words that appear either frequently or infrequently. Because of how the data were acquired, words such as "climate" and "change" appear numerous times. There are also numerous single-occurrence words, which are typically meaningless letter combinations or misspelled words. These words can be considered outliers of the whole corpus, and are not used for the final interpretation.

IV. Experiments

We use *PyTorch* library to train the designed models. Training of each model is performed on 40 epochs or until interruption due to the early stopping procedure. Early stopping is set to have a tolerance of 5 epochs. The *AdamW* optimizer is used with a learning rate of 0.001 and weight decay of 0.0. The criterion used is BCEWithLogitsLoss. For each model, we add l_2 regularization, however different regularization parameters can be set for regularizing word embeddings and metadata. That's why BTM-meta has two possible regularization parameters (λ_{embd}, λ_{meta}) and BTM-latent has four parameters (λ_{embd_1}, λ_{meta_1}, λ_{embd_2}, λ_{meta_2}), two for each weight. The best results for each model with their hyperparameters determined through fine-tuning based on the validation set are provided in Table 1.

Model	Batch size	Regularization	Test accuracy
BTM-init	32768	0.0	0.5802
BTM-meta	8192	(0, 0.01)	0.5827
BTM-latent	2048	(0, 0.001, 0, 0)	**0.6057**

Table 1. Performance of the models

Figure 1 shows BTM-init and BTM-latent learning curves. As expected, both models have considerable bias, which can be fixed by improving their learning capacity, adding hidden layers, and applying non-linearities. This would increase model accuracy but complicate interpretation.

As in Table 1, BTM-latent has the highest accuracy, showing that different authors' audiences favor different items and that adding the author's embedding does increase performance.

V. Interpretation

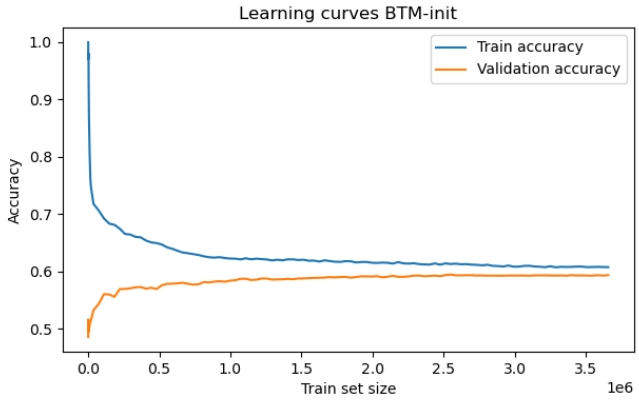
After training the model, we can now interpret it to identify the most appealing words, phrases, and visual appeals, and then summarise strategies for capturing people's attention.

A. Overall Words and Metadata Scores

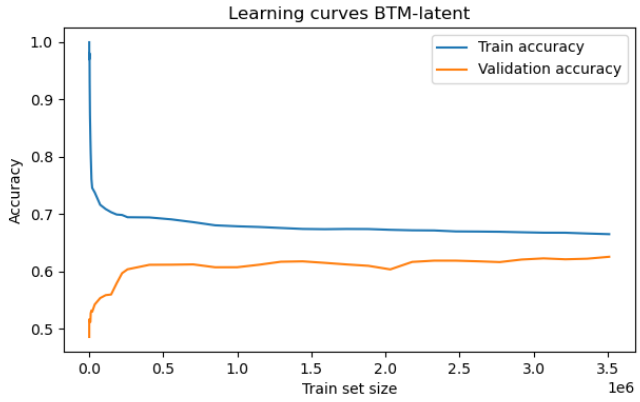
With the weights from the BTM-init model, we can get the scores for different words representing their engagement. The scores of words calculated using the weights in the BTM-meta model were similar to those in the BTM-init model, with an overall 0.1793 higher on average. The top 200 words and bigrams with the highest scores in BTM-init and BTM-init models are shown in Appendix A. Based on this result, we can infer the following strategies:

1) Specific numbers are more likely to draw attention. We notice that the top-ranking words and bigrams contain a lot of numbers and units, such as *sq*, *kg cm*, *95*, *nm*, *mm*. We can infer from this that choosing specific and intuitive numbers rather than descriptions will result in tweets receiving more engagements.
2) Many of the highest-scoring words are related to negative emotions. Words like *liar*, *morons*, *fake*, *idiot*, *stfu* etc. scored very highly, and we speculate that these words may be used by environmentalists to express their dissatisfaction with some news, statements, and people's indifference, or they may be used by anti-environmentalists who are sceptical of some current news and statements. We also observed that words like *die*, *cry*, *extinct*, and *ecocide*, etc., which are typically used to describe negative situations, scored highly. These words imply that the current ecological environment is in a poor state and must be protected, and, to a certain extent, that negative and serious reports are more likely to attract people's attention than positive reports, such as the current achievement.
3) Some celebrity names, including *Pope*,*Obama*, *Putin*, and *(Bill) Nye*, as well as the names of organisations such as the *NYT (New York Times)*, *WWF (World Wide Fund)*, *Fox*, *ICUN (The International Union for Conservation of Nature)*, etc., have high scores. And the word *mp (member of Parliament)* appears many times in top-ranking bigrams associated with other words. This result, on the one hand, confirms the influence of Government officials and organizations on this topic and, on the other, suggests that mentioning or interacting with these influential accounts or public figures could be a strategy for gaining attention.
4) Based on the results, we can infer some of the topics that people are more likely to focus on, such as ecocide (with words *extinction*, *death*), global warming (with words *ice*, *sun*, *glaciers*) and dirty energy (with words *ghg (Greenhouse gas)* , *oil*, *fossil*).

Another interesting finding is that: the word *Y8* got by far the worst score of all other words for both models. *Y8* is the organization where young representatives from *G8* countries



a. Learning curves for the BTM-init model



b. Learning curves for the BTM-latent model

Fig. 1: Learning curves for some of our models. Both figures show that are models suffer from high bias, however, that is the expected price we have to pay to have the interpretable models. BTM-meta has a similar shape of learning curves as BTM-init.

have discussions and come to suggestions or conclusions. It is surprising to see that people actually don't pay attention to how young people react to climate change.

The weights for the metadata labels are shown in table 2. The result implies that having a GIF, video and image attached to the tweet helps draw attention while URLs and hashtags negatively influence tweets' success.

Feature	URL	hashtags	GIFs	video	image
Weights	-0.0820	-0.0138	0.0521	0.0396	0.0383

Table 2. The weights of the metadata features

B. Author Specific Ranking

Using the BTM-latent model, we may calculate the scores of words in the context of different authors to infer their own engaging words, especially for talkative and influential authors. We can calculate overall scores by averaging author vectors and comparing them to the other two models. The average score has increased by 0.4114, but the same patterns remain. Appendix B contains the top 200 words and bigrams.

Then, we choose accounts with over one million followers and determined the top 20 engaging words and bigrams for each account, as well as the top 20 words and bigrams that differed most from the overall score. The latter can be viewed as words that reflect their most popular topics or content based on their characteristics. For instance, the account *@PIB_India* has similar top-ranking bigrams to the overall result, but it also has some unique top-ranking bigrams containing *drought* and *warming* that do not appear in the overall top-ranking bigrams.

In addition, we wish to group the authors according to the content; however, when applying Principal component analysis (PCA) (shown in figure 2) and t-distributed stochastic neighbour embedding (T-SNE) to the author vectors, we were unable to identify any explicit groups. Thus, we select the four authors who are farthest from the centre and from each other (orange point in fig 2), assuming they write tweets that are diametrically opposed and for different audiences. By analysing their engaging words and bigrams, we can see distinct differences. Author 1 has the word *generalawareness*, while Author 3's top-ranking words are all numbers and units.

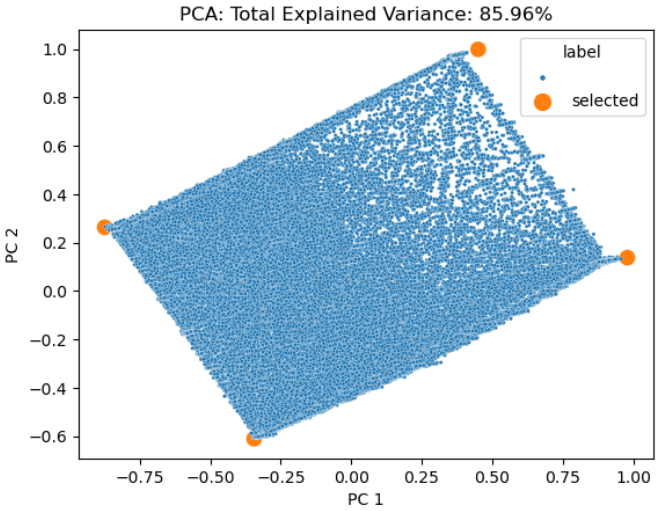


Fig. 2: Visuialzation of Principal component analysis Result

Author 4 has similar words such as *uk* and *g7*, indicating that its audiences may be interested in government affairs.

VI. Conclusion And Future Work

In our project, we employed three interpretable linear models to predict the winners of paired tweets about climate change, and all three models achieved approximately 60% accuracy. By analysing the model's parameters, we identified words, phrases, and visual appeals that could increase the engagement of tweets on this topic. With these results, we can conclude several strategies regarding both the textual level and visual content level. In addition, the implemented BTM-Latent model can help analyse particular authors or author groups.

Future work might focus on adding more binary labels that aren't restricted to content but also include tweet time, text length, etc. Furthermore, in our project, we define engagement as a tweet's likes, retweets, and responses, and a higher engagement signifies a positive impact on climate change. Occasionally, people respond to tweets to disagree or argue. In such cases, our project's engagement score doesn't exactly connect to a tweet's positive impact. Therefore, if we can assess positive engagement, we can create a more effective strategy.

References

[1] Vose, R. S., Easterling, D. R., Kunkel, K. E., LeGrande, A. N., & Wehner, M. F. (2017). Ch. 6: Temperature changes in the United States. climate science special report: Fourth national climate assessment, volume I. https://doi.org/10.7930/j0n29v45

[2] Hayhoe, K., Wuebbles, D. J., Easterling, D. R., Fahey, D. W., Doherty, S., Kossin, J. P., Sweet, W. V., Vose, R. S., & Wehner, M. F. (2018). Chapter 2 : Our changing climate. impacts, risks, and adaptation in the United States: The Fourth national climate assessment, volume II. https://doi.org/10.7930/nca4.2018.ch2

[3] Wuebbles, D. J., Fahey, D. W., Hibbard, K. A., DeAngelo, B., Doherty, S., Hayhoe, K., Horton, R., Kossin, J. P., Taylor, P. C., Waple, A. M., & Yohe, C. P. (2017). Executive summary. climate science special report: Fourth national climate assessment, volume I. https://doi.org/10.7930/j0dj5ctg

[4] Hughes, J. (2022, December 20). How the Twitter algorithm works in 2023. Social Media Marketing & Management Dashboard. Retrieved December 20, 2022, from https://blog.hootsuite.com/twitter-algorithm/

[5] Joulin, A.; Grave, E.; Bojanowski, P.; Mikolov, T. Bag of Tricks for Efficient Text Classification. arXiv August 9, 2016. https://doi.org/10.48550/arXiv.1607.01759.

[6] Kjytay, & Kjytay. (2022, February 2). What is the bradley-terry model? Statistical Odds & Ends. Retrieved December 20, 2022, from https://statisticaloddsandends.wordpress.com/2022/02/01/what-is-the-bradley-terry-model/

[7] UN Global Pulse. (n.d.). Taxonomy (English). How the World Tweets: Climate Change. Retrieved December 20, 2022, from http://unglobalpulse.net/climate/taxonomy/

[8] Grave, E.; Bojanowski, P.; Gupta, P.; Joulin, A.; Mikolov, T. Learning Word Vectors for 157 Languages. arXiv March 28, 2018. https://doi.org/10.48550/arXiv.1802.06893.

[9] Pytorch. PyTorch. (n.d.). Retrieved December 22, 2022, from https://pytorch.org/

Appendix A
Detailed result of BTM-init Model

Here we showed the top 200 words and bigrams with parameters from the BTM-init Model. The result of the BTM-meta Model is very similar.

The Top 200 words are:

sq, kg, cm, mp, ww3, mv, mm, mc, nm, o2, fg, ck, 95, uv, ha, mi, nyt, rug, dnc, ffs, awe,pope, km, msm, sh, bs, 90, dc, ps, ban, hq, alt, bjp, 1kg, yr, gaga, peta, rot, gdp, lbs, 93, b4, ass, sad, 99, sue, cry, ghg, sc, mad, suns, 2050, die, dam, nobel, ww2, rats, 2080, wu, mn, ice, fu, ok, gop, olds, dd, dams, vow, pew, nuke, stfu, 85, act, nazis, 40, 98, potus, rcmp, lv, ch4, dems, af, wtf, sing, 27cm, nukes, txt, dhs, iso, lied, fake, hoax, ship, liars, bc, swan, mf, iucn, ie, ji, gmo, jul, extinct, sued, 3mm, idiots, obama, fbi, nwo, hug, ccp, 2045, men, ms, ark, gaia, oil, omg, war, died, am, mw, ppl, jail, wars, morons, 50, slap, agw, sworn, oprah, rip, fox, pelosi, ripe, muslims, imo, 60, arse, smh, epa, planet, denier, gm, rips, 2100, stans, ft, liar, ss, yrs, rape, harm, g7, ecocide, cr, moron, dies, idiot, czar, iran, asses, wolf, fir, rude, 1m, 2030, libs, brits, she, 47, goose, ever, oceans, old, exxon, 2070, googoo, genocide, blm, unborn, quits, void, kill, hitler, glaciers, bp, muslim, bans, eg, hrs, feet, hes, inuit, bush, mt, fools, vatican, false, 70

And the top 200 Bigrams are:

sq mi, kg ha, sq km, sq ft, sq feet, sq kms, 000 sq, 50 kg, 60 kg, kg bomb, 30 kg, per sq, 7653 kg, 100 kg, 000 kg, 20 kg, million sq, s2 e8, 10 kg, 200k sq, kg per, per kg, 700 sq, sq miles, sq mile, kg co2, billion kg, per m2, kg bags, million kg, kg by, 300 kg, 200 kg, mm mm, kg of, of kg, 000 km2, kg in, bjp mp, kg to, to kg, by m2, kg desciption, 40 cm, kg the, m2 of, one kg, mp sue, 4g kg, 22 kg, and kg, kg and, at kg, 450 kg, 50 cm, kg from, mp majid, 95 uv, million km2, 30 cm, 100 cm, km2 of, kg co2e, ha ha, km2 in, 500 sq, km2 that, mp sir, m2 developments, 3g kg, mp we, 20 cm, 10 cm, mm yr, km2 and, nyt nyt, water hf, cm per, cm is, tr mp, cm has, ndp mp, 90 95, says cm, 90 uv, woman mp, cm are, mp he, cm sea, cm plants, cm square, mp dr, warming mp, kg dm, km mi, cm kejriwal, cm by, by cm, mp was, mp is, under cm, hf 5kiwc, mp has, deputy cm, s2 e5, ww3 we, 93 uv, ww3 putin, mp calls, cm went, of cm, cm of, cm title, title cm, first cm, mp it, desciption px, 15 cm, 99 uv, manipur cm, mp they, says mp, mp says, kg dew, cm eknath, mp warns, all mp, mp defends, 21 cm, dup mp, cm for, mp are, mp claims, mp did, mp after, in cm, cm in, mp calling, mp now, climatechange ico, ico climatechange, 25 cm, mp appointed, appointed mp, cm to, to cm, mp voted, cm long, mp just, 50 mm, losing cm, lb yr, change cm, mp vote, cm year, s2 is, mp tells, hands mp, mp sits, mp ed, cm over, cm desciption, desciption cm, cm since, cm naveen, an mp, the cm, cm the, every mp, by mp, cm rise, cm yogi, 85 uv, mp used, mp speak, sh sh, 98 uv, cm bcn4570, miliband mp, mp president, cm directs, cm sindh, cm nitishkumar, mp if, 19 cm, 90 km, mp backs, parliament mp, bs bs, mp speaks, mp

minister, mp who, british mp, rejecting mp, cm uddhav, 27 cm, cm house, mp mark, workers mp, of mp

APPENDIX B
DETAILED RESULT OF THE BTM-LATENT MODEL

Here we showed the top 200 words and bigrams with parameters from the BTM-latent Model.
The Top 200 words are:

s8, kg, k5, dc, g6, wmw, k3, m2, qotw, k6, obc, lb, k4, hf, ck, 288b, oq, mv, sc, 9per, m1, k8, dq, ico, mc2, pz, xtz, hk, g8, mc, niva, 1v, f9, kv, uv, 9951, km2, dukhan, o2z, k7, 6kg, nkw, sh, qv, ctw, c1, mh, st1, frim, sq, ddamn, bjp, ha, g1, lcf, 27t, utf8, gfe, ch3, bw, mp, lbs, acnur, g2, b3w, gtx, a7, k9, g5, k2, fu, gsm, 30kg, hdc, adb, nc4, 5o, gs, pb, us41, 288bn, m7, pbuh, f6, 4mt, wv, bc, kcal9, g3, bs, cks, dukha, 3959, 27s, o0, gnc, wgbh, disp, vez, sr, kgs, 6d, s2, p4, sqr, duc, ffs, rq, gkc, 60, esc, 8064, ch4, yrs, 50, 1s, clw, 92e, hagee, sl, naacp, 40, ini2021, 9gw, rjm, ddd, 40yrs, 6gw, hha, tsk, msc, ww3, fkn, 3fm, vwe, 800g, 85, ckin, dp, ban, tbm, ps, cd, ajit, starf, ucph, 2kg, 3680, 1kg, 2049, fahn, 8kg, ggfi, dumbf, pds, 5kg, gpc, cu2, da, 90kg, p8, usn, lpg, onepager, msm, cked, deetz, ddf, p9, cr, uq, 2e0, br, l00, cm, j1, 3580, hxl, 065m, yak, cking, othes, k1, p6, ji, nimh, o3, gnw, awais, ww, ssy, xto, opra, gb, fsc, flax, 3hb, or2, 4d, peta

The top 200 bigrams are:

kg ha, 60 kg, 50 kg, 30 kg , 200 kg, 20 kg, kg per, per kg , 10 kg, 300 kg, 7653 kg, kg bags , 000 kg, kg bomb, 100 kg, 450 kg , ac dc, billion kg, kg in, lb yr , million kg, one kg, 22 kg, sh sh , kg from, kg and, and kg, to kg , kg to, of kg, kg of, per m2 , kg the, kg desciption, at kg, kg co2e , per lb, 4g kg, dc has, dc oct , dc md, dc 34, ha ha, 000 lb , outraged dc, dc totem, dc no, bjp mp , dc cancels, dc power, dc never, dc was , 85 uv, dc after, dc usa, kg by , harms dc, say dc, dc don, dc now , dc made, kg co2, dc united, in dc , dc in, dc world, dc who, ck off , dc politicians, dc have, washington dc , dc babylon, for dc, dc for , dc statehood, dc took, dc kennedy , dc will, dc office, on dc, dc on , threatening dc, dc council, dc dozens , dc democrats, dc 20515, dc july , dc fast, dc they, dc you, sh ji , dc snow, around dc, dc demanding , wash dc, dc government, dc if , billion lb, dc that, that dc , dc joined, dc today, 70 uv, dc are , dc this, dc stopmvp, dc cherry , dc time, dc police, 95 uv, energy dc , from dc, dc from, dc clothesline , dc last, dc and, and dc, dc building , deadly dc, dc to, to dc, dc state , of dc, storm dc, dc convened , dc lightning, dc enquirer, dc universe , 90 uv, dc tornadoes, dc dirty , change dc, dc it, dc next, into dc , make dc, dc so, dc rep, dc about , states dc, dc there, dc we, lb full , dc is, bs bs, dc swamp, 75 uv , dc schools, dc yesterday, the dc , dc the, dc metro, dc green, dc highway , dc right, dc new, dc tomorrow, m2 of , dc october, rebellion dc, block dc , dc or, go ck, dc climate, kg dew , kg dm, dc amp, amp dc, dc us , dc desciption, desciption dc, moment dc , dc residents, dc at, at dc, out dc , yrs bc, dc based, dc unhoused, as dc , dc as, our dc, with dc, dc with , dc college, sc has, dc might , dc circuit, dc global, ck all, dc can , dc area, dc comics, lb of, 60 60 , dc bike, dc commuter, dc title , title dc, 60 yrs