Proceedings of the Nineteenth International AAAI Conference on Web and Social Media (ICWSM 2025)

Retrieval Augmented Encoder-Decoder with Diffusion for Sequential Hashtag Recommendation in Disaster Events

Shubhi Bansal, Seerla Parimala, Nagendra Kumar*

Indian Institute of Technology Indore
phd2001201007@iiti.ac.in, cse210001064@iiti.ac.in, nagendra@iiti.ac.in

Abstract

During disasters, access to timely and accurate information is crucial for effective response and recovery efforts. Hashtags have emerged as a lifeline in disaster response, organizing and disseminating critical information on social media, facilitating effective communication and real-time situational awareness. However, existing methods for hashtag recommendation fall short during disasters. Retrieval-based methods rely on fixed predefined hashtag lists, failing to capture dynamic information flow, while generation-based methods lack guidance for generating relevant hashtags. In view of the above, we propose a novel three-stage framework. First, the retriever identifies potential candidate hashtags from a vast collection of tweets annotated with hashtags. Next, a selector narrows down these candidates by analyzing the input tweet and ensuring only the most relevant hashtags are retained. Finally, a diffusion-based sequence-to-sequence encoder-decoder generates informative hashtags by leveraging the refined set of candidate hashtags and the original input tweet. The framework infused with diffusion overcomes the limitations of extant encoder-decoder models that produce generic hashtags due to reliance on maximizing training data likelihood. Our diffusion-based approach excels at capturing the dynamic and informal language of disaster situations by reversing a gradual noising process, allowing it to explore wider possibilities and generate more diverse hashtags. We enhance the generator with self-conditioning for better utilization of predicted sequence information. Furthermore, we devise an adaptive nonlinear noise schedule for balanced denoising across time steps for each token in the generated hashtag sequence. Empirical evaluations reveal that our proposed method exhibits superior performance compared to state-of-the-art hashtag recommendation methods in both the quality of generated hashtags and training time.

Introduction

During disasters, people increasingly rely on social media for updates, assistance, and vital information sharing (Rhodan 2017; Frej 2018). However, this valuable resource remains underutilized due to the sheer volume of information, making it difficult to identify critical updates (Silverman 2017). A study by (Villegas, Martinez, and Krause 2018) found that FEMA's initial damage estimates for Hurricane Harvey overlooked nearly half of the relevant online data, resulting in a significant underestimation of the total cost. This highlights the potential consequences of overlooking online information and the need for tools to effectively filter and utilize it. Hashtags are vital for organizing and disseminating critical information on social networks during disasters. They facilitate effective communication and coordination among emergency responders, government agencies, and public, improving real-time situational awareness. Accurately tagged tweets identify the disaster's nature, location, affected areas, severity, and specific needs of those on the ground. However, approximately half of disaster-related tweets lack informative hashtags (Chowdhury, Caragea, and Caragea 2020), hindering effective response. Therefore, automated hashtag recommendation systems are essential for optimizing information accessibility, efficient filtering of critical updates, improving disaster response, efficient resource allocation, and mitigation efforts.

Recommending hashtags for disaster-related tweets presents unique challenges. The information landscape during disasters is highly dynamic and noisy, with new needs and challenges constantly emerging. Existing retrieval-based and generation-based methods struggle to keep pace with rapidly changing environment. Retrieval-based methods (Cao et al. 2020; Wei et al. 2019; Bansal, Gowda, and Kumar 2024), relying on fixed hashtag lists cannot keep pace with the rapidly changing information. Generation-based methods (Zheng et al. 2021; Mao et al. 2022), though better at understanding new information, may produce inaccurate hashtags without additional guidance. Therefore, disaster-related tweets necessitate a system that can accurately capture evolving needs and challenges faced by affected communities as the situation unfolds, effectively filter and process information from social media data containing informal language and misspellings, and generate hashtags that not only reflect the current situation but also anticipate future needs. Retrieval-Augmented Generation (RAG) techniques provide an effective solution by capitalizing on retrieval and generation approaches. This enables RAG models to leverage existing knowledge while adapting to new information, crucial for hashtag recommendation in disaster scenarios.

Existing hashtag generation methods, predominantly based on encoder-decoder frameworks with Recurrent Neural Networks (RNNs) (Wang et al. 2019; Yang et al. 2020)

*Corresponding Author

or transformers (Mao et al. 2022; Diao et al. 2023) struggle to perform effectively in disaster scenarios. Though tweets have a character limit, RNNs, while capable of capturing sequential information, struggle with long-range dependencies, hindering their ability to process the full context of a tweet. Transformers, while robust, generate generic or repetitive hashtags when faced with the noisy and informal language usage, spelling, and grammar mistakes by users common in disaster situations. Consequently, these methods fail to generate hashtags that accurately reflect the dynamic nature of needs during a disaster, hindering effective communication and response efforts. Inaccurate and irrelevant hashtags can lead to misdirection of resources and delay critical assistance. Diffusion models, which learn to reverse a progressive noising process, have successfully generated high-quality synthetic data across multiple domains (Kong et al. 2020; Ho et al. 2022). This success extends to various Natural Language Processing (NLP) tasks such as unconditional (Austin, Zaïane, and Largeron 2022) and controlled text generation (Li et al. 2022). However, their application to sequence-to-sequence (seq2seq) text generation, particularly for hashtag recommendation in disaster scenarios, remains largely unexplored. Inspired by their potential, we propose the use of diffusion models to recommend hashtags for disaster-related tweets.

To address these challenges, we propose retrieval **A**ugmented encoder-decoder with diffu**S**ion for **S**equent**I**al hashta**G** recomme**N**dation in disast**ER** events (ASSIGNER). The retriever identifies candidate hashtags by searching a large tweet-hashtag corpus for similar tweets and associated hashtags. By comparing the input tweet to retrieved tweets and hashtags, the selector narrows down this collection ensuring that the generator receives the most pertinent hashtags. Our novel diffusion-based generator leverages this refined set and input tweet to generate informative hashtags. It utilizes an encoder-decoder architecture, where the continuous diffusion framework is integrated within the seq2seq generation process. Further, we incorporate self-conditioning and an adaptive non-linear noise scheduler for improved performance. Extensive evaluations on a dataset of disaster-related tweets demonstrate enhanced performance in hashtag generation, achieving superior results in both hashtag quality and training time compared to existing state-of-the-art approaches. Ablation studies confirm the benefits of self-conditioning and the adaptive non-linear noise schedule, highlighting their complementary nature in seq2seq settings.

Our key contributions can be summarized as enlisted below.

- We propose a retrieval augmented diffusion-based seq2seq framework to recommend hashtags for disaster-related tweets. We leverage the synergy of retrieval with the generative power of diffusion models to improve communication and response effectiveness during crises.
- As far as we know, this work presents the first application of diffusion models to hashtag recommendation in disaster scenarios. We adapt the continuous text diffusion model to generate hashtags sequentially using an encoder-decoder transformer architecture.
- We leverage retrieved hashtags from similar tweets to provide contextual information and guide the generation of relevant hashtags for disaster-related tweets.
- Our newly proposed adaptive non-linear noise scheduler significantly improves the quality of generated hashtags by allowing for finer-grained control over the generation process.
- Experiments show that the proposed model performs competitively compared to existing methods in generating high-quality and informative hashtags for tweets about disaster events.

Related Works

Hashtag Recommendation

Keyphrase Extraction: Hashtag recommendation relies on extracting keyphrases directly from the source text (Marujo et al. 2015; Gong, Zhang, and Huang 2015; Zhang et al. 2018). Zhang et al. (2016) identified hashtags as valuable keywords for extracting keyphrases from X (formerly known as Twitter). However, this approach restricts generating hashtags to those already present in the source text. This constraint neglects the creativity of hashtag usage, where users create novel hashtags owing to their backgrounds, proficiency levels and linguistic styles, resulting in suboptimal performance.

Classification: Existing methods that treat hashtag recommendation as a classification problem predefine a limited set of candidate hashtags (20 (Li et al. 2016), 101 (Cao et al. 2020), 1001 (Wei et al. 2019)) and softmax layer to predict the probability of relevant hashtags (Bansal et al. 2024). However, these approaches are limited by their reliance on a fixed set of candidates, which is impractical in dynamic situations such as disasters where new and unforeseen terms may emerge as relevant hashtags. Constantly updating and retraining models, which is time-consuming in urgent situations, makes classification-based approaches unsuitable for disaster-related hashtag recommendation. Thus, we posit it as a generation task to more accurately reflect the natural way users create hashtags.

Generation: Several studies have conceptualized it as a sequence generation task (Wang et al. 2019; Zheng et al. 2021; Mao et al. 2022; Bansal, Gowda, and Kumar 2022), facilitating the creation of diverse and expressive hashtags that effectively convey the core message of the post. However, prior research has given little consideration to popular or trending hashtags. While these approaches produce semantically relevant hashtags, recommended hashtags might not be widely used, hindering the discoverability of content. Some approaches incorporated conversational data to augment the input features (Zhang et al. 2018; Wang et al. 2019) and yield suitable hashtags. However, assuming the availability of conversations prior to annotation is not practical.

Retrieval-Augmented Generation

RAG represents an innovative method that integrates pretrained generative models with information retrieval tech-

niques (Asai et al. 2023; Kocoń et al. 2023). Previous research has explored RAG to address information-driven tasks (Li et al. 2023a; Chen et al. 2023). It has found applications in NLP tasks, such as generating image captions (Ramos, Elliott, and Martins 2023), producing keyphrases (Kim et al. 2021; Gao et al. 2022), neural machine translation (Gu et al. 2018; Hossain, Ghazvininejad, and Zettlemoyer 2020), answering open-ended questions (Guu et al. 2020; Lewis et al. 2020) and knowledge-based dialogue generation (Lian et al. 2019). Furthermore, RAG has been applied to mitigate factual inaccuracies (Raunak, Menezes, and Junczys-Dowmunt 2021), compensate for obsolete knowledge (He, Zhang, and Roth 2022), and improve performance in specialized domains (Li et al. 2023b).

Diffusion

Building on achievements of diffusion models in image generation (Rombach et al. 2022; Song et al. 2021), researchers have investigated their use in text generation. This has led to methods such as Multinomial Diffusion (Hoogeboom et al. 2021) and D3PM (Austin et al. 2021), which tackle the inherent discreteness of text by formulating specialized diffusion processes. DiffusionBERT (He et al. 2023) introduced pre-trained models for language modeling. Bit Diffusion (Chen, Zhang, and Hinton 2022) represents text as binary bits while (Yu et al. 2022) uses a continuous vector space. DiffusionLM (Li et al. 2022) uses semantic embedding spaces and incorporates auxiliary losses for joint learning. Building on DiffusionLM, researchers have improved the quality of synthesized text (Strudel et al. 2022). Extending diffusion models to seq2seq settings, DiffuSeq (Gong et al. 2022) devised encoder-focused framework while SeqDiffuSeq (Yuan et al. 2024) proposed an encoder-decoder paradigm featuring self-conditioning and adaptive noise scheduling. Our work builds upon SeqDiffuSeq, modifying its adaptive nonlinear noise schedule and integrating external knowledge using RAG for sequential hashtag generation.

Methodology

We outline our proposed methodology in this section. Figure 1 illustrates the system architecture. Initially, the system encodes the input tweet. Subsequently, it retrieves semantically similar tweets from a pre-existing tweet-hashtag corpus and extracts their associated hashtags, thereby incorporating external knowledge to enhance relevance. A selector module then refines these retrieved hashtags, filtering and selecting the most pertinent ones based on semantic similarity and relevance, thereby ensuring contextual appropriateness. Following this selection, the system computes embeddings for chosen hashtags and feeds them into a cross-attention mechanism, where attention weights (α values) are calculated to produce a cross-attended feature vector. This vector is then fed into a diffusion-based generative model. This model, employing a Bidirectional AutoRegressive Transformer (BART) encoder-decoder architecture (Lewis 2019), learns the underlying hashtag distribution. During the iterative diffusion process, the system reintroduces previous predictions, along with selected hashtag embeddings as contextual information (self-conditioning), guiding the generative process towards relevant hashtag sequence production. Finally, the decoder generates the final set of hashtag recommendations for the input tweet.

Retrieval-Augmented Generation

To improve the relevance of generated hashtags, we incorporate a retrieval mechanism in our framework that leverages existing knowledge from a curated hashtag-tweet corpus. This module is composed of retriever and selector.

Retriever The retriever module identifies relevant hashtag-tweet pairs from a knowledge database by utilizing the embedding of the input tweet. This approach, inspired by (Zangerle, Gassler, and Specht 2011), leverages the observation that semantically similar tweets often share similar hashtags, reflecting common usage patterns. We construct a knowledge base $\mathcal{D}$ composed of tweet-hashtag pairs (d_i, H_i) where d_i represents a tweet and H_i represents its corresponding set of hashtags. For a new input tweet d_q, the retriever $\mathcal{F}$ compares its embedding with the embedding of every other tweet in the corpus. It then retrieves the top-N most semantically similar tweet-hashtag combinations with corresponding similarity scores.

$$(d_1, H_1, s(d_q, d_1)), ..., (d_N, H_N, s(d_q, d_N)) = \mathcal{F}(d_q|\mathcal{D}) \tag{1}$$

where, $s(d_q, d_i)$ denotes the similarity between the query tweet d_q and the i^{th} retrieved tweet d_i and each H_i contains a set of hashtags $\{h_{i1}, \ldots, h_{i|H_i|}\}$. This retrieval process provides a candidate pool of potentially relevant hashtags based on similar tweets in the knowledge base.

Selector The selector module refines hashtag recommendations by filtering out low-quality and less prevalent hashtags from retrieved pairs. We leverage two key indicators of hashtag prominence to refine the selection process: the semantic relatedness between the input tweet and the retrieved tweet, and the relevance of retrieved hashtags to the input tweet. This multifaceted selection process ensures that chosen hashtags are not only semantically relevant but also reflect popular and widely used terms. The selector is trained using a dataset comprising positive and hard negative samples. Each hashtag in a tweet is considered a positive sample (h^+). To construct hard negative samples (h^-), we employ a BERT-inspired perturbation strategy, where labeled hashtags are modified without altering their semantic meaning. This involves randomly selecting a word within the hashtag to replace it with a synonym, delete it, swap it with an adjacent word, or insert a synonym after it. The resulting training dataset consists of tuples (d_i, h_i^+, h_i^-), where $i = 1, \ldots, N$. The training of the selector module involves minimizing a contrastive loss function, defined as follows:

$$\mathcal{L}_C = -\log \frac{e^{sim(E_{d_i}, E_{h_i^+})/\tau}}{\sum_{j=1}^{L}(e^{sim(E_{d_i}, E_{h_j^+})/\tau} + e^{sim(E_{d_i}, E_{h_j^-})/\tau})} \tag{2}$$

where, sim represents cosine similarity, E_d denotes the embedding of d, L is the mini-batch size, and τ is a temperature hyperparameter.

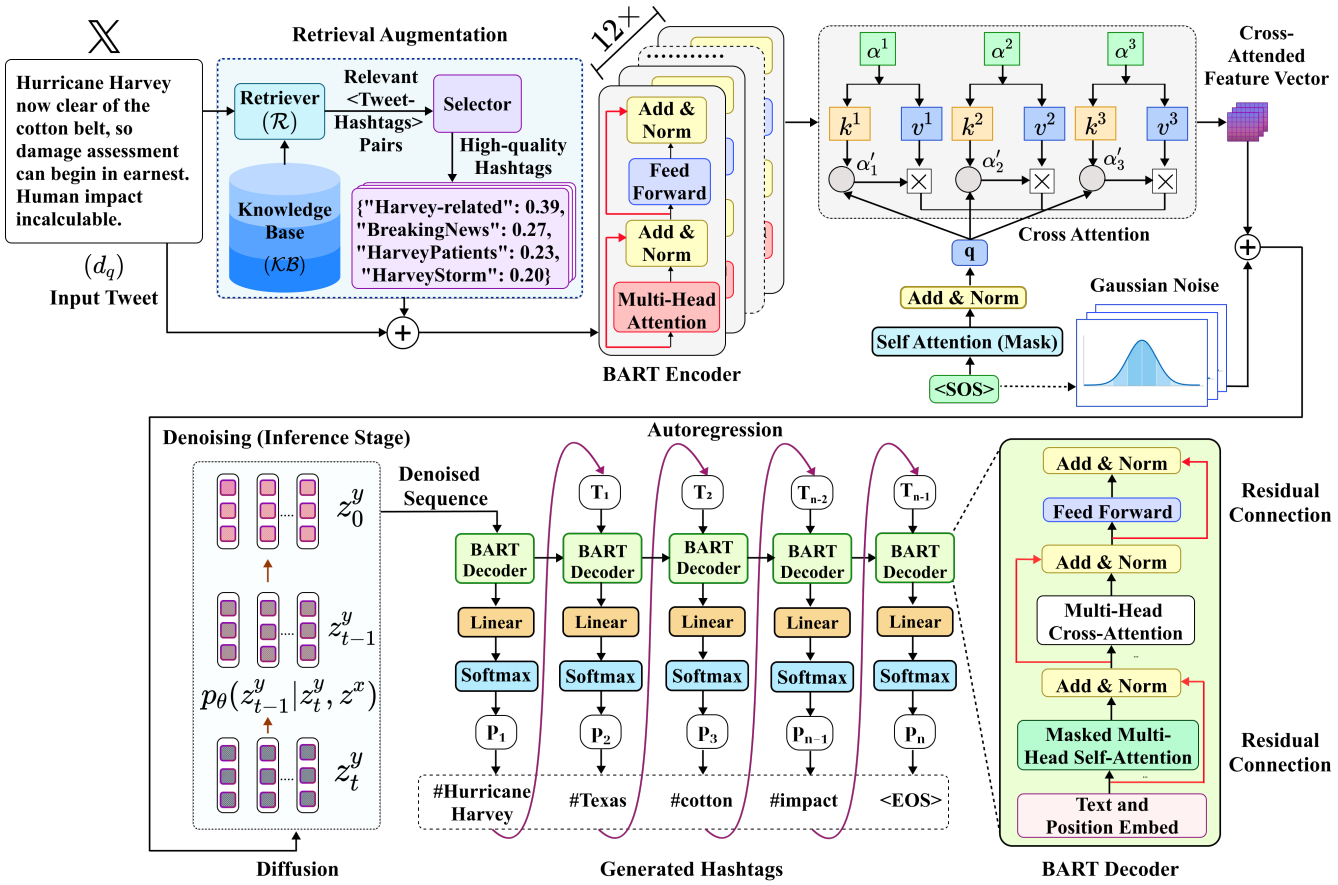


Figure 1: System Overview of ASSIGNER

Diffusion

This section describes the core diffusion model employed for hashtag generation.

Input Encoding Given an input tweet d_q and top-p hashtags $\{\tilde{h}_1, \ldots, \tilde{h}_p\}$ selected by the selector module, we concatenate these hashtags to the input tweet:

$$\tilde{d_q} = \tilde{h}_1 \oplus \cdots \oplus \tilde{h}_p \oplus d_q \tag{3}$$

where, $\oplus$ denotes the concatenation operation. This concatenated input $\tilde{d_q}$ is then fed into BART encoder to obtain its contextualized representation x_e.

$$x_e = \text{BART}_{\text{enc}}(\tilde{d_q}) \tag{4}$$

This embedding (x_e) captures information from relevant hashtags and the input tweet, providing richer context for the diffusion model.

Forward Diffusion Process Our approach involves a forward process that gradually introduces noise into the target output sequence $y_w{}_{i=1}^{l}$, where l represents its maximum length. This noise diffusion process is independent of the input sequence x_e. We first represent the output sequence y_w using an embedding function $f_\phi(.)$ which maps individual word tokens y_w^i to continuous embeddings $f_\phi(y_w^i) \in \mathbb{R}^m$, where m is the embedding dimension and ϕ represents parameters of $f(.)$. The overall output sequence embedding is obtained by concatenating individual token embeddings and is denoted as $f_\phi(y_w) \in \mathbb{R}^{l \times m}$. The forward process begins by applying a Markovian transition parameterized by $q_\phi(v_0|y_w) = N(v_0; f_\phi(y_w), \gamma_0 I)$ is added. The forward process is augmented by $q_\phi(v_0|y_w)$, which incrementally introduces diffusion to the continuous features of v_0. At each time step t, we apply the diffusion distribution $q(v_t|v_{t-1})$ to generate noisier samples. Finally, the original output sequence y_w is converted into v_T which closely resembles random noise drawn from a standard Gaussian distribution.

Reverse Process When reversing the noise injection, diffusion models synthesize data points by progressively drawing samples from the noise-reducing distribution p_θ parameterized by θ. This process transforms noisy samples v_t into progressively clearer samples v_{t-1}. In seq2seq setting, the noise reduction distribution depends on the input representation x_e (which is augmented with selected hashtag embeddings), represented as $p_\theta = p_\theta(v_{t-1}|v_t, x_e)$. When the reverse process reaches $T = 0$, the generated output $\hat{v}^0$ is mapped to its closest word in the embedding space. This mapping is achieved using a rounding distribution $\hat{p}_\phi(y_w|\hat{v}_0)$, ultimately producing the final sequence of hashtags.

Self-Conditioning Through a series of iterative refinements, the reverse process transforms a noisy representation into the final output sequence. At each iteration t, the function $v_\theta^0(v_t, x_e, t)$ takes the current noisy sample v_t and the input embedding x_e to predict a less noisy version of the output, gradually revealing the true sequence. This process inherently discards some information from the previous prediction $\hat{v}_t^0$. To address this information loss, Bit-Diffusion (Chen, Zhang, and Hinton 2022) introduced a self-conditioning technique that incorporates previous sequence predictions as additional input to the denoising function, formulating it as $v_0^\theta(v_t, \hat{v}_t^0, x_e, t)$. Self-conditioning allows the denoising function to refine previous sequence predictions instead of entirely generating new ones. A study (Strudel et al. 2022) has shown that self-conditioning enables text diffusion models to perform better. To integrate self-conditioning technique, we concatenate sequence features $\hat{v}_0^t$ from previous predictions with noisier sequence features v_t, increasing the decoder input dimension to $l \times 2m$. To improve training efficiency, we adopt a strategy where, with 70% probability, $v_\theta^0(v_t, \hat{v}_0^t, x_e, t)$ is trained with the input $\hat{v}_0^t$ set to 0. Alternatively, $\hat{v_0^t}$ is initially approximated using $v_\theta^0(v_t, 0, x_e, t)$, and this estimate is then employed for self-conditioning during training, thus bypassing the need for backpropagation through initial forward pass.

Denoising with Encoder-Decoder Framework For $v_\theta^0(v_t, x_e, t)$, we use the encoder for modeling the input sequence x_e and the decoder for handling the noisy output sequence v_t, augmented with time step embeddings. This encoder-decoder structure provides computational efficiency during generation by allowing the encoder to process the input sequence x_e just a single time during the entire reverse procedure, which may require numerous iterations to produce high-quality output. Our denoising function (v_θ^0) produces samples at the sequence level throughout both training and generation phases, consistent with non-autoregressive approaches to natural language generation. The decoder utilizes an attention mechanism that can attend to all positions within the output sequence at once. This differs from causal attention, which is restricted to attending only to preceding positions. By having access to the full context of the output sequence, the decoder can generate more informed predictions.

Adaptive Sigmoid-based Non-Linear Noise Scheduler We put forward a novel approach for adjusting noise non-linearly at the token level during training in diffusion models. This dynamic adjustment modulates the difficulty of the denoising process for the predicted output sequence, focusing on challenging tokens and thereby improving overall performance. Here, v_θ^0 represents the predicted output sequence at timestep 0 so that it increases sigmoidally with respect to timestep t. This aims to create a smooth progression of difficulty in denoising, making it easier at the start and end, facilitating initial stability and fine-grained refinement, respectively. This refined control over the noise injection process helps in generating high-quality hashtags. Recognizing that different token positions within a sequence hold varying levels of semantic and syntactic importance, we propose individual noise schedules for each token. This is motivated by the observation that inherent properties of token embeddings vary significantly across different positions.

We estimate the complexity of denoising process by examining the training loss at each timestep (t) and token position (i).

$$\mathcal{L}_t^i = \mathbb{E}q_\phi(x_e, y_w, v_t, v_0) \left| v_0^\theta(v_t, \hat{v}_t^0, x_e, t)^i - v_0^i \right|^2 \quad (5)$$

We utilize $\beta_t^i \in [0, 1]$ to regulate the noise intensity at each step. β_t^i variable meticulously determines the noise level at each time step t for each token position i. To achieve an adaptive noise schedule for each token position i, we employ a mapping $\beta^i = \Phi_i(\mathcal{L}^i)$, which connects $\mathcal{L}_t^i$ and β_t^i using a sigmoid function.

$$\beta_t^i = \Phi_i(\mathcal{L}_t^i) = \frac{1}{1 + \exp(-a_i(\mathcal{L}_t^i - b_i))} \quad (6)$$

where, $^*\beta_t^i$ is the noise level at time step t for token position i, $^*\mathcal{L}_t^i$ is the denoising loss at time step t for token position i. $^*a_i$ and b_i are learnable parameters that control the shape of the sigmoid for token position i. This function provides a smooth and flexible way to modify the noise intensity according to the observed complexity of denoising at each token position. We begin by initializing a noise schedule (using a standard cosine schedule) and tracking the loss, $\mathcal{L}_t^i$, at each step. This data is then used to establish the mapping function, Φ_i, which is updated periodically throughout training. In an ideal scenario, the training loss would consistently increase with each time step (t). This is because a larger t indicates a higher level of noise in the input characteristics (v_t) provided to denoising function. Nonetheless, given that total number of time steps (T) is typically very large, we end up with a highly detailed discretization of β^i. This fine granularity, combined with variations in empirical loss estimation, can lead to inconsistencies where the training loss does not strictly increase with each successive time step.

To address this and achieve a smoother mapping function (Φ_i), we employ a coarser discretization (s) for both β^i and $\mathcal{L}^i$. This strategy helps to smooth out minor fluctuations in the observed loss and ensures a more stable and reliable mapping.

$$\mathcal{L}_s^i = \frac{1}{K} \sum_{t=s\times K}^{s\times(K+1)} \mathcal{L}_t^i, \beta_s^i \quad = \frac{1}{K} \sum_{t=s\times K}^{s\times(K+1)} \beta_t^i, s = \left\lfloor \frac{t}{K} \right\rfloor \quad (7)$$

where K represents the stride to uniformly downsample t and $\lfloor \cdot \rfloor$ denotes the floor function. Using the learned sigmoid mapping $\beta_s^i = \Phi_i(\mathcal{L}_s^i)$, we can derive an updated discretized noise schedule $\beta_t^{i,new}$ by $\beta_t^{i,new} = \Phi_i(\mathcal{L}_t^{i,new})$ where $\mathcal{L}_t^{i,new}$'s are evenly taken ranging from the minimum to maximum recorded values. Throughout the training process, we dynamically adjust β^i by repeating this procedure with each training update. This iterative process ensures that the noise schedule remains aligned with the evolving complexity of the denoising task. Algorithm 1 presents the

Algorithm 1: Adaptive Sigmoid Noise Schedule

Input: Losses $\mathcal{L}_t^i$ and noise schedules β_t^i accumulated over each diffusion iteration t and sequence index i.
if Step counter % Update interval == 0 **then**
  for each sequence index i **do**
    Fit the sigmoid function $\Phi_i(\mathcal{L}_t^i) = \frac{1}{1+\exp(-a_i(\mathcal{L}_t^i - b_i))}$ by minimizing the error between β_t^i and $\Phi_i(\mathcal{L}_t^i)$, updating parameters a_i and b_i.
    Generate new loss values $\mathcal{L}_t^{i,\text{new}}$ sampled at uniform intervals between the minimum and maximum observed losses, $\min_t(\mathcal{L}_t^i)$ and $\max_t(\mathcal{L}_t^i)$.
    Compute the updated noise schedule $\beta_t^{i,\text{new}} = \Phi_i(\mathcal{L}_t^{i,\text{new}})$.
  end for
end if
Return: Noise schedule $\beta_t^{i,\text{new}}$ for each diffusion iteration t and sequence index i.

pseudo-code for configuring adaptive sigmoid noise schedule during the training process. The learnable parameters (a_i and b_i) allow the sigmoid to adapt to different loss distributions and token positions.

Training Objective The model parameters θ and ϕ are learned through a variational approximation for the data likelihood to reduce the difference between the learned denoising distribution $p_\theta(v_{t-1}|v_t, x_e)$ and the true posterior distribution $q(v_{t-1}|v_t, v_0)$ from the forward process.

$$\mathcal{L}_{VB} = \mathbb{E}_q[\log \frac{q(v_T|v_0)p(v_T)}{p_\theta(v_0|v_1, x_e)} + \sum_{t=2}^{T} \log \frac{q(v_{t-1}|v_0, v_t)p_\theta(v_{t-1}|v_t, x_e)}{p_\theta(v_0|v_1, x_e)} + \log \frac{q_\phi(v_0|y_w)}{\tilde{p}_\phi(y_w|v_0)}] \quad (8)$$

Since $q(v_{t-1}|v_t, v_0)$ has a Gaussian distribution, we parameterize the denoising distribution inside the Gaussian distribution family $p_\theta(v_{t-1}|v_t, x_e) = \mathcal{N}(v_{t-1}; \tilde{\mu}_\theta(v_t, x_e, t), \tilde{\gamma}_t \mathbf{I})$ where

$$\tilde{\mu}_\theta(v_t, x_e, t) = \sqrt{\frac{\bar{\beta}_{t-1}\gamma_t}{1-\bar{\beta}_t}} v_0^\theta(v_t, x_e, t) + \sqrt{\frac{\beta_t(1-\bar{\beta}_{t-1})}{1-\bar{\beta}_t}} v_t \quad (9)$$

$v_\theta^0(v_t, x_e, t)$ denotes the function that predicts the output representation at each iteration of the reverse pass. Under the Gaussian noise assumption, the objective can be expressed more concisely as:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{q_\phi(v_0, x_e, y_w)}\Bigg[\sum_{t=2}^{T} \mathbb{E}_{q(v_t|v_0)} \left\|v_0^\theta(v_t, x_e, t) - v_0\right\|^2 + \|\tilde{\mu}(v_T, v_0)\|^2 + \left\|v_0^\theta(v_1, x_e, 1) - f_\phi(y_w)\right\|^2 - \log \tilde{p}_\phi(y_w|v_0)\Bigg] \quad (10)$$

where, $q(v_t|v_0) = \mathcal{N}\left(v_t; \sqrt{\bar{\beta}_t}v_0, (1-\beta_t)\mathbf{I}\right)$ for efficient sampling of v_t during training, and $\mu_T(v_0) = \sqrt{\bar{\beta}_T}v_0$ and the denoising function $v_\theta^0(v_t, x_e, t)$, which is modeled using a transformer network with separate components for encoding the input and generating the output. During training, the distribution used for drawing samples q_ϕ includes learnable parameters from token representation model. We utilize the reparameterization trick (Kingma 2013) to enable backpropagation through these parameters.

Inference Given an input tweet d_q and the output from the retriever $\{(d_1, H_1, s(d_q, d_1)), \ldots, (d_N, H_N, s(d_q, d_N))\}$, the selector aggregates retrieved hashtags into a set $\{h_1, \ldots, h_M\}$, where M is the number of unique hashtags. For each hashtag h_m, the selector C computes relevance score between the input tweet and each unique hashtag.

$$r(d_q, h_m) = (C)(d_q, h_m) \quad (11)$$

Here, $r(d_q, h_m)$ denotes relevance score between the input tweet d_q and hashtag h_m, as computed by the selector C. Lastly, we calculate the average similarity between tweets for each hashtag and incorporate the semantic relatedness between the tweet and the hashtag.

$$\rho_i = \left(\frac{1}{n_i}\sum_{j=1}^{n_i} s(d_q, d_j)\right) + r(d_q, h_i) \quad (12)$$

Here, n_i is the frequency of hashtag h_i in retrieved tweets, $s(d_q, d_j)$ denotes the similarity score between d_q and j^{th} retrieved tweet containing h_i, and h_i is obtained from the retriever. We then rank hashtags in descending order according to final scores ρ_i and select top-p hashtags $\{\tilde{h}_1, \ldots, \tilde{h}_p\}$. Since we do not have ground-truth hashtags for the test tweet, only the reverse step of the diffusion process is performed. Starting from random noise v_T, the model iteratively denoises samples to obtain v_0. This denoised output is then passed through the rounding distribution to obtain the predicted hashtag sequence:

$$\hat{y}_w = \hat{p}_\phi(y_w|v_0) \quad (13)$$

Finally, the decoder of BART model generates final hashtag recommendations based on predicted sequence $\hat{y}_w$ and the input embedding x_e.

$$\text{Hashtags} = \text{BART}_{\text{dec}}(\hat{y}_w, x_e) \quad (14)$$

Experimental Evaluations

This section details the experimental configurations followed by a presentation and analysis of results obtained.

Experimental Setup

Dataset This study uses a disaster-related tweet dataset, originally presented by (Chowdhury, Caragea, and Caragea 2020), to investigate hashtag recommendation. The dataset contains tweets during Harvey, Irma, Maria, Mexico earthquake, Chiapas earthquake, and California wildfire crawled using Twitter streaming API and tweets sourced from publicly available datasets (Imran, Mitra, and Castillo 2016; Olteanu et al. 2014; Alam, Ofli, and Imran 2018) To ensure data quality, (Chowdhury, Caragea, and Caragea 2020) implemented a rigorous filtering process removing uninformative, non-English, and duplicate tweets that contained a total of 67,288 tweets spanning a total of 37 types of disasters. We further removed tweets with invalid links and taken down from X. The final dataset used in our study contains 26,665 tweets, 12,230 unique hashtags with an average of 2 hashtags per tweet as depicted in Table 1. The dataset and code for ASSIGNER has been made publicly available[1].

#Tweets	# Hashtags	#Avg. h_t	#Max. h_t	#Min. h_t
26,665	12,230	2	23	1

Table 1: Statistics of the dataset. Here, h_t denotes number of hashtags per tweet.

Compared Methods The performance of our proposed model is evaluated against extant hashtag recommendation methods. These include sequence generation models such as AMNN (2020), SEGTRM (Mao et al. 2022), and HashTation (Diao et al. 2023); keyphrase extraction methods such as LSTM-MTL (Chowdhury, Caragea, and Caragea 2020); retrieval-augmented generation methods such as RIGHT (Fan et al. 2024); and diffusion models including Diffuseq (Gong et al. 2022) and SeqDiffuSeq (Yuan et al. 2024).

Evaluation Metrics To evaluate the quality and diversity of generated hashtags, we employed four metrics namely, BERTScore, dist.1, ROUGE-1, and BLEU. BERTScore (Zhang et al. 2019) assesses the semantic similarity with ground-truth hashtags, ROUGE-1 (Lin 2004) quantifies unigram overlap, BLEU (Papineni et al. 2002) determines the precision of generated hashtag sequences, and distinct unigram (dist. 1) measures the diversity of words within generated sequences. Higher scores of dist. 1 indicate less repetition. We utilize the official ROUGE script[2] (version 0.3.1) for calculating ROUGE scores.

Implementation Details The proposed model, ASSIGNER was implemented using the PyTorch framework. It employs a 6-layered encoder-decoder transformer with Gaussian Error Linear Units (GeLU) activation functions. The model utilizes a diffusion-based approach with 200 iterative steps, where noise is incrementally added and then removed. We employed AdamW optimizer with a learning rate of 10e-4, incorporating a warm-up period of 500 steps followed by a linear decay. The model was trained for 15 epochs with a batch size of 64. The dataset was split into 75% for training, 15% for validation, and 10% for testing, with tweets truncated to a maximum length of 128 tokens. A threshold of 0.7 was used in the selector module. All hyperparameters were optimized based on the validation data. To ensure reproducibility, we set a random seed of 101. To account for potential variability in performance, each experiment was repeated five times, and results are reported as the average of evaluation metrics, with the standard error of the mean falling within a range of 0.01 to 0.02 for BLEU, ROUGE-1, dist. 1, and BERTScore.

[1]https://github.com/abcd3007/ASSIGNER

[2]https://pypi.org/project/pyrouge/

Experimental Results

To analyze our proposed model, we conducted quantitative analysis, ablation studies, and qualitative case studies detailed below.

Methods	BERTScore	dist. 1	ROUGE-1	BLEU
Sequence Generation				
AMNN	0.359	<u>0.892</u>	0.001	0.002
SEGTRM	0.255	0.777	0.001	0.002
HashTation	0.246	0.514	0.001	0.001
Keyphrase Extraction				
LSTM-MTL	0.355	0.700	0.001	0.001
Retrieval-Augmented Generation				
RIGHT	<u>0.389</u>	0.877	0.003	0.001
Diffusion				
Diffuseq	0.344	0.799	**0.014**	**0.012**
SeqDiffuseq	0.235	0.522	0.003	0.001
ASSIGNER	**0.458**	**0.987**	<u>0.008</u>	<u>0.011</u>

Table 2: Effectiveness comparison results of ASSIGNER with existing methods for hashtag recommendation (top-2). The best result is highlighted in bold, while the second-best is underlined.

Quantitative Analysis Table 2 demonstrates that ASSIGNER significantly outperforms established methods across all assessment criteria. ASSIGNER outperforms AMNN by incorporating a retrieval mechanism to focus on relevant candidate hashtags and diffusion-based encoder-decoder architecture (BART) to capture linguistic characteristics of disaster-related tweets. Unlike AMNN, which relies on RNN-based encoder-decoder (BiLSTM-GRU) with softmax layer and produces generic hashtags, ASSIGNER leverages the strength of transformer and diffusion model in capturing complex data distributions, allowing it to grasp the dynamic nature of disaster-related language, leading to accurate hashtag recommendation. While SEGTRM uses segment selection to identify important parts of text, ASSIGNER's retrieval and selector components provide a more focused set of candidate hashtags. Additionally, the diffusion-based generator in ASSIGNER generates diverse and creative hashtags compared to SEGTRM's transformer

decoder. ASSIGNER surpasses LSTM-MTL by moving beyond keyphrase extraction and utilizing a generation-based approach. This allows ASSIGNER to generate novel hashtags not limited to those present in text, unlike LSTM-MTL, which extracts keyphrases directly. While RIGHT incorporates a retrieval mechanism, ASSIGNER further enhances this with a diffusion-based generator and adaptive sigmoid noise scheduling. This allows ASSIGNER to generate diverse and relevant hashtags compared to RIGHT, which relies on a standard generative model. ASSIGNER builds upon DiffuSeq and SeqDiffuSeq to improve hashtag generation. It incorporates an encoder-decoder architecture (BART), self-conditioning, and adaptive sigmoid noise scheduling for enhanced efficiency and performance compared to DiffuSeq's encoder-only framework. Additionally, ASSIGNER extends SeqDiffuSeq by adding a retrieval and selection mechanism to focus on relevant candidate hashtags besides adaptive sigmoid noise scheduling algorithm. This combined approach leads to more accurate and diverse hashtag generation.

Methods	BERTScore	dist. 1	ROUGE-1	BLEU
w/o Self-conditioning	0.145	0.310	0.003	0.003
w/o RAG	0.421	0.987	0.003	0.003
w/o Noise Scheduling	0.412	0.783	0.020	0.030
w/o Diffusion	0.409	0.844	0.002	0.002
w/o Selector	0.416	0.974	0.003	0.002
ASSIGNER	**0.458**	**0.987**	**0.008**	**0.011**

Table 3: Effect of individual component ablation on hashtag generation performance of ASSIGNER. Here, w/o refers to without.

Ablation Studies

- **w/o Self-conditioning**: Our ablation study highlights the crucial role of self-conditioning in ASSIGNER. Removing it drastically reduces performance across all metrics (BERTScore: 0.4581 to 0.1446, dist.1: 0.9872 to 0.3096, ROUGE-1: 0.0078 to 0.0028, BLEU: 0.0113 to 0.0029) as evident from Table 3. This decline is attributed to information loss inherent in standard diffusion process where each denoising step relies solely on the current noisy input, neglecting refined information from previous predictions. Self-conditioning mitigates this by incorporating the previous prediction into the denoising function, allowing the model to refine its estimations and generate contextually relevant hashtags that capture the evolving event-specific language prevalent in disaster-related tweets.
- **w/o RA**: To assess the impact of Retrieval Augmentation (RA) mechanism in ASSIGNER, we conducted an ablation study where RA was removed. We replaced the retriever (which selects relevant candidate hashtags), with a random selection of top-k hashtags from the training dataset. This modification resulted in a noticeable performance drop across all metrics (BERTScore: 0.458 to 0.421, ROUGE-1: 0.008 to 0.003, BLEU: 0.011 to 0.003) as evident in 3. This result highlights the crucial role of RA in providing the diffusion-based encoder-decoder framework with a focused set of relevant candidate hashtags. By leveraging information from similar tweets and their associated hashtags, RA effectively guides the generator towards more informative and accurate hashtag recommendations. These retrieved hashtags serve as guiding signals and a starting point for generating final hashtags, ultimately enhancing their quality and effectiveness.
- **w/o Noise Scheduling**: As shown in Table 3, removing the adaptive sigmoid noise scheduler from diffusion pipeline significantly hinders performance (BERTScore: drops from 0.458 from 0.412, dist. 1 from 0.9872 to 0.783). This underscores the importance of a well-designed noise scheduling for guiding the diffusion process towards meaningful outputs. By applying this scheduler at the token level, ASSIGNER achieves two key advantages namely, contextual adaptation, enabling the model to adjust noise based on each token's specific context, crucial in dynamic disaster situations, and enhanced learning that captures complex inter-token dependencies to recommend pertinent hashtags. This precise control over noise introduction and reduction empowers effective learning and generation of contextually relevant hashtags, supporting information dissemination during critical events.
- **w/o Diffusion** These ablation results underscore the significant contribution of the diffusion-based encoder-decoder framework to ASSIGNER's strong performance. Removing this component and replacing it with a standard encoder-decoder framework leads to a substantial drop in performance across all metrics (BERTScore: 0.458 to 0.409, dist.1: 0.987 to 0.844, ROUGE-1: 0.008 to 0.002, BLEU: 0.011 to 0.002) as can be seen in Table 3. This decline is attributed to limitations of standard encoder-decoder models in capturing the complex and dynamic language characteristic of disaster-related tweets. These models tend to produce generic hashtag recommendations due to their reliance on maximizing training data likelihood. In contrast, the diffusion-based generator in ASSIGNER leverages a gradual noising process, enabling it to explore a wider range of possibilities and generate diverse and informative hashtags. This approach is well-suited for capturing the evolving and informal language used in disaster situations, where new terms and expressions may emerge rapidly, leading to improved performance.
- **w/o Selector** In this ablated variant, the selector is omitted and we directly choose top-p retrieved hashtags. Removing the selector leads to a significant decrease in performance across all metrics. BERTScore drops from 0.458 to 0.416, dist.1 from 0.987 to 0.974, ROUGE-1 from 0.008 to 0.003, and BLEU from 0.011 to 0.002, as can be seen in 3. This decline highlights that the selector plays a vital role in refining candidate hashtags identified by the retriever. By analyzing how closely the input tweet matches the retrieved tweet and hashtags in terms

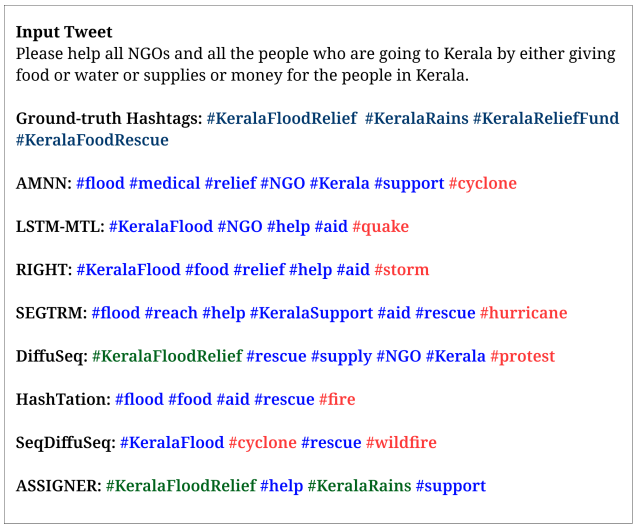


Figure 2: Example of a tweet from test dataset depicting hashtags recommended by various methods. Generated hashtags that match user-assigned hashtags are marked with green, while relevant but non-matching hashtags are marked with blue, and irrelevant predictions are marked with red.

of meaning, the selector ensures that only the most relevant hashtags are passed to the diffusion-based generator. This filtering step is essential, as simply relying on top-p hashtags from the retriever, based on similarity to the input tweet, proves insufficient for generating accurate and informative hashtag recommendations. The selector thus acts as a quality control mechanism, guiding the generator towards optimal hashtag selection and improving overall performance.

Qualitative Analysis Figure 2 presents a qualitative comparison of generated hashtags for an example tweet from the test dataset, alongside ground-truth hashtags and hashtags generated by various methods. As illustrated in Figure 2, ASSIGNER demonstrates a superior ability to suggest relevant hashtags for the given tweet. Notably, ASSIGNER is the only model that correctly identifies two ground-truth hashtags: #KeralaFloodRelief and #KeralaRains. While DiffuSeq also generates #KeralaFloodRelief, ASSIGNER is unique in its ability to simultaneously identify both of these crucial hashtags. Moreover, ASSIGNER effectively captures general terms such as #flood, #medical, and #relief and specific terms such as #Kerala and #NGO, which are relevant to the given tweet. In contrast, other methods exhibit varying degrees of success, but none achieve the same level of accuracy as ASSIGNER. HashTation primarily focuses on generic terms such as #flood, #food, and #aid, lacking the specificity of ASSIGNER. SeqDiffuSeq incorrectly predicts hashtags such as #cyclone and #wildfire, which are not relevant to Kerala floods. AMNN and LSTM-MTL incorrectly predict hashtags #cyclone #quake, respectively. These errors highlight the difficulty other methods have in capturing the specific context of the input tweet. The enhanced performance of ASSIGNER can be attributed to the effective integration of retrieval augmentation with a diffusion-based encoder-decoder framework. The retriever module provides valuable context to the diffusion-based generator by identifying similar tweets and their corresponding hashtags. The selector module further refines these retrieved hashtags, ensuring that only the most relevant candidates are passed to the generator. Furthermore, ASSIGNER's self-conditioning mechanism and adaptive sigmoid noise scheduler contribute to generating high-quality hashtag sequences by exploring a broader spectrum of possibilities. Overall, the qualitative analysis demonstrates ASSIGNER's ability in leveraging existing knowledge, capturing contextual information, and generating diverse hashtag sequences, significantly outperforming other methods.

Conclusion

This paper introduces ASSIGNER, a novel retrieval-augmented encoder-decoder with diffusion for sequential hashtag recommendation in disaster events. ASSIGNER extends continuous text diffusion model to generate hashtags sequentially and a retrieval mechanism that leverages existing knowledge from semantically similar tweets and hashtags. This approach addresses the limitations of existing methods by capturing both semantic relationships among hashtags and contextual information embedded in tweets. Furthermore, a novel adaptive sigmoid noise scheduler is proposed to improve the quality of generated hashtags. Experimental results validate the capability of ASSIGNER in generating relevant and informative hashtags for disaster-related tweets, with the potential to improve information dissemination and response efforts during crises.

References

Alam, F.; Ofli, F.; and Imran, M. 2018. Crisismmd: Multimodal twitter datasets from natural disasters. In *Proceedings of the international AAAI conference on web and social media*, volume 12.

Asai, A.; Min, S.; Zhong, Z.; and Chen, D. 2023. Retrieval-based language models and applications. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 6: Tutorial Abstracts)*, 41–46.

Austin, E.; Zaïane, O. R.; and Largeron, C. 2022. Community topic: topic model inference by consecutive word community discovery. In *Proceedings of the 29th International Conference on Computational Linguistics*, 971–983.

Austin, J.; Johnson, D. D.; Ho, J.; Tarlow, D.; and Van Den Berg, R. 2021. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34: 17981–17993.

Bansal, S.; Gowda, K.; and Kumar, N. 2022. A hybrid deep neural network for multimodal personalized hashtag recommendation. *IEEE transactions on computational social systems*, 10(5): 2439–2459.

Bansal, S.; Gowda, K.; and Kumar, N. 2024. Multilingual personalized hashtag recommendation for low resource Indic languages using graph-based deep neural network. *Expert Systems with Applications*, 236: 121188.

Bansal, S.; Gowda, K.; Rehman, M. Z. U.; Raghaw, C. S.; and Kumar, N. 2024. A hybrid filtering for micro-video hashtag recommendation using graph-based deep neural network. *Engineering Applications of Artificial Intelligence*, 138: 109417.

Cao, D.; Miao, L.; Rong, H.; Qin, Z.; and Nie, L. 2020. Hashtag our stories: Hashtag recommendation for micro-videos via harnessing multiple modalities. *Knowledge-Based Systems*, 203: 106114.

Chen, J.; Zhang, R.; Guo, J.; de Rijke, M.; Chen, W.; Fan, Y.; and Cheng, X. 2023. Continual learning for generative retrieval over dynamic corpora. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 306–315.

Chen, T.; Zhang, R.; and Hinton, G. 2022. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*.

Chowdhury, J. R.; Caragea, C.; and Caragea, D. 2020. On identifying hashtags in disaster twitter data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 498–506.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186.

Diao, S.; Keh, S. S.; Pan, L.; Tian, Z.; Song, Y.; and Zhang, T. 2023. Hashtag-Guided Low-Resource Tweet Classification. In *Proceedings of the ACM Web Conference 2023*, 1415–1426.

Fan, R.-Z.; Fan, Y.; Chen, J.; Guo, J.; Zhang, R.; and Cheng, X. 2024. RIGHT: Retrieval-Augmented Generation for Mainstream Hashtag Recommendation. In *European Conference on Information Retrieval*, 39–55. Springer.

FORCE11. 2020. The FAIR Data principles. https://force11.org/info/the-fair-data-principles/.

Frej, W. 2018. Hurricane florence flood victims turn to social media for rescue. *HuffPost, 2018-09-14, Available at*.

Gao, Y.; Yin, Q.; Li, Z.; Meng, R.; Zhao, T.; Yin, B.; King, I.; and Lyu, M. 2022. Retrieval-Augmented Multilingual Keyphrase Generation with Retriever-Generator Iterative Training. In *Findings of the Association for Computational Linguistics: NAACL 2022*, 1233–1246.

Gebru, T.; Morgenstern, J.; Vecchione, B.; Vaughan, J. W.; Wallach, H.; Iii, H. D.; and Crawford, K. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92.

Gong, S.; Li, M.; Feng, J.; Wu, Z.; and Kong, L. 2022. Diffuseq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*.

Gong, Y.; Zhang, Q.; and Huang, X.-J. 2015. Hashtag recommendation using dirichlet process mixture models incorporating types of hashtags. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 401–410.

Gu, J.; Wang, Y.; Cho, K.; and Li, V. O. 2018. Search engine guided neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Guu, K.; Lee, K.; Tung, Z.; Pasupat, P.; and Chang, M. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, 3929–3938. PMLR.

He, H.; Zhang, H.; and Roth, D. 2022. Rethinking with retrieval: Faithful large language model inference. *arXiv preprint arXiv:2301.00303*.

He, Z.; Sun, T.; Tang, Q.; Wang, K.; Huang, X.-J.; and Qiu, X. 2023. DiffusionBERT: Improving Generative Masked Language Models with Diffusion Models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 4521–4534.

Ho, J.; Salimans, T.; Gritsenko, A.; Chan, W.; Norouzi, M.; and Fleet, D. J. 2022. Video diffusion models. *Advances in Neural Information Processing Systems*, 35: 8633–8646.

Hoogeboom, E.; Nielsen, D.; Jaini, P.; Forré, P.; and Welling, M. 2021. Argmax flows and multinomial diffusion: Learning categorical distributions. *Advances in Neural Information Processing Systems*, 34: 12454–12465.

Hossain, N.; Ghazvininejad, M.; and Zettlemoyer, L. 2020. Simple and effective retrieve-edit-rerank text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2532–2538.

Imran, M.; Mitra, P.; and Castillo, C. 2016. Twitter as a Lifeline: Humanannotated Twitter Corpora for NLP of Crisis-related Messages.

Kim, J.; Jeong, M.; Choi, S.; and Hwang, S.-w. 2021. Structure-augmented keyphrase generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2657–2667.

Kingma, D. P. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kocoń, J.; Cichecki, I.; Kaszyca, O.; Kochanek, M.; Szydło, D.; Baran, J.; Bielaniewicz, J.; Gruza, M.; Janz, A.; Kanclerz, K.; et al. 2023. ChatGPT: Jack of all trades, master of none. *Information Fusion*, 99: 101861.

Kong, Z.; Ping, W.; Huang, J.; Zhao, K.; and Catanzaro, B. 2020. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761*.

Lewis, M. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474.

Li, J.; Sun, S.; Yuan, W.; Fan, R.-Z.; Zhao, H.; and Liu, P. 2023a. Generative judge for evaluating alignment. *arXiv preprint arXiv:2310.05470*.

Li, J.; Xu, H.; He, X.; Deng, J.; and Sun, X. 2016. Tweet modeling with LSTM recurrent neural networks for hashtag

recommendation. In *2016 International Joint Conference on Neural Networks (IJCNN)*, 1570–1577. IEEE.

Li, X.; Chan, S.; Zhu, X.; Pei, Y.; Ma, Z.; Liu, X.; and Shah, S. 2023b. Are ChatGPT and GPT-4 General-Purpose Solvers for Financial Text Analytics? A Study on Several Typical Tasks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 408–422.

Li, X.; Thickstun, J.; Gulrajani, I.; Liang, P. S.; and Hashimoto, T. B. 2022. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35: 4328–4343.

Lian, R.; Xie, M.; Wang, F.; Peng, J.; and Wu, H. 2019. Learning to Select Knowledge for Response Generation in Dialog Systems. In *IJCAI International Joint Conference on Artificial Intelligence*, 5081.

Lin, C.-Y. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, 74–81.

Mao, Q.; Li, X.; Liu, B.; Guo, S.; Hao, P.; Li, J.; and Wang, L. 2022. Attend and select: A segment selective transformer for microblog hashtag generation. *Knowledge-Based Systems*, 254: 109581.

Marujo, L.; Ling, W.; Trancoso, I.; Dyer, C.; Black, A. W.; Gershman, A.; de Matos, D. M.; Neto, J. P.; and Carbonell, J. G. 2015. Automatic keyword extraction on twitter. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 637–643.

Olteanu, A.; Castillo, C.; Diaz, F.; and Vieweg, S. 2014. Crisislex: A lexicon for collecting and filtering microblogged communications in crises. In *Proceedings of the international AAAI conference on web and social media*, volume 8, 376–385.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 311–318.

Ramos, R.; Elliott, D.; and Martins, B. 2023. Retrieval-augmented Image Captioning. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, 3666–3681.

Raunak, V.; Menezes, A.; and Junczys-Dowmunt, M. 2021. The Curious Case of Hallucinations in Neural Machine Translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1172–1183.

Rhodan, M. 2017. 'please send help.'hurricane harvey victims turn to twitter and facebook. *Time, August*, 30.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Silverman, L. 2017. Facebook, twitter replace 911 calls for stranded in houston. *National Public Radio*.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2021. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*.

Strudel, R.; Tallec, C.; Altché, F.; Du, Y.; Ganin, Y.; Mensch, A.; Grathwohl, W.; Savinov, N.; Dieleman, S.; Sifre, L.; et al. 2022. Self-conditioned embedding diffusion for text generation. *arXiv preprint arXiv:2211.04236*.

Villegas, C.; Martinez, M.; and Krause, M. 2018. Lessons from harvey: Crisis informatics for urban resilience.

Wang, Y.; Li, J.; King, I.; Lyu, M. R.; and Shi, S. 2019. Microblog Hashtag Generation via Encoding Conversation Contexts. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1624–1633.

Wei, Y.; Cheng, Z.; Yu, X.; Zhao, Z.; Zhu, L.; and Nie, L. 2019. Personalized hashtag recommendation for micro-videos. In *Proceedings of the 27th ACM International Conference on Multimedia*, 1446–1454.

Yang, Q.; Wu, G.; Li, Y.; Li, R.; Gu, X.; Deng, H.; and Wu, J. 2020. AMNN: Attention-based multimodal neural network model for hashtag recommendation. *IEEE Transactions on Computational Social Systems*, 7(3): 768–779.

Yu, P.; Xie, S.; Ma, X.; Jia, B.; Pang, B.; Gao, R.; Zhu, Y.; Zhu, S.-C.; and Wu, Y. N. 2022. Latent Diffusion Energy-Based Model for Interpretable Text Modelling. In *International Conference on Machine Learning*, 25702–25720. PMLR.

Yuan, H.; Yuan, Z.; Tan, C.; Huang, F.; and Huang, S. 2024. Text Diffusion Model with Encoder-Decoder Transformers for Sequence-to-Sequence Generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 22–39.

Zangerle, E.; Gassler, W.; and Specht, G. 2011. Recommending#-tags in twitter. In *Proceedings of the workshop on semantic adaptive social web (SASWeb 2011). CEUR workshop proceedings*, volume 730, 67–78.

Zhang, Q.; Wang, Y.; Gong, Y.; and Huang, X.-J. 2016. Keyphrase extraction using deep recurrent neural networks on twitter. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, 836–845.

Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K. Q.; and Artzi, Y. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Zhang, Y.; Li, J.; Song, Y.; and Zhang, C. 2018. Encoding conversation context for neural keyphrase extraction from microblog posts. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1676–1686.

Zheng, X.; Mekala, D.; Gupta, A.; and Shang, J. 2021. News meets microblog: Hashtag annotation via retriever-generator. *arXiv preprint arXiv:2104.08723*.

Paper Checklist

1. For most authors...
   (a) Would answering this research question advance science without violating social contracts, such as violating privacy norms, perpetuating unfair profiling, exacerbating the socio-economic divide, or implying disrespect to societies or cultures? Yes, our research aims to leverage AI for social good by enhancing communication and information access during disaster events. This contributes to a deeper understanding of how social media can be used for effective disaster response, potentially aiding in rescue efforts, resource allocation, and community support. Our approach respects privacy by not relying on personally identifiable information and promotes ethical considerations by actively mitigating potential biases in the data and algorithms. By improving information dissemination and facilitating a more equitable response to disasters, this research aligns with societal values and promotes the well-being of individuals and communities.
   (b) Do your main claims in the abstract and introduction accurately reflect the paper's contributions and scope? Yes, please see Abstract and Introduction.
   (c) Do you clarify how the proposed methodological approach is appropriate for the claims made? Yes, please refer to Experimental Results subsection and Appendix.
   (d) Do you clarify what are possible artifacts in the data used, given population-specific distributions? No, we do not have any artifacts in the data employed for this study.
   (e) Did you describe the limitations of your work? Yes, please see the Appendix.
   (f) Did you discuss any potential negative societal impacts of your work? Yes, please see the Appendix.
   (g) Did you discuss any potential misuse of your work? NA
   (h) Did you describe steps taken to prevent or mitigate potential negative outcomes of the research, such as data and model documentation, data anonymization, responsible release, access control, and the reproducibility of findings? Yes, please see Future work in Appendix that discusses preventive steps to be adopted for mitigating any potential biases. For reproducibility, we have made the code and data publicly available and provided the implementaion details (see Experimental Setup).
   (i) Have you read the ethics review guidelines and ensured that your paper conforms to them? Yes, we have ensured that the paper conforms to the mentioned guidelines.
2. Additionally, if your study involves hypotheses testing...
   (a) Did you clearly state the assumptions underlying all theoretical results? Yes, we have stated assumptions wherever necessary.
   (b) Have you provided justifications for all theoretical results? Yes, please see Experimental Evaluations and Appendix.
   (c) Did you discuss competing hypotheses or theories that might challenge or complement your theoretical results? We have compared the performance of our model (ASSIGNER) against several baseline models. This implies an underlying hypothesis that our model is superior to the existing methods.
   (d) Have you considered alternative mechanisms or explanations that might account for the same outcomes observed in your study? We have explored the impact of different hyperparameters (the number of diffusion steps, the probability parameter for self-conditioning) on the model's performance. This implies a hypothesis that certain hyperparameter values will lead to better performance than others. The analysis of the results of the parameter sensitivity study can also be viewed as hypothesis testing, where we are testing the hypothesis that different parameter values will lead to significantly different performance outcomes. Please see Appendix for details.
   (e) Did you address potential biases or limitations in your theoretical framework? Yes, please refer to Discussion section in the Appendix.
   (f) Have you related your theoretical results to the existing literature in social science? Yes, we have related our theoretical results to the existing literature in social science. A detailed discussion of these connections can be found in Introduction and the Related Works section of our paper. We discuss how our work builds upon these existing studies and contributes to a deeper understanding of the social and behavioral aspects of hashtag usage.
   (g) Did you discuss the implications of your theoretical results for policy, practice, or further research in the social science domain? Yes, please refer to theoretical and practical implications under the Discussion section in the Appendix.
3. Additionally, if you are including theoretical proofs...
   (a) Did you state the full set of assumptions of all theoretical results? NA
   (b) Did you include complete proofs of all theoretical results? NA
4. Additionally, if you ran machine learning experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? The url for the code and dataset has been given under the Dataset in Experimental Evaluations Section. Alternatively, please see: https://github.com/abcd3007/ASSIGNER
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? Yes, please see Implementation Details in Experimental Evaluations section for training details and Parameter Sensitivity Study in Appendix for hyperparameters.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? Yes, please see Implementation Details in Experimental Evaluations section

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? Yes, please see Implementation Details in Experimental Evaluations section.

(e) Do you justify how the proposed evaluation is sufficient and appropriate to the claims made? We employ the evaluation metrics widely used for generation tasks (see Appendix). Further, we conduct ablation studies to validate the importance of each novel component of the proposed framework (see Ablation Studies in Experimental Results Section).

(f) Do you discuss what is "the cost" of misclassification and fault (in)tolerance? Yes, please see Implementation Details in Experimental Evaluations section.

5. Additionally, if you are using existing assets (e.g., code, data, models) or curating/releasing new assets, **without compromising anonymity**...

(a) If your work uses existing assets, did you cite the creators? We have given proper credit to original creators of any technique and datasets used with proper citations throughout the paper.

(b) Did you mention the license of the assets? Yes, please see Ethics Statement in Appendix

(c) Did you include any new assets in the supplemental material or as a URL? Yes, the supplemental material includes details of evaluation metrics, compared methods, study on performance of ASSIGNER with different noise schedulers (Table 4), parameter sensitivity studies (Tables 5,6,7, and 8) followed by discussion on merits and demerits of the proposed framework.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? Yes, the dataset used in this study was made publicly available by the authors: https://github.com/JRC1995/Tweet-Disaster-Keyphrase. As the data consists of public posts on social media platforms, it is generally considered that users have provided implied consent for their posts to be used in research and analysis. However, we acknowledge the ethical considerations surrounding the use of social media data and have taken steps to ensure responsible use, such as: Anonymization: We have removed any personally identifiable information from the dataset to protect user privacy. Data usage: The data is solely utilized for academic research purposes and will not be used for any commercial applications. Transparency: We clearly acknowledge the source of the dataset and provide access to the preprocessed data for reproducibility.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? No, the dataset comprises publicly available information, no personally identifiable or sensitive data was collected or used.

(f) If you are curating or releasing new datasets, did you discuss how you intend to make your datasets FAIR (see FORCE11 (2020))? NA

(g) If you are curating or releasing new datasets, did you create a Datasheet for the Dataset (see Gebru et al. (2021))? NA

6. Additionally, if you used crowdsourcing or conducted research with human subjects, **without compromising anonymity**...

(a) Did you include the full text of instructions given to participants and screenshots? NA

(b) Did you describe any potential participant risks, with mentions of Institutional Review Board (IRB) approvals? NA

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? NA

(d) Did you discuss how data is stored, shared, and deidentified? NA

Appendix

Evaluation Metrics

- BERTScore: BERTScore leverages pre-trained contextual embeddings from Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al. 2019) to assess the semantic similarity between generated and reference hashtag sequences. It computes a similarity score by comparing contextualized representations of corresponding tokens in both sequences.

$$BERTScore(G, R) = \frac{1}{|G|} \sum_{g \in G} \max_{r \in R} \cos(c, r) \tag{15}$$

  Here, G represents the generated hashtag sequence, R denotes the reference hashtag sequence, g and r are contextualized embeddings of each hashtag in G and R, respectively, and $cos(g, r)$ denotes the cosine similarity between embeddings g and r.

- Distance-1 (dist. 1): This metric assesses the similarity between two sequences based on the minimum number of edits required to make them identical.

$$Distance{-}1(G, R) = \text{minimum number of edits}(G \rightarrow R) \tag{16}$$

  High values of Distance-1 implies high diversity.

- BLEU: Bilingual Evaluation Understudy (BLEU) measures the precision of n-gram matches between generated and reference hashtag sequences. It calculates the overlap of n-grams of varying lengths (typically 1 to 4) and combines them with a brevity penalty to discourage overly short generations.

$$BLEU(G, R) = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{17}$$

  Here, BP represents the brevity penalty that penalizes generated sequences that are shorter than the reference

sequence. The variable N denotes the maximum n-gram order considered in the calculation, typically set to 4. The n-gram precision accounting for length differences is represented by p_n. BP is calculated as follows:

$$\text{BP} = \begin{cases} 1, & \text{if } c > r \\ e^{1-(r/c)}, & \text{otherwise} \end{cases} \tag{18}$$

where, c and r represent the number of tokens in the generated and ground-truth sequences, respectively.

- ROUGE-L (Longest Common Subsequence) measures the longest common subsequence of words between the generated and reference hashtag sequences. It focuses on recall, assessing the extent to which the generated sequence covers the reference sequence.

$$ROUGE - L(G, R) = \frac{LCS(G, R)}{|R|} \tag{19}$$

Here, $LCS(G, R)$ is the length of the longest common subsequence between sequences G and R. $|R|$ is the length of actual sequence i.e., ground-truth hashtags (R). We utilize ROUGE-1 to assess the similarity between the generated hashtag sequence and ground-truth sequence by measuring the overlap of unigrams. This metric is widely used for sequence generation tasks and can identify relevant hashtags even if they are not identical to the ground-truth, which is crucial in hashtag recommendation where multiple hashtags can contribute to conveying the overall topic. Additionally, we examine n-gram overlaps between generated and ground-truth hashtags to evaluate the model's ability to identify and utilize salient information from text for hashtag generation.

Role of Semantic Similarity Metrics in Hashtag Generation: Hashtags serve as a bridge among tweets and broader conversations on social media. Unlike traditional text generation tasks, where Exact Matches (EM) score are often critical, hashtag generation requires a balance between relevance, popularity, and semantic coherence. For instance, a tweet about "flood relief efforts" could be associated with multiple semantically similar hashtags, such as #FloodRelief, #DisasterResponse, or #EmergencyAid. While these hashtags are not exact matches, they are contextually appropriate and align with the tweet's content. Therefore, evaluating hashtag generation solely based on EM would overlook semantic relationships that make hashtags effective in real-world applications.

Limitations of Exact Match Metrics Exact Match (EM) metrics, while useful in tasks such as question answering or information retrieval, are less suitable for evaluating hashtag generation due to the following reasons:

- Variability in Hashtag Usage: A single concept can be expressed using multiple hashtags (e.g., #COVID19, #Coronavirus, #Pandemic). EM metrics would penalize such variations, even though they are semantically equivalent.
- Contextual Relevance: Hashtags often capture broader themes or sentiments rather than specific keywords. For example, a tweet about "donating to earthquake victims" might use hashtags such as #HumanitarianAid or #DisasterRelief, which are contextually relevant but not exact matches.
- Dynamic Nature of Hashtags: Social media trends evolve rapidly, and new hashtags emerge frequently. EM metrics cannot account for the dynamic and adaptive nature of hashtag usage.

Advantages of Semantic Similarity Metrics To address limitations of EM metrics, we employ semantic similarity metrics such as BERTScore, ROUGE-1, and BLEU. These metrics are better suited for evaluating hashtag generation because of the following reasons:

- Capture Contextual Relevance: BERTScore, for instance, leverages contextual embeddings from pre-trained language models (BERT) to measure the semantic overlap between generated and ground-truth hashtags. This ensures that contextually appropriate hashtags are rewarded, even if they are not exact matches.
- Evaluate Coherence and Fluency: ROUGE-1 and BLEU assess the overlap between generated and ground-truth hashtags at the n-gram level, providing insights into the coherence and fluency of generated sequences.
- Align with Real-World Use Cases: In practice, hashtags are used to categorize and discover content based on themes rather than exact keywords. Semantic similarity metrics align with this use case by prioritizing relevance over exact matches.
- Complementary Role of dist. 1: In addition to semantic similarity metrics, we used dist. 1 to evaluate the proximity of generated hashtags to ground-truth hashtags in the embedding space. This metric complements BERTScore, ROUGE-1, and BLEU by providing a geometric perspective on the relevance of generated hashtags.

Together, these metrics offer a comprehensive evaluation framework that balances semantic relevance, coherence, and contextual appropriateness.

Dataset Analysis

Disaster Type Distribution To evaluate potential biases in our dataset, we provide a detailed breakdown of the distribution of tweets across different disaster categories. Table 4 presents the number and percentage of tweets associated with each disaster type. As shown in Table 4, the dataset exhibits a diverse representation of disaster types, with floods being the most prevalent (29.64%), followed by hurricanes (16.44%) and earthquakes (10.80%). Conversely, disasters such as tornadoes (2.35%), typhoons (2.42%), and viruses (1.25%) are less represented. This imbalance reflects the frequency and visibility of disasters in social media discourse, where high-impact events such as floods and hurricanes naturally generate more engagement. While this skew mirrors real-world attention patterns, it may limit the model's generalizability to less frequent disasters. Further, we discuss implications of this imbalance in the limitations section.

Disaster	Number of Tweets	(in %)
Tornado	627	2.35
Hurricane	4386	16.44
Fire	1567	5.87
Earthquake	2882	10.8
Flood	7904	29.64
Haze	1830	6.86
Typhoon	645	2.42
Virus	332	1.25
MERS	735	2.76
Cyclone	2486	9.32
Tsunami	1861	6.98
Explosion	1410	5.56

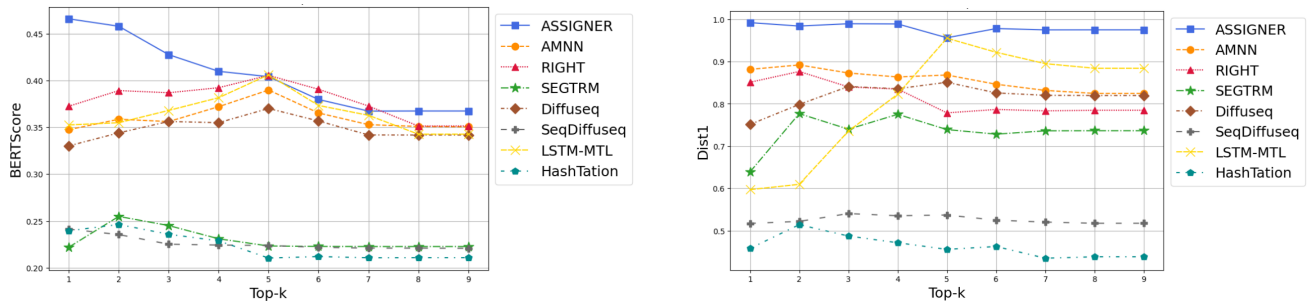
Table 4: Dataset distribution by number of tweets and percentage of total tweets for each disaster type.

Compared Methods

1. AMNN (Yang et al. 2020) employed a seq2seq encoder-decoder architecture with CNN for visual and Bi-LSTM for textual feature extraction from multimodal microblogs. An attention mechanism identifies salient information, and a GRU-based decoder generates the hashtag sequence.
2. SEGTRM (Mao et al. 2022) proposed a model for microblog hashtag generation that operates in three phases. The encoder processes the input text using segment tokens and various attention mechanisms. A segment-selector block identifies important segments based on semantic similarity, while the decoder generates hashtags sequentially using selected segmental representations. The model efficiently determines the number of hashtags required and learns to generate hashtags based on post content.
3. HashTation (Diao et al. 2023) The authors propose a multi-component framework for hashtag recommendation and tweet classification, with four main modules namely, Hashtag Generator, Tweet Attention Module (TAM), Entity Attention Module (EAM), and Tweet Classifier. It begins with Hashtag Generator using self-attention mechanism to create hashtags of an input tweet. TAM is combined with a cross-document attention network to capture latent topics in relevant tweets within a collection and thus improve hashtag generation. EAM employs a graph attention network to extract and utilize semantic information at the entity level, thereby constructing an entity graph from named entities present in tweets. The Tweet Classifier then utilizes a transformer-based encoder equipped with a classification head to classify tweets.
4. LSTM-MTL (Chowdhury, Caragea, and Caragea 2020) developed a joint-layer Long Short Term Memory (LSTM) network trained using Multi-Task Learning (LSTM-MTL) to recommend hashtags for disaster-related tweets. The authors incorporate features capturing informal writing and identify relevant hashtags based on disaster name, location, and situational awareness. The model's variant, utilizing ELMo embeddings, Parts of Speech (POS) tags, and CNN-encoded phonetic features, achieves the best overall performance.
5. RIGHT (Fan et al. 2024) proposed a mainstream hashtag recommendation framework comprising a retriever, selector, and generator. The retriever employs sparse (BM25) and dense (SimCSE) retrieval techniques to identify relevant tweet-hashtag pairs. The selector utilizes a contrastive learning approach with three features (hashtag similarity, frequency, and positive/negative samples) to filter non-mainstream hashtags. The generator combines the input tweet with selected hashtags and employs a generative model fine-tuned with cross-entropy loss to recommend final set of hashtags, ranked by similarity and frequency.
6. Diffuseq (Gong et al. 2022) a diffusion-based model for conditional text generation adapted for seq2seq tasks. The model employs a partial noising strategy, injecting noise only into the target sequence during the forward process. In the reverse process, a transformer architecture predicts the mean and standard deviation of the distribution at each step to denoise the target sequence. The training objective is designed as a variational lower bound with regularization terms. The model also utilizes importance sampling to address training inefficiencies. During inference, sequences are generated by sampling from a learned diffusion process and employing Minimum Bayes Risk (MBR) decoding for quality enhancement.
7. SeqDiffuseq (Yuan et al. 2024) proposed a diffusion-based model for seq2seq language generation. In the forward process, the target output sequence is transformed into random noise. The reverse process utilizes a denoising function, conditioned on the input sequence, to iteratively reconstruct the sequence. The model employs an encoder-decoder transformer architecture and incorporates self-conditioning to improve text quality. It also features an adaptive noise schedule that adjusts the denoising difficulty at each token position and time step, enhancing generation performance.

Visualisation of Quantitative Results

To enhance the readability of our experimental results, Figure 3 presents a comparative analysis of hashtag recommendation models' performance across varying recommendation set sizes ($top-h$), ranging from 1 to 9. As depicted in Figure 3a, ASSIGNER consistently achieves the highest BERTScore values across all $top-h$ settings, indicating superior semantic similarity between its generated and ground-truth hashtags. The observed stability of ASSIGNER's BERTScore, even with increasing recommendation set sizes, demonstrates its robustness. In contrast, AMNN, RIGHT, SeqDiffuseq, LSTM-MTL, HashTation, and SEGTRM, exhibit lower BERTScore values, highlighting their limitations in capturing semantic relevance. Figure 3b further illustrates ASSIGNER's superior performance in terms of hashtag diversity, achieving dist. 1 scores close to



(a) BERTScore (b) dist. 1

Figure 3: Effectiveness comparison curves. The proposed ASSIGNER significantly outperforms the compared methods in evaluation metrics.

1.0 across all $top - h$ values. While AMNN and RIGHT show relatively better diversity compared to other state-of-the-art (SOTA) methods, their scores remain significantly lower than that of ASSIGNER. SeqDiffuseq, LSTM-MTL, HashTation, and SEGTRM exhibit substantially lower dist. 1 scores, indicating limited diversity. Thus, the graphical representation in Figure 3 visually confirms ASSIGNER's competitive advantage in both semantic similarity and hashtag diversity. The consistent and significant gaps in both BERTScore and dist. 1 between ASSIGNER and existing methods underscore the efficacy of our proposed approach.

Performance Comparison with Noise Scheduling Algorithms

To investigate the impact of different noise scheduling algorithms on performance of ASSIGNER in recommending hashtags for disaster-related tweets, we conducted experiments with various schedulers. As demonstrated in Table 5, the token-level adaptive sigmoid scheduler achieves the highest BERTScore (0.4581), dist. 1 (0.9872), ROUGE-1, and BLEU scores. This scheduler outperforms others, including token-level adaptive (Fibonacci, exponential) and non-adaptive Gaussian scheduler. The success of the token-level adaptive sigmoid scheduler is attributed to its precise and dynamic noise control, which is crucial for disaster-related hashtags where keyword relevance can fluctuate rapidly. The sigmoid curve introduces noise gradually to each token, promoting exploration of hashtag spaces, and then reduces it sharply for fine-grained refinement. This dynamic approach enhances contextual sensitivity by adjusting noise based on specific context of each token within the input tweet and generated sequence. Furthermore, the sigmoid curve, combined with token-level adaptation, allows the model to effectively learn intricate inter-token relationships, crucial for generating pertinent hashtags.

Noise Scheduler	BERTScore	dist. 1	ROUGE-1	BLEU
Gaussian	0.167	0.302	0.001	0.001
Adaptive Linear	0.414	0.846	0.007	0.010
Adaptive Quadratic	0.202	0.565	0.002	0.002
Adaptive Cubic	0.171	0.427	0.001	0.0004
Adaptive Fibonacci	0.383	0.733	0.004	0.002
Adaptive Cosine	0.210	0.405	0.005	0.006
Adaptive Exponential	0.310	0.602	0.012	0.014
Adaptive Sigmoid	**0.458**	**0.987**	**0.008**	**0.011**

Table 5: Performance comparison of ASSIGNER with various noise scheduling algorithms for disaster-related hashtag recommendation, showing optimal results with the token-level adaptive sigmoid scheduler. The best result is highlighted in bold, while the second-best is underlined

Input Tweet:
@LiberalJaxx: @trkrsvoice: #Truckers have blocked #Westboro in at the Flying J in #Joplin! If you're near, stand with them! #tcot ...

Ground-truth Hashtags: #Truckers #Westboro #Joplin #tcot

AMNN: #Truckers #blockade #protest #storm

LSTM-MTL: #Truckers #roadblock #aid

RIGHT: #Truckers #Westboro #support #blockade #relief

SEGTRM: #protest #Joplin #roadblock #cyclone

DiffuSeq: #Truckers #stand #aid

HashTation: #Joplin #blockade #protest

SeqDiffuSeq: #Truckers #roadblock #support #quake

ASSIGNER: #Truckers #Joplin #Westboro #cyclone #support

Figure 4: Example tweet showing hashtag generation errors from different methods, including misclassification (red: #storm, #cyclone, #quake), overgeneralization (blue: #blockade, #protest), and irrelevant predictions.

Error Analysis

To gain a deeper understanding of the model's performance and to address the prevalence of irrelevant predictions, we conducted an error analysis. Figure 4 presents an example tweet which describes a trucker protest in Joplin from test dataset, comparing hashtags generated by ASSIGNER and SOTA methods against ground-truth hashtags. As depicted in Figure 4, a notable prevalence of irrelevant pre-

dictions is observed across various methods. SeqDiffuSeq predicts #quake, and AMNN predicts #storm, neither of which are contextually relevant to the input tweet. Similarly, SEGTRM generates #cyclone, an entirely unrelated hashtag. This demonstrates a common failure mode where methods struggle to accurately capture the specific context of the tweet, generating hashtags that are completely disconnected from the event being described. HashTation also exhibits overgeneralization, generating generic hashtags such as #protest and #blockade. While these hashtags are related to the event, they lack the specificity of ground-truth hashtags (#Westboro and #Joplin). Even ASSIGNER, despite its superior overall performance, generates #cyclone in this instance. The #cyclone hashtag is irrelevant as it represents a distinct event type, lacks geographical or semantic connection to the tweet's content, contradicts other hashtags and the overall message of the tweet. This error suggests a lack of robust contextual understanding and may be attributed to biases within training data.

Parameter Sensitivity Analysis

- Number of Recommended Hashtags ($top-h$): To investigate the impact of this parameter on ASSIGNER's performance, we conducted experiments with varying values of $top-h$, ranging from 1 to 9. This parameter significantly influences the user experience and effectiveness of the recommendation system. The results, presented in Figure 3, show that ASSIGNER achieves the best performance with $top-h$ value set to 2, indicating that recommending two hashtags strikes an optimal balance between providing sufficient information and avoiding overwhelming the user with excessive suggestions. Increasing $top-h$ from 1 to 2 leads to a slight improvement in performance across most metrics. However, further increasing $top-h$ beyond 2 results in a gradual decline in performance, suggesting that recommending too many hashtags can lead to decreased relevance and user satisfaction. This decline could be attributed to the inclusion of less relevant or redundant hashtags in $top-h$ list, potentially diluting the overall quality of recommended hashtags.
- Probability for Self-conditioning (p): The probability parameter (p) in ASSIGNER controls the extent of self-conditioning during training. With probability p, the model conditions on past outputs to generate the next hashtag. With probability $(1-p)$, it explores a wider range of hashtag possibilities without self-conditioning. To determine the optimal p, we experimented with values ranging from 0.1 to 0.9. As shown in Table 6, p value set to 0.7 yielded the best performance, balancing self-conditioning with exploration to generate diverse hashtags. Increasing p from 0.1 to 0.7 led to a noticeable improvement in performance. However, further increasing p beyond 0.7 slightly reduced performance, indicating that excessive reliance on self-conditioning hinders the ability of ASSIGNER in generating diverse hashtags.
- Number of Steps in Diffusion (T): The number of steps (T) in diffusion significantly influences the quality and diversity of generated data. A larger T allows for more variation in data but makes it harder to recover from noise. Conversely, a smaller T limits variation but simplifies noise recovery. To investigate the impact of T on ASSIGNER's performance, we conducted experiments by varying T from 100 to 400. As shown in Table 7, ASSIGNER achieves the best performance with 200 diffusion steps. Increasing T from 100 to 200 improves performance. However, further increasing them beyond 200 results in a slight performance decline, indicating that excessive noise hinders ASSIGNER's ability to generate high-quality hashtags. Therefore, we set T=200 to ensure an effective tradeoff between data variation and noise recovery, generating high-quality and diverse hashtags.
- Stride in Sigmoid Noise Scheduler (K): To determine the optimal stride (K) for discretizing the noise schedule in the adaptive sigmoid scheduler, we conducted experiments with varying values. As shown in Table 8, K value set to 15 yielded the best performance across all evaluation metrics. This value likely balances capturing dynamic denoising difficulty with smoothing noise

Values	BERTScore	dist. 1	ROUGE-1	BLEU
0.1	0.447	0.992	0.005	0.008
0.3	0.410	0.919	0.011	0.014
0.5	0.453	0.991	0.005	0.007
0.7	**0.458**	**0.987**	**0.008**	**0.011**
0.9	0.450	0.990	0.006	0.010

Table 6: Performance of ASSIGNER with varying probability parameter (p) ranging between 0.1-0.9 for self-conditioning during diffusion, showing optimal performance at p=0.7.

Values	BERTScore	dist. 1	ROUGE-1	BLEU
100	0.401	0.977	0.004	0.006
200 (Ours)	**0.458**	**0.987**	**0.008**	**0.011**
300	0.439	0.992	0.006	0.006
400	0.417	0.984	0.006	0.013

Table 7: Performance of ASSIGNER with varying number of steps (T) in Diffusion ranging between 100-400, showing optimal performance ar T=200.

Values	BERTScore	dist. 1	ROUGE-1	BLEU
5	0.411	0.896	0.003	0.003
10	0.452	0.988	0.005	0.009
15	**0.458**	**0.987**	**0.008**	**0.011**
20	0.450	0.990	0.007	0.009
25	0.450	0.984	0.008	0.001
30	0.455	0.989	0.001	0.001

Table 8: Performance of ASSIGNER with varying parameter (k) ranging between 5-30 in sigmoid noise scheduler, showing optimal performance at k=15.

Methods	Time (in seconds)
Sequence Generation	
AMNN	273
SEGTRM	146
HashTation	238
Keyphrase Extraction	
LSTM-MTL	204
Retrieval-Augmented Generation	
RIGHT	312
Diffusion	
Diffuseq	169
SeqDiffuseq	185
ASSIGNER	137

Table 9: Training time per epoch (in seconds) on NVIDIA Tesla T4 GPU. All models were run on the same hardware. Times reported here are the average of multiple runs.

in the loss signal, thereby enabling the sigmoid function to learn a more stable mapping between loss and noise level. Values of K that were either too small (5) or too large (30) resulted in reduced performance, likely due to overfitting to noise or oversimplification of loss-noise relationship, respectively. Therefore, K value set to 15 facilitates effective learning of individual token schedules without excessive sensitivity to minor fluctuations in the loss signal.

Efficiency and Scalability Analysis

To assess ASSIGNER's suitability for real-time disaster response platforms, we evaluate its computational efficiency and scalability.

Computational Time Analysis We analyzed and compared the computational time of ASSIGNER with various SOTA methods. Table 9 presents the training time, in seconds, for each method. Experiments were conducted on a Linux Server equipped with Intel(R) Xeon(R) Silver 4215R CPU @ 3.20 GHz, 256-GB RAM, and 16-GB NVIDIA Tesla T4 GPU. As observed from Table 9, ASSIGNER demonstrates the lowest training time among the compared methods. This efficiency stems from its streamlined architecture, which combines a retrieval mechanism, a selector module, and a diffusion-based generative model. Specifically, the retrieval mechanism identifies semantically similar tweets and associated hashtags, reducing the search space for the generative model. The selector module filters and refines retrieved hashtags, ensuring the generator focuses on pertinent candidates. The diffusion-based generator, built on a BART-based encoder-decoder, employs self-conditioning to guide the generative process, minimizing iterations.

Despite its computational efficiency, ASSIGNER maintains state-of-the-art performance in quantitative metrics while recommending high-quality and contextually relevant hashtags. This balance makes it suitable for real-world applications, particularly in time-sensitive scenarios such as disaster management and social media communication.

Methods	BERTScore	dist. 1	ROUGE-1	BLEU
Cosine similarity	0.198	0.402	0.0003	0.001
SimCSE (Ours)	**0.458**	**0.987**	**0.008**	**0.011**

Table 10: Performance comparison of retrieval methods (Cosine Similarity vs. SimCSE) for hashtag recommendation on disaster-related tweets.

Addressing Diffusion Training Challenges While diffusion-based models are known for their effectiveness, they suffer from long training times. ASSIGNER mitigates this challenge through several key design choices:

- Efficient Retrieval Mechanism: ASSIGNER employs a retrieval mechanism to identify semantically similar tweets and associated hashtags from an existing corpus. This reduces the burden on the diffusion model by providing high-quality and contextually relevant hashtags as initial inputs, thereby shortening the training time.
- Selector Module: The selector module refines retrieved hashtags, filtering out irrelevant or low-quality hashtag candidates. This ensures that the diffusion model focuses only on the most pertinent hashtags, further improving training efficiency.
- Self-conditioning in Diffusion: ASSIGNER's diffusion-based generative model leverages self-conditioning, where previous predictions and selected hashtag embeddings are reintroduced as additional context during the iterative diffusion process. This guides the model towards generating relevant hashtag sequences more efficiently, reducing the number of iterations required for convergence.
- Streamlined Architecture: By integrating retrieval augmentation, selection, and diffusion into a unified framework, ASSIGNER minimizes redundant computations and optimizes resource utilization. This is reflected in its training time of 137 seconds, which is significantly lower than other diffusion-based models such as Diffuseq (169 seconds) and SeqDiffuseq (185 seconds).

Thus, ASSIGNER effectively addresses computational challenges associated with diffusion training while maintaining high performance.

Efficiency of Retrieval Mechanism To evaluate the efficiency and effectiveness of our retrieval mechanism, we compared our chosen dense retrieval method, SimCSE, with an approach using cosine similarity on tweet embeddings. Table 10 demonstrates the significant performance improvements achieved by SimCSE across all evaluation metrics. SimCSE, by leveraging contrastive learning, produces embeddings that are more effective at capturing semantic similarities, leading to improved retrieval of relevant tweet-hashtag pairs. This improved accuracy directly translates to

a more effective selection of candidate hashtags for the subsequent generation stage.

While SimCSE is computationally more intensive than basic cosine similarity during the embedding generation phase, its accuracy reduces the workload of selector and generator modules, leading to overall efficiency gains. Furthermore, high-quality retrieved hashtags enable more effective filtering by the selector, reducing iterations required by the diffusion model. By precomputing embeddings for the entire tweet-hashtag corpus, the retriever reduces online computational overhead. Our experiments demonstrate that SimCSE-based retrieval offers both accuracy and efficiency in RAG frameworks.

Scalability and Real-time Deployment Scalability is primarily dependent on retriever's ability to handle a growing corpus of tweets. The low training time and memory requirements of ASSIGNER make it suitable for deployment in disaster response platforms as a REST API or edge-computing module. The modular design allows for optimization of individual components for further speed improvements. However, real-time deployment in resource-constrained environments may require further optimization, such as model quantization or pruning, which we leave to future work.

Discussion

Theoretical Implications The proposed framework, ASSIGNER offers several theoretical contributions to the field of NLP and hashtag recommendation, enlisted below:

- The integration of a retrieval mechanism with a diffusion-based generator presents a novel approach to hashtag recommendation. This hybrid framework leverages strengths of retrieval and generation, overcoming limitations of traditional methods that rely solely on either retrieval or generation.
- The introduction of an adaptive sigmoid noise scheduler at the token level represents a significant advancement in diffusion models for text generation. This technique allows for fine-grained control over the denoising process, enabling ASSIGNER to capture the dynamic nature of language and generate more contextually relevant hashtags.
- Incorporating self-conditioning into diffusion-based generator enhances the ASSIGNER's ability to utilize previously predicted sequence information, leading to improved coherence and relevance in hashtag generation.
- The proposed framework tackles the challenge of hashtag recommendation in disaster scenarios, where language is often informal and dynamic. By effectively capturing these nuances, the model contributes to a better understanding of how hashtags can be used to organize and disseminate information during critical events.

Practical Implications ASSIGNER, leveraging its retrieval augmented diffusion architecture offers a robust solution for enhancing information dissemination and situational awareness in critical real-world scenarios, particularly disaster management and social media communication.

1. Rapid Information Dissemination and Enhanced Searchability: ASSIGNER can be integrated into social media platforms or emergency response systems to automatically generate relevant hashtags for incoming posts during disaster events. This facilitates the immediate categorization and indexing of critical information, enabling emergency responders and the public to quickly search for and access relevant updates. For instance, a system could analyze incoming tweets about a flood and automatically append hashtags such as #FloodAlert, #EvacuationZone, or #ReliefEfforts. By ensuring that posts are tagged with relevant and widely understood hashtags, ASSIGNER significantly improves the discoverability of time-sensitive information.
2. Real-Time Situational Awareness and Decision Support: The model can be deployed to analyze real-time social media streams, generating hashtags that provide insights into the evolving disaster situation. Emergency management agencies can use generated hashtags to monitor affected areas, identify emerging needs, and track the spread of information. For example, during a wildfire, ASSIGNER could identify hashtags related to specific evacuation routes or requests for medical assistance. This real-time hashtag generation can be integrated into dashboards and decision-support systems, providing up-to-the-minute insights for informed decision-making.
3. Streamlined Communication and Coordination: ASSIGNER can be used to standardize hashtag usage across multiple communication channels, ensuring that different stakeholders use consistent terminology. This is particularly important in large-scale disaster response efforts, where coordination between various agencies and organizations is crucial. The model can also be used to identify and promote the use of widely recognized and informative hashtags, facilitating efficient communication and coordination.
4. Targeted Needs Assessment and Resource Allocation: By analyzing generated hashtags, emergency responders can quickly identify specific needs and requests for assistance. For instance, hashtags such as #MedicalAid or #ShelterRequest can be used to prioritize resource allocation and ensure that aid reaches those who need it most. ASSIGNER can also be used to identify emerging patterns and trends in social media data, providing valuable insights for targeted interventions.
5. Social Media Analytics and Public Sentiment Analysis: Generated hashtags can be used to analyze public sentiment and track the impact of disaster on social media. This information can be used to inform public communication strategies and improve disaster response efforts. For example, the model can be used to identify hashtags related to public concerns or misinformation, enabling authorities to address these issues proactively.

In summary, ASSIGNER's retrieval augmented diffusion architecture offers a practical solution for leveraging social media data in real-world disaster management scenarios, enhancing information dissemination, coordination, and decision-making.

Limitations and Potential Societal Impacts While ASSIGNER offers several advantages, it is essential to acknowledge its limitations and potential societal impacts.

- Dataset Limitations:
  - The dataset exhibits a non-uniform distribution across disaster types, potentially introducing biases favoring hashtags associated with more prevalent disasters.
  - The dataset primarily consists of English tweets, limiting the model's generalization to multilingual disaster scenarios. While ASSIGNER's diffusion-based architecture is theoretically language-agnostic, its current performance is optimized for English due to training data constraints. Multilingual generalization remains a challenge, as informal language, dialects, and cultural nuances in non-English tweets may reduce recommendation accuracy.
- Model Limitations:
  - The effectiveness of selector module depends on the comprehensiveness and diversity of tweet-hashtag corpus. Biases in the corpus could influence selection, leading to incomplete or skewed recommendations.
  - Hashtags recommended by ASSIGNER could amplify biases or contribute to misinformation if the training data is biased or inaccurate.
  - Edge Cases: ASSIGNER may underperform in edge cases, such as tweets with ambiguous language, sarcasm, or very low information content. These cases present challenges for both the selector and generator modules, potentially resulting in less relevant hashtag suggestions.
- Societal Impacts: Over-reliance on automated hashtag recommendations could diminish human oversight and critical thinking in hashtag selection, potentially overlooking context-specific information.

Future Work To further enhance the efficacy and societal impact of automated hashtag recommendation systems for disaster response, future research should focus on several key areas. First, expanding and diversifying the training corpus can reduce bias and improve the model's ability to handle diverse disaster scenarios. Second, developing more robust selection mechanisms that are less susceptible to ambiguity and better capture disaster-related language characteristics is crucial. Future researchers can optimize the retrieval process to improve efficiency and scalability for large-scale applications. To enhance cross-linguistic adaptability, future work will explore posts in multiple languages and corresponding datasets. Future work may incorporate EM metrics alongside semantic similarity metrics to provide a more holistic evaluation of hashtag generation models. By addressing these limitations, future research can contribute to the development of more reliable, responsible, and effective automated hashtag recommendation systems for disaster response.

Ethical Considerations and Safeguards While ASSIGNER demonstrates potential for enhancing disaster communication, it is imperative to acknowledge and address potential ethical risks.

1. Misinformation Amplification: The automated generation of hashtag recommendations by ASSIGNER may inadvertently amplify misinformation if the training data contains biases or inaccuracies. Malicious actors could exploit this functionality to disseminate hashtags that propagate false claims, such as designating a hazardous area as a #SafeZoneDuringHurricane. Although the selector module aims to mitigate irrelevant hashtags, it cannot guarantee complete elimination of harmful content.
2. Bot-Generated Spam: Automated accounts may misuse hashtag recommendations to flood platforms with irrelevant and promotional content during disaster events. This could disrupt legitimate information dissemination and impede effective disaster response.

Mitigation Strategies:

- Human-in-the-Loop Validation: We recommend implementing a post-processing filter that cross-references generated hashtags with verified information sources. Furthermore, moderators should review recommended hashtags before dissemination in critical scenarios.
- Bias Audits and Corpus Maintenance: Researchers should regularly audit the retriever's corpus to identify and remove biased or outdated hashtags and implement strategies to ensure corpus diversity.
- Rate Limiting: Implementing rate-limiting and anomaly detection mechanisms can aid in mitigating the impact of bot-generated spam.
- Explainability: Providing confidence scores for recommended hashtags can enable users to filter low-confidence suggestions, enhancing transparency and user control.

We emphasize that ASSIGNER should augment, not replace, human decision-making in disaster response.

Ethics Statement

The dataset used in this study was publicly available under the terms of the Apache License Version 2.0, January 2004 [3]. We further refined the data according to our needs. Proper attribution has been given to the original dataset creators. While the dataset is publicly available, we acknowledge the potential for inherent biases that could affect the model's predictions. We are committed to ongoing efforts to identify and mitigate such biases in future work. Additionally, we recognize the potential for misinformation and harmful content in disaster-related tweets. Our model is designed to prioritize relevant and informative hashtags, and we plan to incorporate mechanisms to filter out inappropriate or misleading content in future iterations. We are mindful of the ethical implications of automated hashtag recommendation, particularly during sensitive events such as disasters. We are committed to ensuring that our research is used responsibly and ethically. Moreover, we will continue to explore potential safeguards to prevent the misuse of our model.

[3] https://github.com/JRC1995/Tweet-Disaster-Keyphrase/tree/master/Data