Transformer-based identification of stochastic information cascades in social networks using text and image similarity

Panagiotis Kasnesis†, Ryan Heartfield*, Xing Liang*, Lazaros Toumanidis†, Georgia Sakellari*, Charalampos Patrikakis†, George Loukas*

†*University of West Attica,* **University of Greenwich*

Abstract

Identifying the origin of information posted on social media and how this may have changed over time can be very helpful to users in determining whether they trust it or not. This currently requires disproportionate effort for the average social media user, who instead has to rely on fact-checkers or other intermediaries to identify information provenance for them. We show that it is possible to disintermediate this process by providing an automated mechanism for determining the information cascade where a post belongs. We employ a transformer-based language model as well as pretrained ResNet50 model for image similarity, to decide whether two posts are sufficiently similar to belong to the same cascade. By using semantic similarity, as well as image in addition to text, we increase accuracy where there is no explicit diffusion of reshares. In a new dataset of 1,200 news items on Twitter, our approach is able to increase clustering performance above 7% and 4.5% for the validation and test sets respectively over the previous state of the art. Moreover, we employ a probabilistic subsampling mechanism, reducing significantly cascade creation time without affecting the performance of large-scale semantic text analysis and the quality of information cascade generation. We have implemented a prototype that offers this new functionality to the user and have deployed it in our own instance of social media platform Mastodon.

Keywords: Information Cascade, Semantic Textual Similarity, Image Similarity, Deep Learning

1. Introduction

When coming across a new piece of information posted on social media, users may wish to assess its trustworthiness. To do so, they either rely solely on their own knowledge and intuition or take considerable time to check where this information came from in the first place and whether it has been modified since first published. However, investigation on information provenance is not trivial and as such, many social media users will not have the time, motivation or knowledge to conduct it. Instead, they may rely on intermediaries, such as third-party fact-checkers or the social media platforms to do it for them. Even if we assume that these intermediaries are always correct and trustworthy themselves, by the time a false rumour has been fact-checked, it has already spread to a large part of the population. In fact, there is a trade-off between the number of people required to flag a post before it is forwarded for professional assessment versus the number of people exposed to it until it is assessed [1]. At the same time, misinformation travels faster than reliable information (one sixth of the time it took truth to reach 1500 people in [2]), and posts made by individuals or organisations who are experts in a particularly subject or topic (which is going viral) may not necessarily be visible to users due to author/post popularity (e.g., followers, likes, re-shares etc.) [3].

If users themselves were able to identify more easily the provenance of a post's information at the point of accessing it, they would think twice before resharing it and this would naturally curb the spread of "infodemics". Here, we take the first steps towards such a provision. Contrary to most existing research in this area, where information cascades are built in a deterministic manner based on explicit resharing (e.g., retweets on Twitter), our approach is stochastic, looking at the degree of similarity between different posts. The little prior work that exists in this area has used statistical word similarity, which however misses posts where the semantics may be the same even if the wording is not. In addition, the previous work has used only textual similarity, while the spreading of news or rumours on social media makes heavy use of images (the average number of reposts with images being estimated to be 11 times larger than those without images [4]). Here, we explore whether incorporating image similarity together with textual similarity can improve the identification of information cascades in social media.

Specifically, this paper introduces the following novel contributions to the body of machine learning techniques for addressing misinformation in social media [5]:

- A method for monitoring implicit information diffusion and its resulting information cascades over social networks

- A method for improving clustering performance by combining textual and image similarity detection based on deep learning

- An efficient post subsampling method to increase the scalability of our approach based on sentence embeddings

- A prototype tool implementing automated information cascade identification on an existing social media platform

2. Related Work

2.1. Identifying information cascades

Information diffusion has been studied since the beginning of the social media phenomenon as part of the pattern and knowledge discovery dimension of Camacho et al.'s four dimensions of social media analysis [6]. Using explanatory or predictive modelling, the aim is typically to derive latent information about users and communities of users [7]; why information has been diffused in a particular way; where it will be diffused in the future [8] and whether [9, 10] or how [11] it will "go viral" (for marketing [12], political [13] or other reasons). In terms of provenance of information in social media, most existing research has focused on explicit diffusion, as captured for example through retweets on Twitter and shares on Facebook [14, 15]. This kind of provenance is deterministic, as the social media platform itself guarantees the path the information travelled. However, after users come across a post on social media, they may repeat its content without explicitly resharing it word for word. The information is still spreading, yet this cannot be captured by explicit diffusion models.

Having utilised post similarity between users' own posts and their friends' recent posts to reconstruct information cascades, Barbosa et al. [16] reported that at least 11% of interactions are not captured by the explicit reply and retweet/share mechanisms. Taxidou et al. [17] have also shown that limiting to explicit resharing cannot capture accurately the influence that a post has had. Instead, they proposed looking at implicit diffusion too, and in their work they suggested reconstructing information cascades using statistical word similarity based on TF-IDF (Term Frequency–Inverse Document Frequency). Here, we adopt the same direction of implicit diffusion leading to stochastic information cascades, but we progress beyond statistical similarity to semantic similarity, as different users may describe the same information using very different wording. In addition, the same or very similar images may be used to describe the same piece of news even if the text appears different. In these cases, considering image similarity in conjunction with semantic text similarity can add context that has not been previously considered in identifying information cascades in social media.

2.2. Transformers in text similarity tasks

For Natural Language Processing (NLP) tasks, such as those gaining increasing attention in social media for analysing information provenance and credibility, Deep Learning (DL) models and in particular Recurrent Neural Networks (RNNs) empowered with Long-Short Term Memory (LSTM), have gained widespread popularity [18] because of their ability to capture the semantics of the words and in consequence generalize over a range of contexts. Recent works use baseline machine learning models such as Latent Dirichlet Allocation (LDA) empowered with word semantics to improve clustering of aspect terms according to their aspect category [19] and topic modeling [20]. Support Vector Machines (SVM) have also been used towards this direction by being fed with two dense vectors to determine the degree of semantic similarity between two input sentences. The first one utilizes word-to-word similarity based on Word2Vec embeddings [21] and the latter is built using the word-to-word similarity based on external sources of knowledge [22].

However, these DL and baseline NLP architectures have been observed to lack the capability to support inductive transfer learning when it comes to new NLP tasks, because fine-tuning pretrained word embeddings (e.g. Word2Vec [21], Glove [23]) only target a model's first layer and also because the main task model (e.g., the specific NLP task to be addressed) requires training from scratch. In response to this limitation, Language Models (LM) have been proposed [24], which distinguish contextually between similar words and phrases by incorporating the distribution over sequences of words into model weights. Initially, LM architectures were found to lack computational efficiency, since they preclude parallelization, making it a constraint when it comes to training big sequence lengths. However, recent work based on Transformers-based network architectures [25] have revolutionized

NLP problems by replacing the RNNs with Multi-Head Self-Attention (see Subsection 3.1). Transformers rely on an encoder-decoder architecture to extract the meaning from word representations and their relationships, and can be fine-tuned on a wide range of NLP tasks, such as question answering and paraphrase identification, without substantial architecture modifications [26].

Bidirectional Encoder Representations from Transformers (BERT) [26] is a LM based on a transformer network [25] designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. For pretraining, BERT relies on self-supervised learning and, in particular, has two objectives: a) Masked Language Modeling (MLM), and b) Next Sentence Prediction (NSP). In MLM, a random sample of the tokens (15% of the input sentence) is removed and replaced with the special token [MASK]. The objective of the model is to predict the masked tokens using a cross-entropy loss function. Regarding NSP, it is a binary classification task that aims at predicting whether two sentences follow each other in the original text, thus negative examples are artificially created by pairing sentences from different documents.

Robustly Optimized BERT Pretraining Approach (RoBERTa) [27] is an optimized BERT successor with several modifications to improve the LM pretraining: a) training the model longer, with bigger batches, over more data; b) removing the NSP objective; c) training on longer sequences; and (d) dynamically changing the masking pattern of the MLM. As a result, RoBERTa has managed to surpass BERT's performance on every NLU task included in GLUE (General Language Understanding Evaluation) benchmark [28], including Paraphrase Identification (PI) and Semantic Textual Similarity (STS) tasks.

Surprisingly, despite their generalizability in several tasks, BERT and RoBERTa do not provide efficient sentence embeddings [29]. Averaging the word embeddings of BERT provides worse latent sentence representations than other models trained on this task, such as Universal Sentence Encoder (USE) [30], a transformer-based network combined with a deep averaging network [31] specifically trained to produce meaningful sentence embeddings. To this end, Sentence-BERT (SBERT) and Sentence-RoBERTa (SRoBERTa) models have been introduced in [29]. They are comprised of two identical networks (e.g., BERT), where each one has a different sequence as input and the objective is to decide whether the two sentences are semantically similar by using cosine similarity as a distance metric, extracting useful embeddings in this way.

2.3. Image in information diffusion tasks

In addition to text, information diffusion in social media has also been studied in relation to images, for predicting the future popularity of a given piece of information [32, 33] or the proliferation of misinformation [34]. For example, Jin et al. [4] have found that images used in disinformation can have distinctive distribution patterns both visually and statistically. McParlane et al. [35] have focused on image popularity prediction by considering visual appearance, content and context. Relevant to our work is Cheng et al.'s work [33] which used image matching to identify copies of the same image and place them into corresponding cascades, but without considering text similarity.

More recently, pretrained deep learning models such as VGG16, VGG19, ResNet50, InceptionV3, Xception, InceptionResNetV2 are increasingly adopted to retrieve high level image features [32][36][37][38]. In [36], pre-trained model InceptionResNet V2 was used to derive useful information from photos for popularity prediction in social media. VGG19 was adopted in [37] to extract deep features in addition to extracting basic features including texture and colour of images. Galli et al. [32] have used VGG16 to take sentiment into consideration for social media popularity prediction.

In this paper, we propose the use of two deep learning architectures to extract both visual and textual information and fuse them together afterwards to evaluate how similar two posts are. In particular, we collected posts from Twitter to monitor how information spreads in social media by identifying diffusion of the posts containing the same or similar content (i.e., text and/or images). This could benefit not only misinformation detection but also various pattern recognition applications such as information retrieval, classification, clustering and change detection.

3. Discovering Probabilistic Information Cascades

In social media, implicit information diffusion processes [17] between posts can manifest over varied conditions based on their content, such as whether a post contains text, an image, video, URL, or any combination of these. If different posts have sufficient similarity between these respective content features, they can be considered the same or slightly different versions of the same information. Here, we focus on discovering information cascades taking into consideration both text and image content similarity. Below, we provide a detailed overview of the algorithms and models used, with examples demonstrating the objectives of these methods for reliably linking posts within associated information cascades, followed by an overview of their integration into a systematic information cascade discovery pipeline.

3.1. Text similarity

Text similarity deals with determining how similar two pieces of text are. It is considered to be a Natural Language Understanding (NLU) problem that, unlike NLP, deals with machine reading comprehension. Therefore, the objective of text similarity is to identify whether two or more pieces of text represent the same information, albeit with varied use of language, and as such, a trained Artificial Intelligence (AI) model should be able to process natural language in a way that is flexible and not exclusive to a single task, genre or dataset. Typically, in the field of NLP and NLU, this is considered to be an AI-hard problem[28].

To develop our text similarity evaluation for information cascade discovery, we have chosen RoBERTa$_{LARGE}$ model for the text similarity and feature extraction tasks. RoBERTa follows an encoder-decoder network architecture. The encoder part is composed of a stack of $N = 12$ identical layers, where each of them has two sub-layers connected in a residual manner and followed by layer normalization. The first sub-layer is a Multi-Head Self-Attention mechanism, and the second is a fully connected feed-forward neural network. Residual blocks introduce skip connections are employed around each of the two sub-layers and finally produce embedding outputs of dimension $d_{model} = 1024$. The decoder is composed of a stack of $N = 12$ identical layers, but includes a further sub-layer (three in total) to perform Multi-Head Attention over the output of the encoder. Like the encoder, residual connections are used for merging their outputs, followed by layer normalization [25, 27].

The efficiency of transformers is mostly based on the Multi-Head Self-Attention mechanism, which defines which parts of a sentence are highly related with each other. In practice, this mechanism makes use of a set of queries Q applied to a set of keys K and provides the most relevant values V. The Self-Attention is given by:

$$A = softmax\left(\frac{QK^T}{\sqrt{d}}\right)V \tag{1}$$

where d is the dimensionality of the key vectors used as a scaling factor. The Multi-Head Self-Attention enables the model to attend to several and different representation subspaces at different positions by concatenating the outputs of the heads.

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^o \tag{2}$$

where h denotes the number of heads, which in the RoBERTa$_{LARGE}$ case are equal to 16. W^o represents the weights of the dense layer that follows the Multi-Head Self-Attention.

An advantage of RoBERTa against BERT for text-based information cascade identification is its pretrained architecture which benefits from a more diverse range of datasets (larger corpus). For example, its training corpus includes the CommonCrawl News dataset[1] which contains 63 million English news articles and has a larger vocabulary size (50 thousand units) compared to BERT's (30 thousand units).

We used the SRoBERTa large [2] model pretrained on two NLI datasets, SNLI [39] and MultiNLI [40]. SNLI consists of 570,000 sentence pairs annotated with the labels entailment, contradiction, and neutral, while MultiNLI is a collection of 433,000 crowdsourced sentence pairs, containing the same labels but covering a range of genres of spoken and written text. SRoBERTa$_{LARGE}$ was trained using a batch size of 16, Adam optimizer with learning rate 2e-5, and a linear learning rate warmup over 10% of the training data [29].

The model was retrained and evaluated afterwards on the Semantic Textual Similarity Benchmark (STS-B) dataset [41] reaching a score of 86.39 in Spearman's rank correlation; it is a collection of sentence pairs, comprised by 7,000 training and 1,400 test samples, drawn from news headlines, video and image captions, and NLI data. The pairs are human-annotated with a similarity score from 1 (lowest) to 5 (highest), while the task is to predict these scores. A model's performance on this task is evaluated using Pearson and Spearman correlation coefficients, while it should be noted that it is a regression task.

In our approach, we exploit the retrained on STS-B model to extract useful text embeddings from the input posts. The extracted embeddings are represented by an array consisting of 1024 float numbers. After acquiring

[1]http://commoncrawl.org/2016/10/newsdataset-available

[2]https://github.com/UKPLab/sentence-transformers

Table 1: Examples included in the STS-B train set

Examples	Normalized STS score
1: A man is smoking. **2**: A man is skating.	0.10
1: Three men are playing chess. **2**: Two men are playing chess.	0.52
1: A man is playing the cello. **2**: A man seated is playing the cello.	0.85

the embeddings of an input posts we apply cosine similarity to identify the N most similar existing posts and pass them to STS service.

In relation to textual similarity and paraphrase identification, we used two alternative approaches to train our STS model. The RoBERTa$_{LARGE}$ was trained seperately on the Microsoft Research Paraphrase Corpus (MRPC) for Paraphrase Identification (PI) and on STS-B for STS. The MRPC dataset [42] is a corpus of sentence pairs (3,700 training and 1,700 test samples) included in online news sources, annotated by humans to define whether the sentences in the pair are semantically equivalent; it is imbalanced (68% positive, 32% negative pairs). Unlike STS-B, MRPC is a dataset handled by classification algorithms.

We trained the model for both datasets using a batch size of 8 and Adam optimizer with learning rate 1e-5 for 5 epochs, achieving results almost identical with those reported in [27]. The evaluation of these two models is presented in Section 4. It should be noted that while the RoBERTa$_{LARGE}$ MRPC model produces outputs from 0 to 1, the RoBERTa$_{LARGE}$ STS-B model produces outputs from 1 to 5. So, during the decision process, they are normalized by dividing by 5. Table 1 presents some examples from the STS-B training set.

3.2. Image similarity

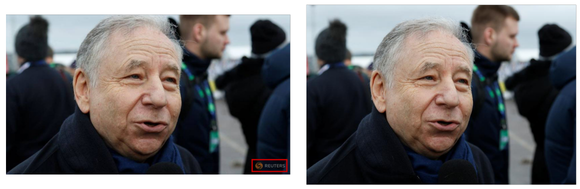
Due to context, such as date and occasion, the conditions for assessing image similarity in information diffusion tend to be stricter than text similarity. For example, consider two separate images of a politician taken in direct point-of-view, standing at the exact same lectern, in the exact same room, holding a government press conference on television on different days. In both images, the politician is wearing a suit, in one image blue, and in the other black. In this case, the images are likely to yield high similarity with respect to their content, but they should be considered different images and representative of different information contextually. On the contrary, considering the two images of the same nature, where the politician wearing the black suit, on the same day, with a news broadcasting logo overlaid on the bottom right of the image, and the other with no news channel logo visible, should be considered the same image and representative of the same information contextually.

However, relying on similarity analysis of images alone for reliable information cascade discovery is naturally prone to false positives, because images related to branding and advertisements (e.g., the "breaking news" image or a company's logo) are often reused. This may cause the erroneous creation of information cascades between them when there is no real connection between them other than the reuse of a generic image. To address this, it makes sense to combine image similarity with text similarity and deriving a combined similarity metric.

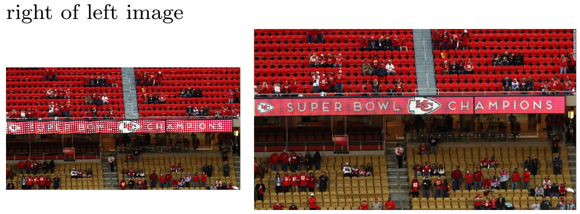
To illustrate the requirements of strict image similarity in information cascade generation, in Figure 1 we provide an example of three pairs of social media post images from our TNCD dataset, which are related to the same piece of information. In *Exhibit A*, we can observe that the exact same image has been used between two posts, with a small news logo overlaid on the bottom right the first image, and with the images being a different resolution. In *Exhibit B*, the same image has been used, with the first image being a lower quality, and smaller resolution than the second. Finally, in *Exhibit C*, it is clear these are different images but are related to the same sportsperson, at the same event.

To evaluate image similarity in the context of information cascade discovery, we adopted existing approaches in image embeddings and metric learning. In image embedding, a robust and discriminative descriptor is learned to represent each image as a compact feature embedding. Typical descriptors include SIFT [43], LBP [44], ORB [45], HOG [46] and Convolutional Neural Network (CNN) embedding's [47]. In this work we employ feature descriptors generated by an existing CNN which employs unsupervised learning to extract latent features, implemented in a Keras pretrained model, as the base for our image feature embeddings generation. For the purposes of comparison, we have used two CNNs: *ResNet50* [48] (Figure 2) and the Visual Geometry Group (VGG) submission to the ImageNet Challenge [49].

The image embeddings are extracted by the deep CNN network, which has multiple layer (M) and n_m neurons in the m^{th} layer (m= 1,2, ...M). For a given



(a) **Exhibit A (0.962 similarity)**: The same image, with different resolution and news broadcaster logo on bottom right of left image



(b) **Exhibit B (0.945 similarity)**: The same image with different resolution and varying image quality

(c) **Exhibit C (0.689 similarity)**: Different images of same sportsman at the same event.

Figure 1: Comparison of image similarity based on strict cascade link requirements

image, E_m is the output of the m layer, where $E_m = \sigma(W_m x + b^m)$: W_m is the projection matrix to be learnt in the m^{th} layer and b^m bias vector; σ is the non-linear activation function. In each of the CNN networks, a parametric non-linear function f: image $\rightarrow E_m$ projects an image of D dimensions into a sub-space of N dimensions in the m^{th} layer. In this sub-space similar images would be closer to each other and dissimilar images to be further apart.

Residual Networks (ResNets) introduce skip connections to skip blocks of convolutional layers, forming a residual block [48]. These stacked residual blocks greatly improve training efficiency and largely resolve the vanishing gradient problem present in deep networks. With a top five accuracy of 93.29%, ResNet50 model won the ImageNet challenge [49] (or ILSVRC), which is an annual competition using a subset of ImageNet [50] (a large visual database designed for use in visual object recognition of over 15 million labelled high-resolution images belonging to roughly 22,000 categories) and

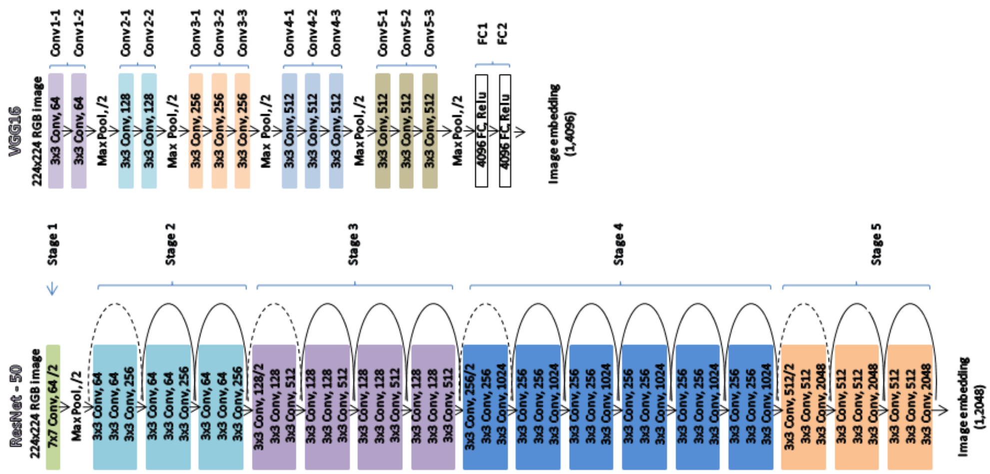


Figure 2: VGG16 and ResNet50 image embeddings

is designed to foster the development and benchmarking of state-of-the-art algorithms. ResNet50 learns a $2048N$ dimensional embeddings of an image from the last layer of stage five (see Figure 2). In contrast, VGG16 has 13 convolutional and 3 Fully Connected (FC) layers, and was employed to learn a $4096N$ dimensional embeddings of an image from FC2 layer. See Figure 2 for a process comparison between ResNet50 and VGG16 image embedding.

In metric based learning, a distance metric is utilised to learn from CNN-embeddings in an latent space to effectively measure the similarity of images. Considerable efforts have been made to define intuitive image distances in information retrieval [51, 52, 53, 54], including Cosine similarity, which measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether they are pointing in roughly the same direction. It is often used to measure image similarity as well as document similarity in text analysis (as in Section 3.1).

For each pair of images (I_i, I_j) with image embeddings (E_{mi}, E_{mj}), image similarity is computed by cosine similarity on image embedding features based on Eq. 3:

$$ImageSimilarity_{(I_i,I_j)} = \frac{\sum_{n=1}^{N} E_{(mi,n)} * E_{(mj,n)}}{\sqrt{\sum_{n=1}^{N} E_{(mi,n)}^2} * \sqrt{\sum_{n=1}^{N} E_{(mj,n)}^2}} \quad (3)$$

3.2.1. Information cascade pipeline

To identify information cascades in a manner which is practical for real-world deployment, we have developed a pipeline for iterative evaluation of social media posts as they are shared online. When a new post is published, we immediately assess its similarity against all existing posts published up to that point. This is possible by employing an efficient subsampling technique using cosine similarity analysis, which we describe in step 2 of the *Information Cascade Pipeline* below. To demonstrate the utility of the subsampling process, in Figure 7 (see section 4.3) we illustrate how the pipeline cosine subsampling latency, combined with RoBERTa$_{LARGE}$ STS latency (i.e., using a fixed subsample of i posts based on the highest cosine scores, as described in step 3 of the pipepline), is capable of processing millions of posts in under 5 s using our single computer testbed configuration. The *Information Cascade Pipeline* can therefore support information cascade discovery in webscale online social media platforms.

The *Information Cascade Pipeline* implements the following steps: 1) Extract Feature Embeddings, 2) Subsample Candidate Posts, 3) Semantic Text Similarity, 4) Post link threshold algorithm. In Figure 3, the *Information Cascade Pipeline* is illustrated visually with notation for each processing steps' algorithmic inputs and outputs (See Table 2 for notations).

In step 1, after a new post p is published to a social media platform and stored in the platform database, its text content p_t and image content p_m are extracted to generate post sentence embeddings ($p_{t,f}$) and image sentence embeddings ($p_{m,f}$), using our RoBERTa$_{LARGE}$ and ResNet50 models, respectively. Note that the Information Cascade Pipeline is only activated for newly published posts if the post contains at least three words, with or without an image. Where this condition is met, extracted feature embeddings are stored in a post database alongside existing original post content for future post similarity analysis (i.e., when new posts are published). In step 2, the set of all existing post feature vectors E is queried from the post database and a pairwise comparison of the newly published post text and image feature embeddings (p_t, p_m) is made with each of the existing posts' in E ($e_{t,f}$, $e_{m,f}$). For each pairwise comparison, for both text $e_{t,f}$ and image

$e_{m,f}$ feature vectors, a cosine similarity score is generated with the results $s_{t,\alpha}$, $s_{m,\alpha}$ added to cosine similarity sets S_t and S_m, respectively. Next, a subset of text T_t and image T_m samples is selected from each cosine similarity set S_t, S_m, based on the highest respective cosine score, for example, where $T_t = S^{\sim}_{t,a_i} \cup \{max(S_t \setminus S_{t_i})\}$. In our experiment, for text, we have selected $i = 8$ as the upper limit of existing posts to forward to semantic text similarity analysis, for images as our aim to find the most similar image in all existing posts, we have used $i = 1$. In step 3, for each $s_t \in T_t$, we compute the semantic text similarity (STS) score $s_{t,\beta}$ (using our STS-b fine-tuned RoBERTa$_{LARGE}$ model) for all eight existing post text feature embeddings in T_t, adding these to the set $T_{t,\beta}$, forwarding the computed STS scores for cascade link threshold analysis. Step 4 represents the final processing step where the sets of subsampled STS scores $T_{t,\alpha}$ and image cosine similarity scores $S_{m,\alpha}$ are assessed by the post link threshold algorithm which evaluates whether the text and image similarity scores satisfy a predefined threshold for creating a cascade link. Here, θ_t, θ_m represent the link threshold for semantic text similarity and imagine cosine similarity, respectively. Based on our experiments, we have derived optimal θ for text and image similarity using a gridsearch during the RoBERTa$_LARGE$ and ResNet50 fine-tuning process. In step 4, the algorithm also checks if the existing subsampled post has a cascade ID $s_c \neq 0$, or not $s_c = 0$ (i.e., where 0 refers to the default cascade ID for singleton posts that have no cascade association). If the subsampled post's text and image similarity with the new post is equal to or above the required similarity threshold the subsampled post's is checked to see if it has an existing cascade ID assigned to it. If the the subsampled post has a cascade ID, the newly published post p_c, linking the newly published post to corresponding information cascade. Otherwise, a new cascade ID is created for both the new and subsampled post by selecting the next highest cascade number in the existing set of cascade IDs C queried from the post database, where $p_c = 1 + max_{c \in C}$. If no comparison threshold is satisfied, the newly published post is considered a singleton post and is assigned the default cascade ID $s_c = 0$. Note that in the case of STS and cosine score ties for the new post across multiple subsampled posts, time is used as a tiebreaker to ensure a single link is created for a post in any given information cascade.

In Figure 4 an example of the *Information Cascade Pipeline* output is shown for an identified cascade in our TNCD dataset. Here, the pipeline shows that it has linked primarily via semantic text similarity, where $\theta_t = 0.5$, as derived from the gridsearch optimisation, and $s_{t,\beta} \geq \theta_t$). Note that, should

Table 2: List of symbols for Information Cascade Pipeline

Variable	Definition
p_t	Raw text from post
p_m	Raw image from post
$p_{t,f}$	Extracted SRoBERTa$_{LARGE}$ text feature embeddings
$p_{m,f}$	Extracted ResNet50 image feature embeddings
E	Set of existing post text & feature embeddings ($e_{t,f}$, $e_{t,f}$) & cascade IDs (e_c)
$e_{t,f}$	Text feature embeddings for post $e \in E$
$e_{m,f}$	Image feature embeddings for post $e \in E$
S_t	Set of all text feature embeddings cosine scores $\forall e \in E$
S_m	Set of all image feature embeddings cosine scores $\forall e \in E$
$s_{t,\alpha}$	Text cosine similarity for post $s_t \in S_t$
$s_{m,\alpha}$	Image cosine similarity for post $s_m \in S_m$
T_t	Set of top i cosine scores for text $s_{t,\alpha}$ in set S_t
T_m	Set of top i cosine scores for text $s_{m,\alpha}$ in set S_m
$s_{t,\beta}$	Semantic textual similarity (STS) score for subsampled post $s_t \in S_t$
$T_{t,\beta}$	Subset of text semantic text similarity(STS) scores
s_c	Cascade ID for subsampled post $s \in S$
C	Set of all existing Cascade IDs
θ_t	Link threshold for text similarity
θ_m	Link threshold for image similarity
p_c	Assigned cascade ID for new post p

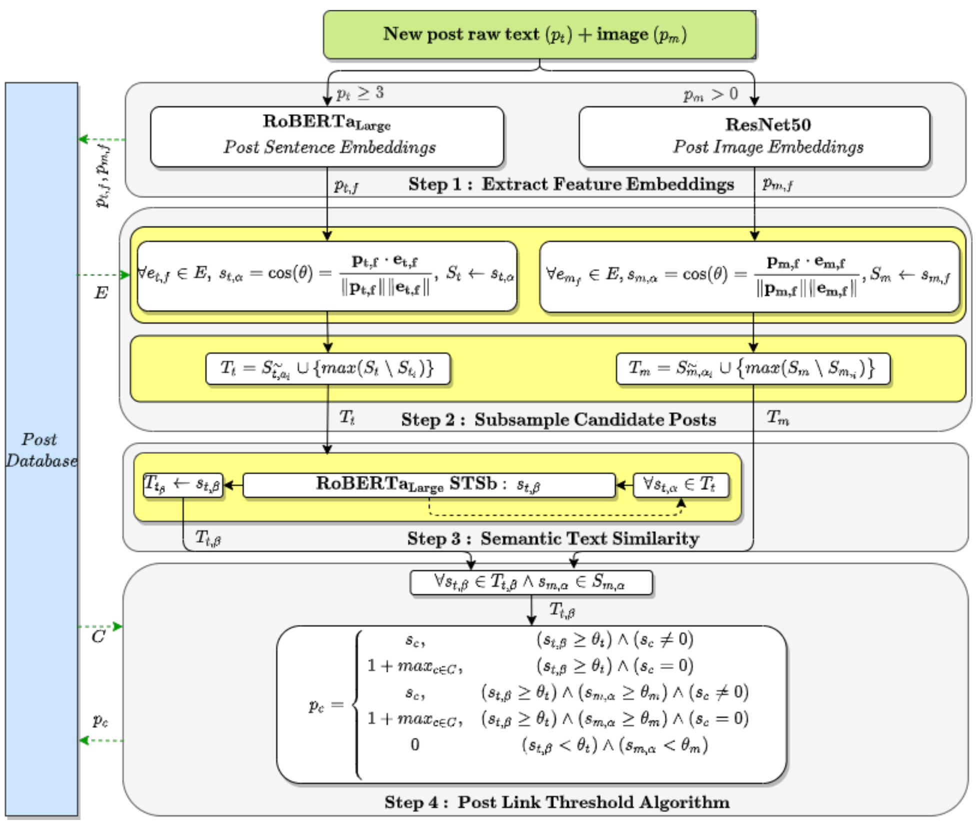


Figure 3: Information Cascade Pipeline

$s_{t,\beta} < \theta_t$ (0.5 in this case) for the fourth post in the case, the cascade pipeline would still have correctly linked the fifth post in the cascade, based on its image similarity cosine score.

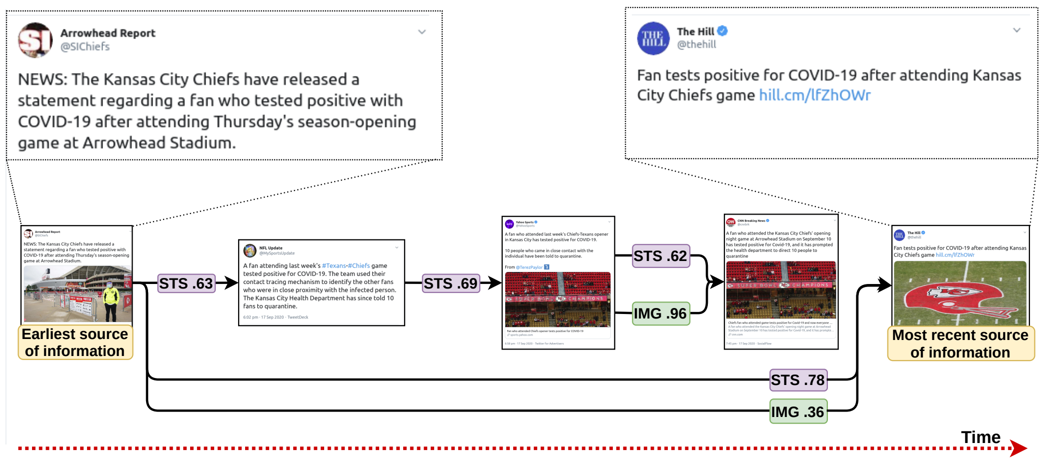


Figure 4: Example of Information Cascade Pipeline output for a identified cascade in the TNCD dataset

4. Experimental Analysis and Validation

4.1. Experiment methodology and testbed

For the experimental analysis of the Information Cascade Pipeline, we have pre-trained multiple models for text and image similarity, where each set of models was validated on publicly available datasets optimised for their respective inference tasks. The experiments were executed on a single computer workstation equipped with a NVIDIA GTX 1080 Ti GPU featuring 11gigabytes RAM, 3584 CUDA cores and a bandwidth of 484GB/s. We used the Python numpy library for matrix multiplication, Re library for text preprocessing (i.e., regular expression operations), emoji[3] library to convert emojis into text and Transformers[4] and Simple Transformers[5] frameworks for retraining and evaluating the RoBERTa model. In the case of TF-IDF,

[3]https://github.com/carpedm20/emoji
[4]https://github.com/huggingface/transformers
[5]https://github.com/ThilinaRajapakse/simpletransformers

Table 3: Details on TNCD

Parameters	Validation set	Test set
no. of posts	600	600
no. of posts in cascade	306	281
no. of cascades	57	60
no. of posts with images	579	599
min no. of posts in a cascade	2	2
max no. of posts in a cascade	12	13

we used the NLTK library[6] to remove English stop words and scikit-learn[7] to compute the features. To accelerate the tensor multiplications, we used the CUDA Toolkit with cuDNN, which is the NVIDIA GPU-accelerated library for deep neural networks.

4.2. TNCD dataset

To evaluate the performance of our approach we collected 1,200 news items posted on Twitter. We call this the Twitter News Cascade Dataset (TNCD). It contains posts (text and images) retrieved from sportsmen, politicians and news channels accounts, most from September 2020. We used the tweepy[8] library to access the Twitter API. The posts are human-annotated regarding whether they belong to a particular information diffusion cascade or not. Table 3 presents some of the characteristics of the created dataset. It is equally split into validation and test set, with each set containing 600 posts. This was done in order to tune the values of θ_t and θ_m (see next subsection). It should be noted that all posts contain text but not all contain images.

4.3. Performance evaluation

To assess the effectiveness of our *Information Cascade Pipeline* and demonstrate the usefulness of its hybrid text and image similarity detection model ensemble (using RoBERTa$_{large}$ for semantic text similarity, finetuned on the STS-b dataset), we have conducted a comparative analysis of the TNCD

[6]https://www.nltk.org/
[7]https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
[8]https://www.tweepy.org/

dataset across four different algorithms that could be applied in step 3 of Information Cascade Pipeline. Namely, different pipeline configurations for semantic text analysis which leverage 1) a standard pretrained SRoBERTa$_{LARGE}$ text similarity model (pretrained on the SNLI and MRPC datasets), 2) a pretrained RoBERTa$_{LARGE}$ text similarity model fine-tuned for paraphrase identification classification tasks using the MRPC dataset, 3) a TF-IDF feature extraction model using cosine similarity (based on work in presented in [17], and 4) pretrained RoBERTa$_{LARGE}$ text similarity model fined-tuned on the STS-B dataset. All of the above were evaluated, also by combining them with the ResNet50 image similarity model, (as well as with VGG16 combined with pretrained RoBERTa$_{LARGE}$ on the STS-B, for comparison) as part of the hybrid text and image cascade generation process. Each pipeline configuration was evaluated using the "Post Link Threshold Heuristic" defined in the Information Cascade Monitoring pipeline architecture (see Figure 3).

For evaluating each pipeline configuration's performance, we have selected the Fowlkes-Mallows index (FMI) [55], which is typically used to determine the degree of similarity between clusters of data points obtained via a clustering algorithm. Common evaluation metrics such as accuracy and F1-score used in classification are not applicable to clustering algorithms, or machine learning approaches which assign a group-based identity to data points, since their performance evaluation is not as simple as counting the number of false positives and false negatives, or the precision and recall. This is due to the fact that the evaluation metric should not consider the exact values of the cluster labels but rather check whether a cluster is comprised of similar data according to a set of ground truth labels. The FMI metric provides a suitable metric for measuring the performance of information cascade generation according to the confusion matrix analysis used in our experiment training and testing results (e.g., True Positive (TP) - post correctly linked to a cascade, True Negative (TN) - post correctly not added to a cascade, False Positive (FP) - post incorrectly added to a cascade, False Negative (FN) - post incorrectly not added to a cascade). This is because information cascades can be naturally grouped as clusters of interrelated data points. The FMI score itself is represented in a range from 0 to 1, where the higher the value the more similar the datapoints within a given information cascade:

$$FMI = \frac{TP}{\sqrt{(TP + FP)(TP + FN)}} \tag{4}$$

where TP depicts the true positives, i.e. the number of pairs of posts that belong to the same cascade in both the ground truth labels and the predicted ones), FP the false positives, i.e. the number of pairs of posts that belong to the same cascade in the true labels but not in the predicted labels, and FN the false positives, i.e. the number of pairs of posts that belong in the same cascade in the predicted labels and not in the true labels.

During the preprocessing phase, for the case of the transformer-based approaches we removed usernames (e.g., USER) and URLs, while the included emojis were "deemojified" into text (e.g., :smile). On the other hand, for the TF-IDF approach we removed also the English stop words from the posts' texts and punctuation before computing the TF-IDF features. Afterwards, to optimise the selection of text and image similarity threshold parameters θ_t and θ_m in the "Post Link Threshold Heuristic", we perform a grid-search of their parameters. In Figure 5, each heatmap illustrates the FMI score achieved for different text and image similarity cascade link thresholds parameters across each grid-search iteration. We have excluded the RoBERTa$_{LARGE}$ MRPC model from the best θ_t search since it is trained on a binary classification task, and as a result this threshold is already defined to 0.5. The best θ_t was 0.25 for TF-IDF, 0.5 for RoBERTa$_{LARGE}$ STS-B and 0.6 for USE and SRoBERTa$_{LARGE}$. For all approaches, the optimal θ_m was 0.9.

The evaluation of each pipeline semantic text similarity configuration (Table 4) shows that RoBERTa$_{LARGE}$ fine-tuned on MRPC achieves the lowest performance. This was expected as the MRPC dataset focuses on paraphrase classification rather than semantic text similarity. By comparison, SRoBERTa$_{LARGE}$ pretrained on the NLI and STS-B datasets model achieves a higher FMI score (validation +11.5% and test +9.66%), while the USE model reached even higher FMI scores, 84.06% and 84.03% for the validation and test set respectively. For RoBERTa$_{LARGE}$ fine-tuned on STS-B, the model outperforms RoBERTa$_{LARGE}$ MRPC by a FMI score of over 18%. This improvement in performance is reasonable given the problem definition of information cascade monitoring focuses on the semantic similarity between text (STS-B), and fine-tuning the model further on this dataset optimises its attention task towards semantic text similarity tasks. Moreover, the RoBERTa$_{LARGE}$ STS-B surpasses by over 7% the performance of TF-IDF-based approach presented in [17] on the validation set and over 4.5% for the case of the test set. Finally, our proposed text (RoBERTa$_{LARGE}$ STS-b) and image (ResNet50) ensemble detection model obtained the high-

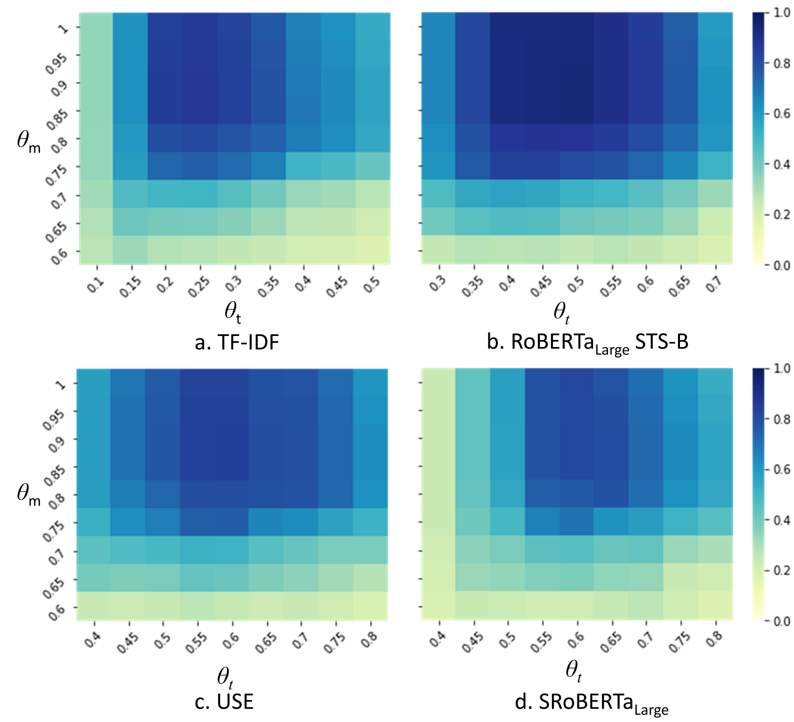


Figure 5: Heatmaps representing the influence of the θ_t, θ_m values to the obtained FMI on the: a. TF-IDF, b. RoBERTa$_{LARGE}$ STS-B, c. USE, and d. SRoBERTa$_{LARGE}$ on the validation set

est FMI score and incidentally provided the most accurate configuration for information cascade monitoring. We observe that including image similarity in the information cascade monitoring process has led to a meaningful performance benefit for all model configurations we have tested and for RoBERTa$_{LARGE}$ STS-b (validation: from 92.07% to 93.40%, test: 91.55% to 92.02%), which was the best performing model. Furthermore, we examined also the use of VGG16 embeddings obtaining almost identical scores with those of ResNet50 (validation: 93.40%, test: 91.96%); however, extracting embeddings in VGG16 is more computationally expensive (VGG16 has approximately five times the number of model parameters defined in ResNet50), which results in significantly increased execution latency (i.e., for 1,000 iterations the inference time per image is 0.117 ms for the VGG16 while only 0.052 ms for the ResNet50). Moreover, it is worth mentioning that by following a greedy approach (i.e., excluding sentence embedding-based subsampling) we

Table 4: Information cascade discovery performance

Model Integration	Validation FMI	Test FMI
TF-IDF[17] (text)	84.40%	86.80%
TF-IDF[17] (text) + ResNet50 (image)	86.00%	87.50%
USE (text)	82.10%	84.23%
USE (text) + ResNet50 (image)	84.06%	84.93%
RoBERTa$_{LARGE}$ MRPC (text)	66.41%	66.37%
RoBERTa$_{LARGE}$ MRPC (text) + ResNet50 (image)	69.92%	73.34%
SRoBERTa$_{LARGE}$ (text)	81.42%	81.15%
SRoBERTa$_{LARGE}$ (text) + ResNet50 (image)	81.42%	83.00%
RoBERTa$_{LARGE}$ STS-B (text)	92.07%	91.55%
RoBERTa$_{LARGE}$ STS-B (text) + ResNet50 (image)	**93.40**%	**92.02**%

obtained the same cascade FMI scores for both VGG16 and ResNet50 image similarity models when used in conjunction with our STS-B model. Therefore, for execution latency performance reasons alone, we selected ResNet50 as the image similarity deep learning architecture in our Information Cascade Pipeline.

To validate the performance of our heuristic algorithm which integrates the combination of text and image similarity for cascade link selection, we have performed an experimental comparison with related research by Sakaki et al., who proposed in [56] an alternative formula for combining text (linear SVM classifier over Bag of Words) and image (Scale-invariant feature transform with SVM) similarity models:

$$Score_{combined} = Score_{text} \times a + Score_{image} \times (1 - a) \tag{5}$$

where a is set as a ratio of the text score and an image score to combine two scores appropriately. The authors used a equal to 0.244. However, for the case of our dataset we found out that the best a is 0.95 and the $Score_{combined}$ term should be above or equal to 0.45 in order for a post to be included in a cascade. Table 5 presents the obtained results using ResNet50 and RoBERTa$_{LARGE}$ for image and text similarity respectively. Experimental results with our Twitter dataset reported that our heuristic algorithm outperforms the method proposed by Sakaki et al., which reported a validation FMI score of 92.64% and test score of 91.85%, compared to 93.40 and 92.02 for our approach, respectively. At the time of writing and to the best of our knowledge, there has been no study other than Sakaki et al.'s exploring the

Table 5: Performance of comparison of text and image integration heuristic algorithms

Integration Algorithm	Validation FMI	Test FMI
Sakaki et al. (2014) [56]	92.64%	91.85%
Our Method*	**93.40%**	**92.02%**

*See Figure 3 - Step 4

integration between post text and image similarity modelling in social media information cascade or diffusion analysis.

Figure 6 shows a tree-based representation of the information cascades identified. Here, black links represent TP connections in the cascade, while the red links represent FP connections in the cascade. As shown, the largest information cascade presented in our TNCD dataset is correctly identified to consist of 13 posts. Delving deeper into the predicted FP links (Table 6), we can observe that some can easily be confused as similar even by human annotators. The first example presented in Table 6 presents two posts that talk about the political relationship between the U.S and Iran, with the first mentioning that the U.N. sanctions against Iran have been restored, while the second one that they will be reimposed. The posts included in the second example pair refer both to fatal car accidents, and the street number included in the first post equals the age of the driver in the second post.

Table 6: False positive examples on the TNCD test set

Examples	STS score
1: The Trump administration has declared that all U.N. sanctions against Iran have been restored, a move most of the rest of the world rejects as illegal. **2**: U.S. says U.N. sanctions on Iran to be reimposed Saturday. What does that mean?	0.5386
1: The driver who died heading eastbound in a pickup truck on State Road 40 when the driver of a sport utility vehicle entered a curve and veered into the eastbound lane. **2**: The man in his 40s was fatally injured and pronounced dead at the scene	0.5177

In line with the previous state of the art [17], we evaluate the performance of the *Information Cascade Pipeline* with respect to its computation latency

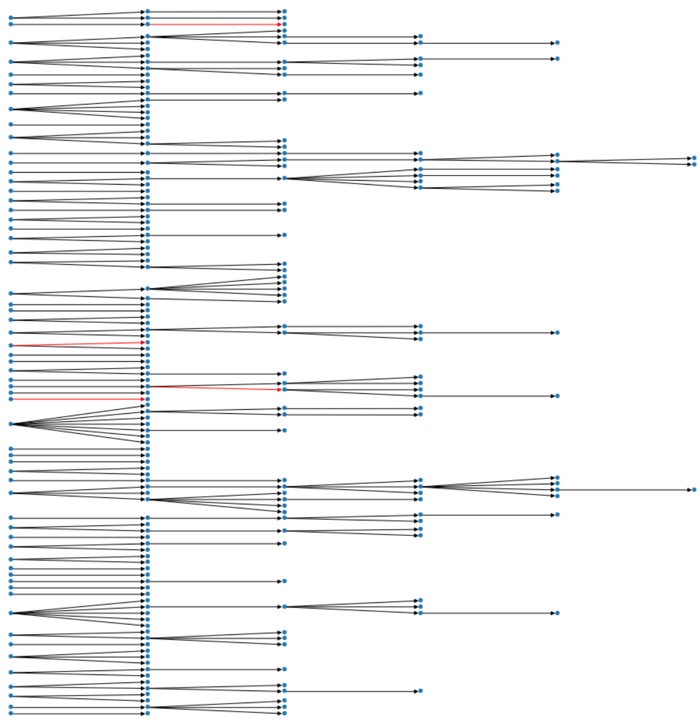


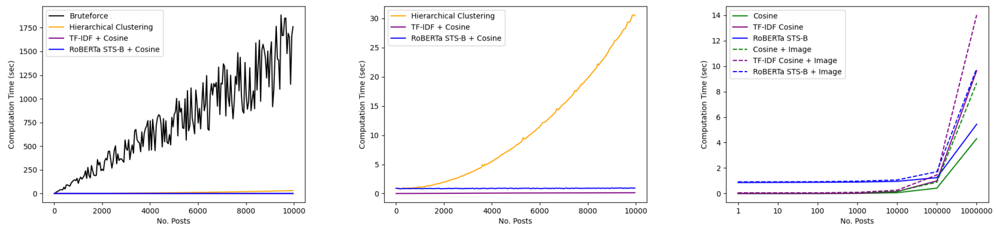
Figure 6: Created cascades in the TNCD test dataset (red links represent the false positive links)

when processing newly published posts on a social media platform. As our objective is to integrate the *Information Cascade Pipeline* into a real tool for supporting the assessment of information trustworthiness in social media (Section 5), our analysis takes into consideration the latency for information cascade analysis of each new post published. Therefore, here, processing latency represents the total processing time required to assess information cascade association for every new post. In Figure 7, we first compare the processing latency of a new post with all existing posts E, for up to 10,000 posts across three methods: 1) bruteforce (greedy) pairwise STS processing with no subsampling, 2) hierarchical clustering subsampling ([57]) + STS subset (subset $i = 8$), 3) cosine similarity subsampling mechanism + STS subset (subset $i = 8$), and 4) TF-IDF estimation followed by cosine similarity. In subfigure 7a, we observe an expected high linear increase in processing latency as the number of stored posts for brutefore comparison increases (approximately 29 minutes for 10,000 posts), whereas for clustering, cosine comparison (which includes a fixed STS subsample of eight posts) and TF-IDF, the processing latency is orders of magnitude lower and relatively stable as the number of stored posts increases. Subfigure 7b shows that hierarchical clustering also follows a relatively linear processing delay compared to cosine subsampling, albeit with significantly reduced processing time compared to STS bruteforce (approximately 30 s for 10,000 posts). In subfigure 7c, cosine subsampling takes approximately 4 s to process 1,000,000 posts. The results demonstrate that our *Information Cascade Pipeline* cosine similarity subsampling with a fixed-size STS subset, can support web scale analysis providing lower estimation time than the previous TF-IDF approach [17] above 10,000 examples. This is due to the fact that the estimation of TF-IDF index, similarly to that of the cosine similarity, increases as the number of stored posts increase, while the RoBERTa-based STS estimation is applied only to 8 posts, and is therefore constant. Subfigure 7c displays, also, the computational expense of including the image processor in our pipeline. Similarly to text similarity, finding similar images is based on applying cosine similarity over ResNet's embeddings so it is highly dependent on the number of stored posts. As result, the computational cost of including image similarity as well is almost twice as high when compared to ours text-based similarity approach, however, it is still reasonable; it is few milliseconds higher ($\approx$160ms) than using only the TF-IDF based text analysis, while having a much higher information cascade discovery performance ($\approx$7%). Moreover, it is worth highlighting that the estimation of TF-IDF requires updating the already estimated and stored TF-IDF in the database TF-IDF. By comparison, our method storage of text embeddings is static and does not require continuous updates. Note that this functional behaviour is not reflected in the plots, which display only the estimation times and not the transactions with the database.

5. Prototype implementation of the Information Cascade Pipeline mechanism

In this section, we provide an overview of our prototype *Information Cascade Pipeline* implementation on a private instance of the decentralised social media platform *Mastodon* created for the *EUNOMIA*[9] project. Figure

[9]https://eunomia.social



(a) (b) (c)

Figure 7: Performance evaluation of Information Cascade Pipeline computation time: a) text similarity only (including bruteforce); b) text similarity (no bruteforce); c) text Vs. text+image (no bruteforce)

8 is a high-level illustration of the *Information Cascade Pipeline* integration within the platform. Specifically, the information cascade monitoring prototype is an independent module which interfaces with *EUNOMIA*'s private Mastodon API to access posts' information, whilst receiving new published post content via the *EUNOMIA* services orchestrator. Our prototype implements a post analysis component which communicates with the internal post database, text and image similarity components. Here, the *Information Cascade Pipeline* described in Figure 3 is activated when a published post meets a predefined minimum word length for cascade processing.

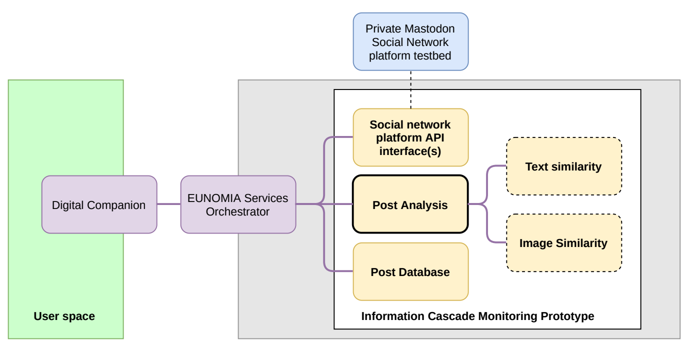


Figure 8: High level overview of the information cascade monitoring prototype within the the EUNOMIA system architecture

Figure 9 shows a screenshot of the prototype information cascade user interface, presented to the user as a side panel that is accessed via the "Show other similar posts" link shown under each post that belongs to a cascade. The *Information Cascade Pipeline* has identified an information cascade and

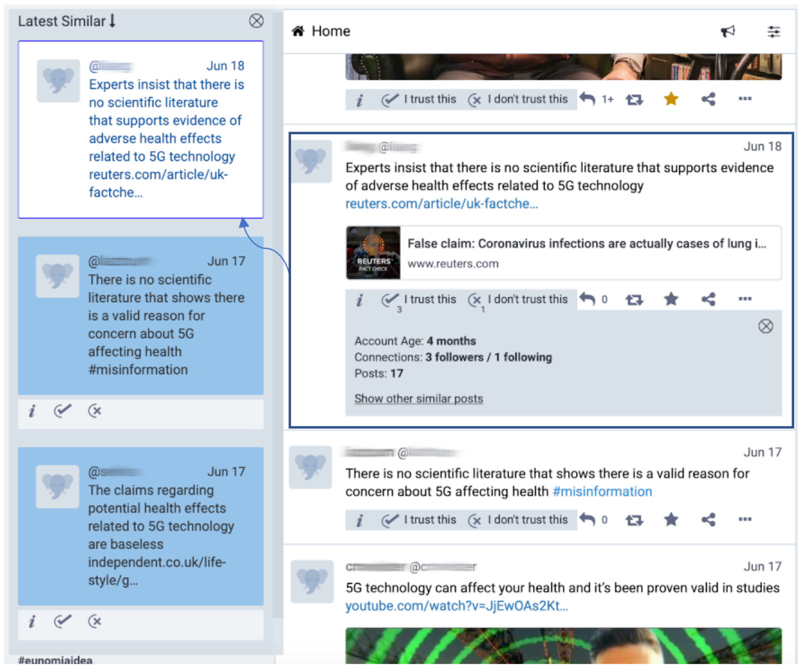


Figure 9: Screenshot of the information cascade as visualised to the EUNOMIA user

has ordered it chronologically, highlighting to the user the earliest and most recent posts in the cascade.

6. Conclusions

Identifying cases in social media where information has spread or been replicated by users, without them explicitly resharing it, is a complex task. Intelligent mechanisms capable of autonomously monitoring the implicit diffusion of information on social media can help analyse the true virality and spread of information as it propagates in real-time. Importantly, such mechanisms can help a user identify the provenance of information and how it may have changed over time. Here, we progressed beyond the state of the art in this direction by applying semantic as opposed to statistical similarity, as well as by incorporating also image similarity. This involved employing a transformer-based model and a deep Convolutional Neural Network for textual and image similarity respectively. In addition, our post subsampling approach was able to make our method applicable to real-world online social networks. We implemented and deployed our prototype in our own instance of the decentralized social media platform Mastodon. While we have found the prototype to already be practical, it is not able to re-evaluate the membership of posts in existing cascades. In particular, the similarity of orphan posts (not yet included in a cascade) should be re-estimated after a certain time. This would decrease false negatives, but would need to be performed in a manner that is scalable for a real-world social media platform. Also, larger transformer-based architectures could be exploited to increase the performance of semantic textual similarity and information fusion mechanisms extracting relational embeddings from text and image pairs, and in this way enable an end-to-end approach. We consider these as interesting directions for future research.

Acknowledgment

The work presented in this paper has been supported through the European Commission's H2020 Innovation Action programme under project EUNOMIA (Grant agreement No. 825171).

References

[1] J. Kim, B. Tabibian, A. Oh, B. Schölkopf, M. Gomez-Rodriguez, Leveraging the crowd to detect and reduce the spread of fake news and misinformation, in: Proceedings of Eleventh ACM International Conference on Web Search and Data Mining, ACM, 2018, pp. 324–332.

[2] S. Vosoughi, D. Roy, S. Aral, The spread of true and false news online, Science 359 (2018) 1146–1151.

[3] T. Porat, P. Garaizar, M. Ferrero, H. Jones, M. Ashworth, M. A. Vadillo, Content and source analysis of popular tweets following a recent case of diphtheria in Spain, European journal of public health 29 (2019) 117–122.

[4] Z. Jin, J. Cao, Y. Zhang, J. Zhou, Q. Tian, Novel visual and statistical image features for microblogs news verification, IEEE Transactions on Multimedia 19 (2017) 598–608.

[5] M. Choraś, K. Demestichas, A. Giełczyk, Á. Herrero, P. Ksieniewicz, K. Remoundou, D. Urda, M. Woźniak, Advanced machine learning

techniques for fake news (online disinformation) detection: A systematic mapping study, Applied Soft Computing 101 (2020).

[6] D. Camacho, A. Panizo-LLedot, G. Bello-Orgaz, A. Gonzalez-Pardo, E. Cambria, The four dimensions of social network analysis: An overview of research methods, applications, and software tools, Information Fusion 63 (2020) 88–120.

[7] S. Cavallari, E. Cambria, H. Cai, K. C.-C. Chang, V. W. Zheng, Embedding both finite and infinite communities on graphs [application notes], IEEE Computational Intelligence Magazine 14 (2019) 39–50.

[8] M. Li, X. Wang, K. Gao, S. Zhang, A survey on information diffusion in online social networks: Models and methods, Information 8 (2017) 118.

[9] K. Subbian, B. A. Prakash, L. Adamic, Detecting large reshare cascades in social networks, in: Proceedings of 26th International Conference on World Wide Web, 2017, pp. 597–605.

[10] W. Xie, F. Zhu, J. Xiao, J. Wang, Social network monitoring for bursty cascade detection, ACM Transactions on Knowledge Discovery from Data (TKDD) 12 (2018) 1–24.

[11] S. S. Singh, A. Kumar, K. Singh, B. Biswas, Lapso-im: A learning-based influence maximization approach for social networks, Applied Soft Computing 82 (2019) 105554.

[12] C. Ashley, T. Tuten, Creative strategies in social media marketing: An exploratory study of branded social content and consumer engagement, Psychology & Marketing 32 (2015) 15–27.

[13] A. Panizo-LLedot, J. Torregrosa, G. Bello-Orgaz, J. Thorburn, D. Camacho, Describing alt-right communities and their discourse on twitter during the 2018 us mid-term elections, in: Proceedings of International conference on complex networks and their applications, Springer, 2019, pp. 427–439.

[14] I. Taxidou, P. M. Fischer, Online analysis of information diffusion in twitter, in: Proceedings of 23rd International Conference on World Wide Web, 2014, pp. 1313–1318.

[15] I. Taxidou, T. De Nies, R. Verborgh, P. M. Fischer, E. Mannens, R. Van de Walle, Modeling information diffusion in social media as provenance with w3c prov, in: Proceedings of 24th International Conference on World Wide Web, 2015, pp. 819–824.

[16] S. Barbosa, R. M. Cesar-Jr, D. Cosley, Using text similarity to detect social interactions not captured by formal reply mechanisms, in: Proceedings of 11th International Conference on e-Science, IEEE, 2015, pp. 36–46.

[17] I. Taxidou, S. Lieber, P. M. Fischer, T. De Nies, R. Verborgh, Web-scale provenance reconstruction of implicit information diffusion on social media, Distributed and Parallel Databases 36 (2018) 47–79.

[18] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. S. Corrado, M. Hughes, J. Dean, Google's neural machine translation system: Bridging the gap between human and machine translation, ArXiv abs/1609.08144 (2016).

[19] S. Poria, I. Chaturvedi, E. Cambria, F. Bisio, Sentic lda: Improving on lda with semantic similarity for aspect-based sentiment analysis, 2016 International Joint Conference on Neural Networks (IJCNN) (2016) 4465–4473.

[20] E. Ekinci, S. I. Omurca, NET-LDA: a novel topic modeling method based on semantic document similarity, Turkish Journal of Electrical Engineering and Computer Sciences 28 (2020) 2244–2260.

[21] T. Mikolov, K. Chen, G. S. Corrado, J. Dean, Efficient estimation of word representations in vector space, CoRR abs/1301.3781 (2013).

[22] H. T. Nguyen, P. H. Duong, E. Cambria, Learning short-text semantic similarity with word embeddings and external knowledge sources, Knowl. Based Syst. 182 (2019).

[23] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of Conference on Empirical Methods in Natural Language Processing, 2014, pp. 1532–1543.

[24] A. M. Dai, Q. V. Le, Semi-supervised sequence learning, in: Proceedings of Advances in Neural Information Processing Systems 28, 2015.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of Annual Conference on Neural Information Processing Systems, 2017.

[26] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2019.

[27] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, in: Proceedings of International Conference on Learning Representations, 2019.

[28] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. R. Bowman, Glue: A multi-task benchmark and analysis platform for natural language understanding, in: Proceedings of EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, 2018, pp. 353–355.

[29] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 2019.

[30] D. M. Cer, Y. Yang, S. yi Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y.-H. Sung, B. Strope, R. Kurzweil, Universal sentence encoder, ArXiv abs/1803.11175 (2018).

[31] M. Iyyer, V. Manjunatha, J. L. Boyd-Graber, H. Daumé, Deep unordered composition rivals syntactic methods for text classification, in: Proceedings of ACL, 2015.

[32] F. Gelli, T. Uricchio, M. Bertini, A. D. Bimbo, Image popularity prediction in social media using sentiment and context features, in: Proceedings of 23rd ACM international conference, IEEE, 2015, pp. 907–910.

[33] J. Cheng, L. A. Adamic, P. A. Dow, J. Kleinberg, J. Leskovec, Can cascades be predicted?, in: Proceedings of 23rd international conference on World wide web, 2014, pp. 925–936.

[34] P. Qi, J. Cao, T. Yang, J.Guo, J. Li, Exploiting multi-domain visual information for fake news detection, in: Proceedings of IEEE International Conference on Data Mining, 2019, pp. 518–527.

[35] P. J. McParlane, Y. Moshfeghi, J. M. Jose, Nobody comes here anymore, it's too crowded; predicting image popularity on flickr, in: Proceedings of International Conference on Multimedia Retrieval, 2014.

[36] M. Meghawat, S. Yadav, D. Mahata, Y. Yin, R. R. Shah, R. Zimmermann, A multimodal approach to predict social media popularity, in: Proceedings of IEEE Conference on Multimedia Information Processing and Retrieval, 2018.

[37] J. Lv, W. Liu, M. Zhang, H. Gong, B. Wu, H. Ma, Multi-feature fusion for predicting social media popularity, in: Proceedings of 25th ACM international conference on Multimedia, 2017.

[38] A. Canziani, A. Paszke, E. Culurciello, An analysis of deep neural network models for practical applications, arXiv preprint (2017).

[39] S. R. Bowman, G. Angeli, C. Potts, C. D. Manning, A large annotated corpus for learning natural language inference, in: Proceedings of Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 632–642. doi:10.18653/v1/D15-1075.

[40] A. Williams, N. Nangia, S. Bowman, A broad-coverage challenge corpus for sentence understanding through inference, in: Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 1112–1122. doi:10.18653/v1/N18-1101.

[41] D. M. Cer, M. T. Diab, E. Agirre, I. Lopez-Gazpio, L. Specia, Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation, ArXiv abs/1708.00055 (2017).

[42] W. B. Dolan, C. Brockett, Automatically constructing a corpus of sentential paraphrases, in: Proceedings of Third International Workshop on Paraphrasing, 2005.

[43] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International journal of computer vision 60 (2004) 91–110.

[44] T. Ahonen, A. Hadid, M. Pietikainen, Face description with local binary patterns: Application to face recognition, IEEE transactions on pattern analysis and machine intelligence 28 (2006) 2037–2041.

[45] E. Karami, S. Prasad, M. Shehata, Image matching using sift, surf, brief and orb: Performance comparison for distorted images, in: Proceedings of Newfoundland Electrical and Computer Engineering Conference, 2015.

[46] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proceedings of IEEE Computer Society conference on computer vision and pattern recognition, IEEE, 2005, pp. 886–893.

[47] S. Appalaraju, V. Chaoji, Image similarity using deep cnn and curriculum learning, arXiv preprint abs/1709.08761 (2017).

[48] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[49] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proceedings of International Conference on Learning Representations, 2015.

[50] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems 25 (2012) 1097–1105.

[51] G. Palubinskas, Mystery behind similarity measures MSE and SSIM, in: Proceedings of IEEE International Conference on Image Processing, IEEE, 2014, pp. 575–579.

[52] L. Wang, Y. Zhang, J. Feng, On the euclidean distance of images, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (2005) 1334–1339.

[53] D. Huttenlocher, G. Klanderman, W. Rucklidge, Comparing images using the hausdorff distance, IEEE Transactions on Pattern Analysis and Machine Intelligence 15 (1993) 850–863.

[54] D. Sejal, T. Ganeshsingh, K. R. Venugopal, S. S. Iyengar, L. M. Patnaik, Acsir: Anova cosine similarity image recommendation in vertical search, International Journal of Multimedia Information Retrieval 6 (2017) 143–154.

[55] E. B. Fowlkes, C. L. Mallows, A method for comparing two hierarchical clusterings, Journal of the American statistical association 78 (1983) 553–569.

[56] S. Sakaki, Y. Miura, X. Ma, K. Hattori, T. Ohkuma, Twitter user gender inference using combined analysis of text and image processing, in: Proceedings of VL@COLING, 2014.

[57] P. Kasnesis, R. Heartfield, L. Toumanidis, X. Liang, G. Loukas, C. Patrikakis, A prototype deep learning paraphrase identification service for discovering information cascades in social networks, in: Proceedings of IEEE International Conference on Multimedia and Expo Workshops, 2020.