Detecting Symptoms using Context-based Twitter Embeddings during COVID-19

Roshan Santosh[1], Sharath Chandra Guntuku[1], H. Andrew Schwartz[2], Johannes Eichstaedt[3], Lyle Ungar[1]
[1]University of Pennsylvania [2]Stonybrook University [3]Stanford University
{roshansk@seas,sharathg@seas,ungar@cis}.upenn.edu,
has@cs.stonybrook.edu, eich@stanford.edu

Abstract

In this paper, we present an unsupervised graph-based approach for the detection of symptoms of COVID-19, the pathology of which seems to be evolving. More generally, the method can be applied to finding context-specific words and texts (e.g. symptom mentions) in large imbalanced corpora (e.g. all tweets mentioning #COVID-19). Given the novelty of COVID-19, we also test the proposed approach generalizes to the problem of detecting Adverse Drug Reaction (ADR). We find that the approach applied on Twitter data can detect symptom mentions much prior to their being reported by the Centers for Disease Control (CDC).

1 Introduction

The COVID-19 pandemic has interrupted many everyday behaviors. SARS-nCOV is a relatively new virus and gaps in knowledge persist about how it affects the body, and consequently, its symptoms and symptom severity. In the early phases of the pandemic, patients and providers in affected areas used social media to exchange information about symptoms and clinical treatment (Iacobucci, 2020; Stokes et al., 2020). While social media can be non-representative and contain misinformation (Singh et al., 2020), it provides an open forum for the public to share their perceptions, concerns, and understanding of health and science. The use of social media has increased dramatically (>20%) as individuals shelter in place (Venkatraman, 2020).

Social media could enable early symptom discovery for diseases such as COVID-19 where the pathology is not completely known and our knowledge of it is evolving (Del Rio and Malani, 2020). The most prominent symptoms such as fever, cough, and shortness of breath were known early on during the COVID-19 pandemic. However, others such as changes in smell/taste, body aches, and diarrhea were added later to the symptom list by the CDC (Grant et al., 2020).

Using social media to gather information on public health is a growing focus of research, with a special emphasis on discovering side effects of drugs (pharmacovigilence) (O'Connor et al., 2014), often using labeled datasets to build supervised machine learning models (Luo et al., 2017).

We propose a natural language processing framework to automatically detect emerging symptoms using Twitter data. Our approach is built on the hypothesis that by identifying token embeddings that capture the context of symptom mentions, new tokens used in a similar context can be identified through embedding similarity (Devlin et al., 2018). Our approach shares similarities with the idea of lexicon development (Etzioni et al., 2008; Bontcheva et al., 2013), which uses an unsupervised graph-based approach for the labeling new words given a few labeled words. However, the graph is initiated with words of interest that have already been identified.

Our method's focus on a specific context allows it to search through large imbalanced corpora to identify context-specific (e.g. symptoms) tweets. This differentiates it from previous works by (Wu et al., 2019; Mpouli et al., 2020) that identify domain specific lexicon. Further, the approach by (Wu et al., 2019) relies on a domain specific corpus and topic modeling to build a lexicon, which would require the construction of a symptom-specific COVID-19 corpus.

2 Method

As is the case with several applications involving creating word lists associated with a construct or topic (Das and Smith, 2012), symptom mentions associated with COVID-19 come in different forms and shapes - often difficult to curate in

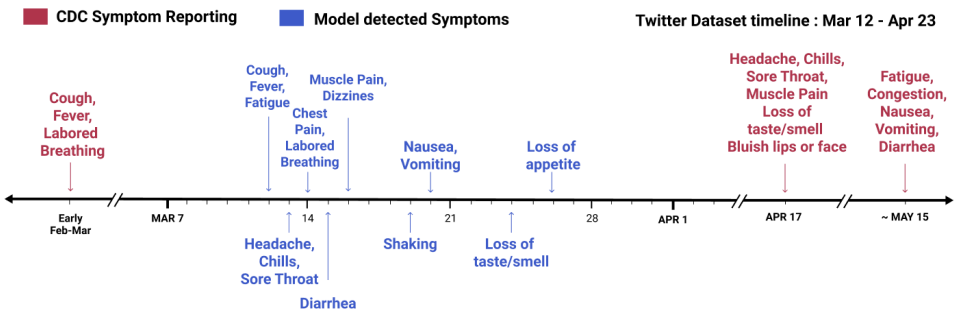

Figure 1: Comparative timeline of symptom detection by our approach against CDC reporting

advance (Rúa, 2007). The approach we propose assumes that we know at least one word of interest (i.e., a seed word) along with few corresponding seed texts where the seed word has been used in the desired context. For the case of emerging symptom detection, *cough*, a seed text could be '*I have a dry cough, chest pain and feeling lethargic as hell plus a headache*'.

2.1 Manual Context-Text Approach

Given the seed word and corresponding seed texts, BERT (*bert-base*) embeddings (Devlin et al., 2018) for the seed word are extracted from each of the texts. The BERT embedding for each token was computed by summing the hidden states of the last 4 layers of BERT. Individual embeddings from each of the seed texts are then averaged to generate a representative embedding for the seed word. We use 5 seed texts that capture part of the considerable variance associated with the symptom context.

Using the representative embedding for the seed word, an exhaustive search is performed across the dataset at a token level to identify the tokens that are most similar to the seed word, where similarity is measured using cosine similarity (one minus cosine distance). All tokens with a similarity value less than a minimum threshold (set empirically at 0.3) are excluded. Similarity scores of all occurrences of a given word are averaged.

2.2 Graph-based Iterative Training Approach

The previous model required text for every new seed word and didn't allow multiple runs with different seeds to learn from each other. To address this, we propose an iterative trainable search model that develops a similarity-based word graph. The model retains the search methodology of the earlier approach, but also includes a graph element and a trainable search parameter that improves the detection of context-specific words with increased iterations.

The directed and weighted word graph of the model represents the connections (based on similarity) between tokens. Each node in the graph corresponds to a word and is characterized by the representative embedding of the word. The edges have weights corresponding to the similarity score between the connected words (nodes). The second component of the model is the so-called 'Context Embedding', which represents the trainable parameter of the model. The context embedding is conceptualized to be an embedding vector that represents the specific context that we are interested in. Initialized by the representative embedding of the seed word, the context embedding incorporates embeddings from other words over iterations, to develop into a more robust representation of the specified context.

2.3 Algorithm

Initialization Initialize graph **G** by setting the root node with the representative embedding of the seed word. Initialize a queue **Q** by adding the seed word to it. The context embedding **CEmb** is also initialized to the representative embedding of the seed word. **CEmb** $\leftarrow Emb\{\text{Seed word}\}$, where $Emb\{x\}$ denotes the representative embedding of token x.

Procedure The specific steps used in the algorithm are as follows:

1. Pop next word from **Q**, denoted by t. Initialise a new node in **G** corresponding to t and set the node embedding to $Emb\{t\}$.
2. Initialise the query embedding q as $q \leftarrow k * \textbf{CEmb} + (1-k) * Emb\{t\}$.
3. Iterate through all tokens in the data, comparing their embeddings against the query

embedding q. All tokens with similarity less than the minimum similarity threshold $minSimThresh$ are dropped.

4. Select the top n words based on their similarity to q. Add these words to **Q**. Instantiate new nodes (if one doesn't already exist) for these words in **G** and add outgoing edges from t to these new nodes.
5. If all words for a given depth are explored, the top m words corresponding to that depth are selected based on similarity to **CEmb**. The context embedding is then updated by averaging $CEmb$ with the representative embeddings of the selected words, as shown in Eq 1

$$\textbf{CEmb} \leftarrow \frac{\textbf{CEmb} + \sum_{i=0}^{m} \text{Emb}\{x_i\}}{m+1} \quad (1)$$

6. Stop iterations when either **Q** is empty or when the maximum depth $maxDepth$ of **G** is achieved. Otherwise, repeat from Step 1.

3 Experiments

3.1 Manual Context-Text Approach

We tested our approach on a Twitter dataset containing tweets related to COVID-19, collected between March 12 to April 23 using the #COVID-19 tag. Our experiments were run on a random subset of this dataset containing 1 Million tweets.

Table 1: Words returned (and their similar scores) by our approach

Seed : Cough (Manual)		Seed : Cough (Graph model)	
Word	**Sim.**	**Word**	**Sim.**
fever	0.67	vomiting	0.84
throat	0.62	fever	0.83
##tis (pneumonitis)	0.61	throat	0.82
headache	0.61	congestion	0.75
nose	0.59	headache	0.71
breathing	0.58	coughing	0.70
congestion	0.57	asthma	0.69
##itis (bronchitis)	0.57	##itis (bronchitis)	0.64
taste	0.55	nausea	0.63
##raine (migraine)	0.54	##hea (diarrhea)	0.45

Our objective for this dataset is to be able to identify new symptoms of COVID-19 mentioned in tweets. We ran tests with *cough*, an established symptom, as the seed word. Seed texts (tweets) were selected where *cough* was used as a symptom to ensure that the correct context is captured. Top 10 results are shown in Table 1. With just a single seed word and corresponding text as input, the model could identify key symptoms of COVID.

3.2 Graph-based Iterative Training Approach

We evaluate our graph-based approach on 2 different datasets, with each dataset having a different context - namely COVID-19 Symptom Detection and Adverse Drug Reaction Identification.

3.2.1 COVID-19 Symptom Detection

We repeat our tests on the COVID-19 Twitter dataset on a subset of 1M tweets. We use *cough* as the seed word, while k is 0.3, $maxDepth$ is 3 and n is 5. The resulting graph from our model is shown in Figure 2. The size of the nodes represent the number of occurrences of a token as a symptom while the color intensity of the nodes represent the similarity values computed for the node during the graph building process.

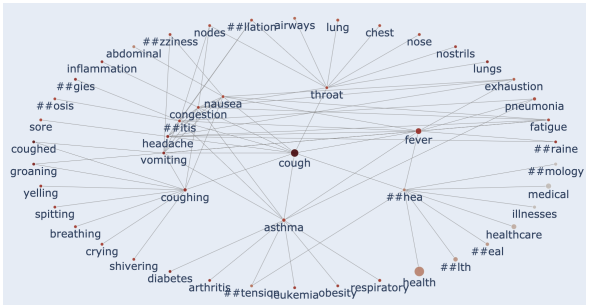

Figure 2: Symptom model graph for COVID-19 Tweet dataset

We observe that the model identified a wide range of symptoms ranging from common symptoms like fever, fatigue to less common ones like headache, vomiting, (chest) congestion, nausea, mig-##raine.

Evaluation Though a quantitative evaluation of our approach is not straightforward, we evaluate our approach by computing the precision in detecting correct words that fit the specified context.

For the problem of symptom detection, precision is calculated as the percentage of the actual symptoms detected by our model. Given that our model outputs a ranked list of words, precision is computed by looking at the top p results, where p represents the threshold for computing precision (Table 2).

Through a manual inspection of the top 100 results, rare to-be-confirmed symptoms like eye irritation, vertigo, anemia were detected. This marks a key utility of our approach as it helps generate

Table 2: Precision for Symptom Detection

Model	Seed word	Precision		
		p = 5	p = 10	p = 20
Manual	cough	0.8	0.9	0.8
Manual	fever	1.0	0.9	0.75
Manual	fatigue	0.8	0.8	0.75
Graph	cough	1.0	0.9	0.9

potential symptom candidates which can guide further evaluation.

3.2.2 Adverse Drug Reaction (ADR) Detection

For the second dataset, we use an annotated ADR dataset (Sarker and Gonzalez, 2015), where 13% of the tweets are labeled as ADR. The objective of this dataset is the identification of words denoting adverse drug reactions. Therefore, the specific context that we are interested in capturing is different from the previous dataset where the context of interest was the identification of symptoms of a disease. By testing our model on this dataset, we also test the ability of our approach to generalize to new tasks.

For the experiment, the seed word used is *pain*. k is 0.2, $maxDepth$ is 3 and n is 5. The resulting graph from our model is shown in Figure 3.

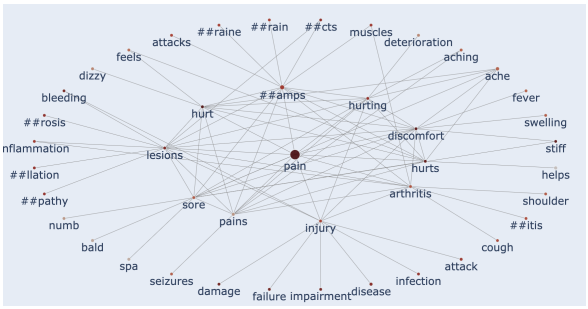

Figure 3: ADR model graph for Statins dataset

Some of the key ADR identified include inflammation, bleeding, muscle (pain), (skin) lesions, tremors, discomfort, and (calcium) deposits.

Evaluation Similar to COVID-19 symptom detection evaluation, we evaluate the model's performance for ADR detection, where a positive word represents an adverse reaction to a drug (Table 3).

Table 3: Precision for ADR Detection

Model	Seed word	Precision		
		p = 5	p = 10	p = 20
Manual	pain	1.0	0.9	0.8
Graph	pain	1.0	0.9	0.95

4 Discussion

The COVID-19 pandemic evolved in a global climate of confusion and uncertainty. The professional and lay public alike speculated on disease course, severity, and symptoms. The COVID-19 symptoms first observed appeared to be largely non-specific to COVID-19 (e.g., cough, fever). Finding COVID-specific symptoms (such as the sudden loss of the sense of smell) is important and potentially of clinical significance as large populations are being risk-assessed. The "digital exhaust" of social media encodes informal case reports of symptoms and discussions of media content about the virus alike. In principle, it could allow for the generation of a "master list" of COVID-19 symptom candidates, which the public health and medical community can, in turn, consider for further evaluation as COVID-specific markers (Chan et al., 2020).

In this study, we present a unsupervised learning approach to generate such a "master" list of COVID-19 symptoms, using the identification of words matching a specific symptom context. Through a preliminary evaluation, our approach shows high sensitivity in detected words. The current drawbacks of our approach include the inability to detect multi-word phrases as well as slow processing time. As can be seen in Figure 1, the approach detected headache, chills, sore throat, diarrhea, and other symptoms around a month before the CDC reported them. Given the novelty of COVID, the current method is hard to evaluate. We, therefore, considered the approach in the more studied context of detecting adverse drug reactions and show that the approach generalizes to this domain.

The approach relies on the Context Embedding contribution parameter (k). By varying k, we observe a phenomenon analogous to 'Exploration vs Exploitation' (Coggan, 2004): which, in principle, means that this method can be calibrated for different use cases. In the early phase of the disease, for example, a low k parameter may be chosen to aid in the generation of symptom candidates to be considered in light of the emerging clinical literature on COVID-19 and known physiological and biological interactions in the human body. A high k may be chosen to yield the subset of COVID-19 symptoms that are more robustly associated with the disease, at the cost of missing infrequent (albeit potentially specific) disease markers.

References

Kalina Bontcheva, Leon Derczynski, Adam Funk, Mark A Greenwood, Diana Maynard, and Niraj Aswani. 2013. Twitie: An open-source information extraction pipeline for microblog text. In *Proceedings of the international conference recent advances in natural language processing RANLP 2013*, pages 83–90.

AKM Chan, CP Nickson, JW Rudolph, A Lee, and GM Joynt. 2020. Social media for rapid knowledge dissemination: early experience from the covid-19 pandemic. *Anaesthesia*.

Melanie Coggan. 2004. Exploration and exploitation in reinforcement learning. *Research supervised by Prof. Doina Precup, CRA-W DMP Project at McGill University*.

Dipanjan Das and Noah A Smith. 2012. Graph-based lexicon expansion with sparsity-inducing penalties. In *Proceedings of the 2012 conference of the North American chapter of the Association for Computational Linguistics: human language technologies*, pages 677–687.

Carlos Del Rio and Preeti N Malani. 2020. Covid-19—new insights on a rapidly changing epidemic. *Jama*, 323(14):1339–1340.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S Weld. 2008. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74.

Michael C Grant, Luke Geoghegan, Marc Arbyn, Zakaria Mohammed, Luke McGuinness, Emily L Clarke, and Ryckie Wade. 2020. The prevalence of symptoms in 24,410 adults infected by the novel coronavirus (sars-cov-2; covid-19): A systematic review and meta-analysis of 148 studies from 9 countries. *Available at SSRN 3582819*.

Gareth Iacobucci. 2020. Covid-19: diabetes clinicians set up social media account to help alleviate patients' fears. *BMJ*, 368:m1262.

Yuan Luo, William K Thompson, Timothy M Herr, Zexian Zeng, Mark A Berendsen, Siddhartha R Jonnalagadda, Matthew B Carson, and Justin Starren. 2017. Natural language processing for ehr-based pharmacovigilance: a structured review. *Drug safety*, 40(11):1075–1089.

Suzanne Mpouli, Michel Beigbeder, and Christine Largeron. 2020. Lexifield: a system for the automatic building of lexicons by semantic expansion of short word lists. *Knowledge and Information Systems*, pages 1–21.

Karen O'Connor, Pranoti Pimpalkhute, Azadeh Nikfarjam, Rachel Ginn, Karen L Smith, and Graciela Gonzalez. 2014. Pharmacovigilance on twitter? mining tweets for adverse drug reactions. In *AMIA annual symposium proceedings*, volume 2014, page 924. American Medical Informatics Association.

Paula López Rúa. 2007. Keeping up with the times: lexical creativity in electronic communication. *Lexical Creativity, Texts and Contexts/ed. by J. Munat.–Amsterdam*, pages 137–159.

Abeed Sarker and Graciela Gonzalez. 2015. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *Journal of biomedical informatics*, 53:196–207.

Lisa Singh, Shweta Bansal, Leticia Bode, Ceren Budak, Guangqing Chi, Kornraphop Kawintiranon, Colton Padden, Rebecca Vanarsdall, Emily Vraga, and Yanchen Wang. 2020. A first look at covid-19 information and misinformation sharing on twitter. *arXiv preprint arXiv:2003.13907*.

Daniel C Stokes, Anietie Andy, Sharath Chandra Guntuku, Lyle H Ungar, and Raina M Merchant. 2020. Public priorities and concerns regarding covid-19 in an online discussion forum: Longitudinal topic modeling. *Journal of general internal medicine*, page 1.

A Venkatraman. 2020. Weekly time spent in apps grows 20% year over year as people hunker down at home. *App Annie*.

Sixing Wu, Fangzhao Wu, Yue Chang, Chuhan Wu, and Yongfeng Huang. 2019. Automatic construction of target-specific sentiment lexicon. *Expert Systems with Applications*, 116:285–298.