Automated Misinformation Detection with Natural Language Processing and Network Science Models

Ian Kloo

Thesis Proposal

April 30, 2024

Committee

Dr. Kathleen M. Carley, Chair

Dr. Brandy Aven

Dr. Hong Shen

COL Dave Beskow

Carnegie Mellon University

Agenda

- Background and Motivation (3 – 7)
- Detection Methodology Overview (8 – 11)
- Detection Study Details (12 – 29)
- Validation Study Overview (30 - 31)
- Research Workload and Timeline (32 – 33)
- Limitations and Boundary Conditions (34 – 36)
- Notable Similar Work (37 – 38)
- Conclusions (39 - 40)

Major questions and motivation

- **Q1: How can we detect misinformation in large datasets at the post level in a way that is useful for social cybersecurity research?**
- **Q2: How is misinformation used in information operations?**
  - **Q2.1: Does the base rate of misinformation use differ between the BEND maneuvers?**
  - **Q2.2: Do different actors employ misinformation differently in their maneuver sets?**
  - **Q2.3: Is misinformation linked to more effective maneuvers?**
- Why does this matter?
  - No existing methodology to detect misinformation that is scalable and agnostic to topic and platform(Q1).
  - Misinformation is used by those maneuvering in information space, but previous studies have not explored how this tactic is employed within the BEND framework (Q2).

How this fits within the existing social cybersecurity research context

- BEND provides a framework for describing information space maneuvers according to their intents (and, soon, their effects)
- Key components of BEND analyses:
  - Content:
    - Topics (***what*** are they discussing?)
    - Stance (***what side*** of the issue?)
    - Linguistics (***how*** are they talking about it?)
  - Tactics:
    - Bots (***how*** are they employed?)
    - **Misinformation** (***how*** is it incorporated in maneuvers?)

How this is different from fact-checking

- Fact-checkers determine the truth value of a message using one of:
  - Manual adjudication of claims
  - Semi-automated techniques that detect claims that are “check-worthy” and send them to human adjudicators
  - Fully-automated techniques that check “check worthy” claims against repositories of facts
- Our system detects potential misinformation based on the linguistic content of a claim, the conversation surrounding a claim, and the inferred reliability of the author
  - Fully automated
  - Unbiased by human fact-checkers or information repositories
  - Leverages both network science and NLP approaches

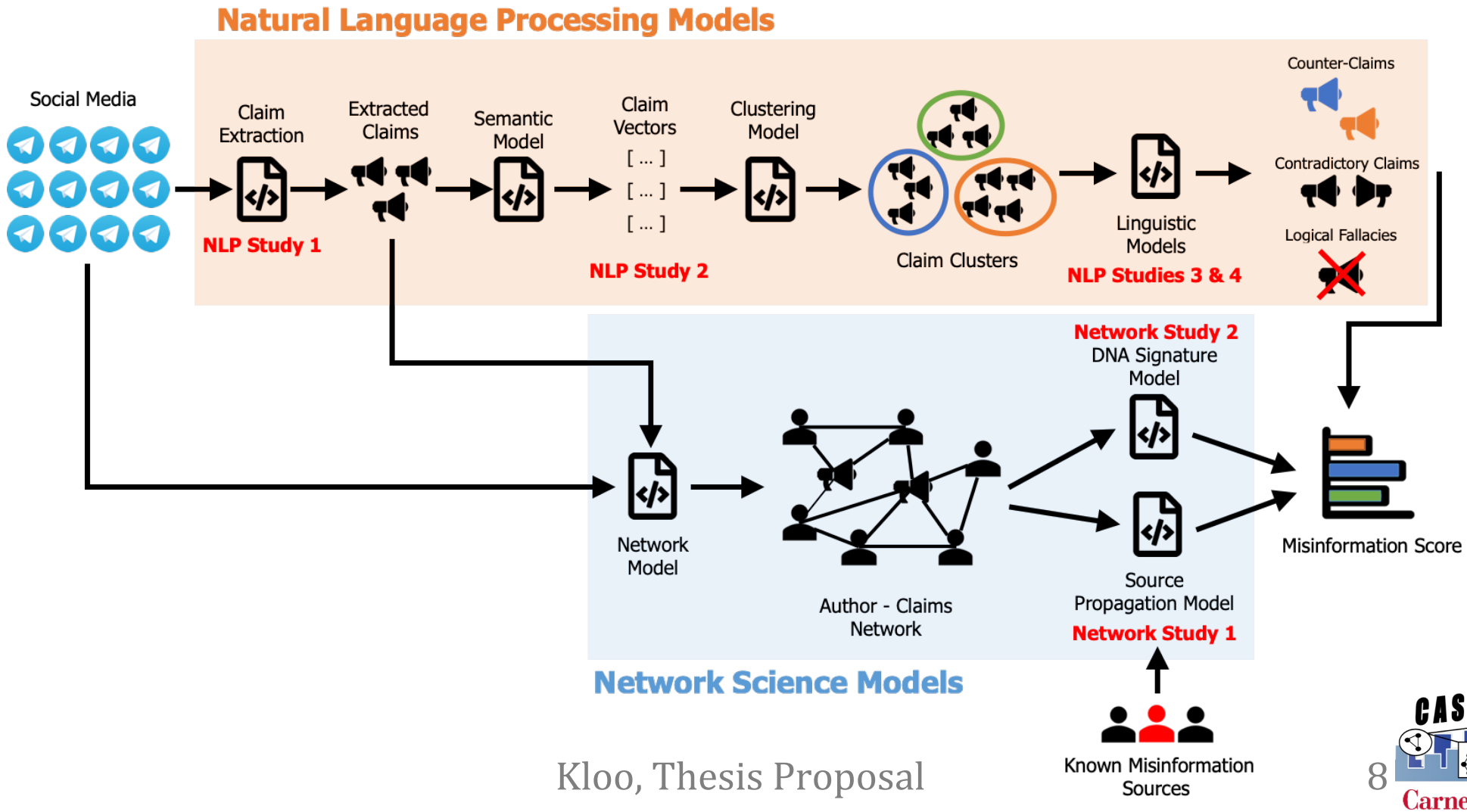
Key Definitions

- Claim = an assertion of the verifiable truth of something.
  - Examples of claims:
    - Masks don't prevent COVID-19 infections.
    - Russia invaded Ukraine to protect ethnic Russians.
  - Examples of statements that are not claims:
    - Nobody should be forced to wear a mask.
    - Russia's invasion of Ukraine is sad.
- Misinformation = false information spread, **regardless of the intent to mislead**
- Disinformation = false information deliberately spread to deceive people
  - A subset of misinformation with a known motive
- Propaganda = information, ideas, or rumors deliberately spread widely to help or harm a person, group, movement, institution, nation, etc.
  - Could contain misinformation (or disinformation), but could also be completely truthful
- Conspiracy Theory = a theory that rejects the standard explanation for an event and instead credits a covert group or organization with carrying out a secret plot

What this System will Attempt to Detect

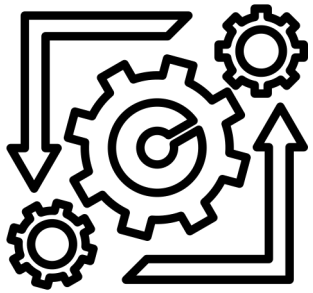
- In scope:
  - Misinformation, and therefore, disinformation – but we will not attempt to differentiate these by detecting intent to deceive.
  - The subset of propaganda that employs misinformation.
  - The (majority) subset of conspiracy theories that employ misinformation.
- Out of scope:
  - Propaganda that employs only truthful information.
  - Conspiracy theories that only rely on factual information.

Misinformation Detection Pipeline



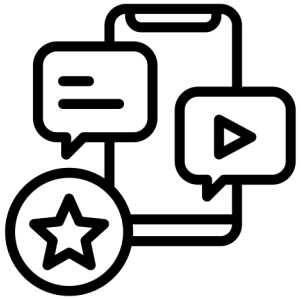
Detection Pipeline Requirements

Scalable/fast enough to run on a laptop (with reasonable data size)



Integrates with ORA, Netmapper, and Botbuster

- Either work seamlessly along these tools or be integrated directly into one of them (e.g., as an ORA report)



Agnostic to specific social media platforms

- Can leverage, but not rely on features specific to one platform (e.g., hashtags, retweets)

Theory – How Counter/Contradictory Claims Indicate Potential Misinformation

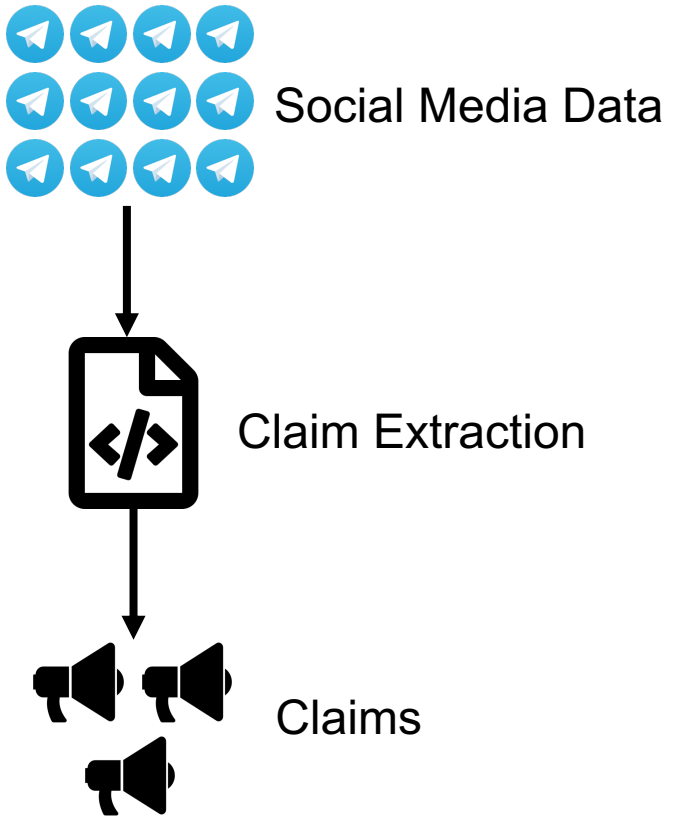
- Counter and contradictory claims **negate** each other, so there are two possibilities:
  - Both claims are false
  - One claim is true and the other is false
- Both claims cannot be true, so at least one is misinformation
- Logical fallacy detection and network approaches indicate which claim(s) is/are likely misinformation

Overview of Study Goals

- NLP
  1. **Claim Detection and Extraction** → find and extract claims from social media
  2. **Claim Clustering** → cluster similar claims so they can be compared
  3. **Counter/Contradictory Claim Identification** → compare similar claims to find counter and contradictory claims, indicating potential misinformation
  4. **Logical Fallacy Detection** → detect logical fallacies in claims to indicate misinformation
- Network
  1. **Unreliable Source Propagation** → determine likely sources of unreliable information with network propagation, indicating misinformation
  2. **Dynamic Network Signature Model** → determine network signatures that indicate misinformation

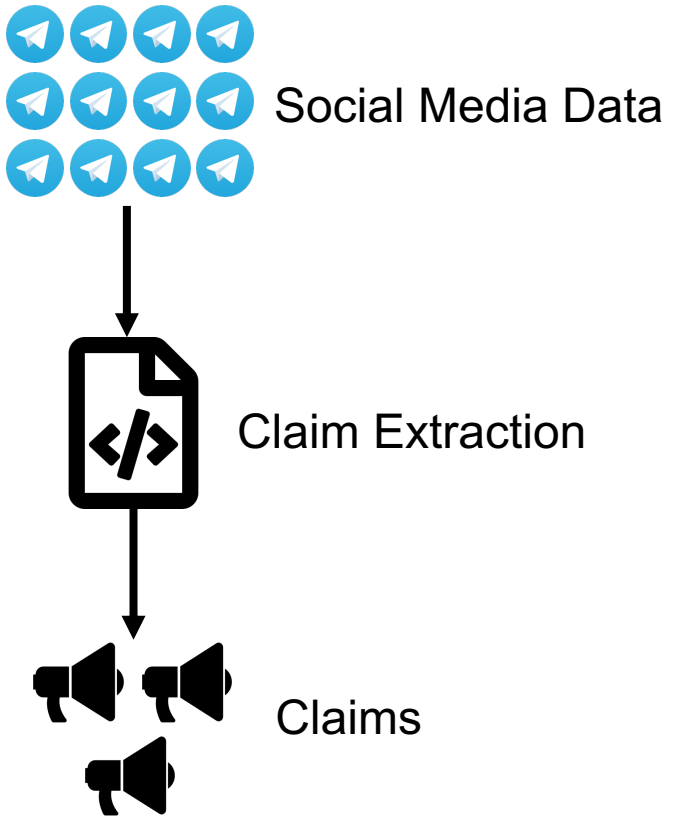
NLP 1: Claim Extraction (Overview)

- Goal = unsupervised identification and extraction of claims from social media text
- Method overview:
  - A two-layer modeling approach:
    1. Classify social media messages as containing or not containing a claim.
    2. Extract the claim span from the message.
- Notable existing work:
  - Rarely applied to social media
  - SemEval-2023 task focused on Twitter, but only on **causal** claims related to **medicine**
  - CheckThat! 2019 and 2021 tasks focused on the **check-worthiness** of **COVID-19 and political** claims on Twitter, but this disregards claims that cannot be fact checked and those in other domains
- Contribution = expand claim detection to generalize across topics and on multiple social media platforms (not just Twitter).



NLP 1: Claim Extraction (Data)

- Requirements
  - Multiple social media platforms (to be able to argue that this system generalizes across platforms)
  - >= 3 topics (to be able to argue that this system generalizes to unseen topics)
- Twitter → existing data, but need to expand
  - CheckThat! Twitter data contains 1,000 Tweets
  - Human-labeled, well-defined protocol using 5 annotators
  - COVID-19 and US politics are the only topics
- Telegram and Reddit → need to build
  - Self-labeled data from Telegram and Reddit following the same labeling protocol.
- See slide 25 for more information about data preparation.



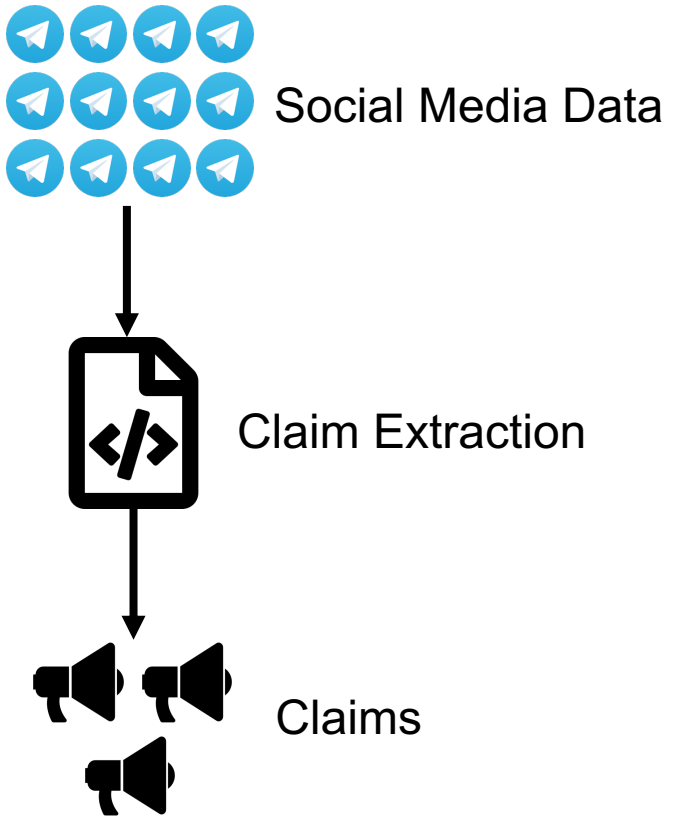
NLP 1: Claim Extraction (Message Classification)

- Input = social media text
- Output = binary label (contains claim or not)
- Training Data:
  - Text → binary label (contains claim or not)
- Modeling:
  - Stage 1: **task-tune** several transformer models (distilBERT, BERT, RoBERTa, distilRoBERTa) to classify messages that contain claims, select best model.
  - Stage 2: domain-adaptive pre-training (**DAPT**) on social media data to improve classification performance.
  - Stage 3: ensemble multiple models to create **voting classifier**, evaluate computation cost/benefit for any performance increase

Models	Levels	Values Used
Transformer-based Encoder Models	4	BERT, RoBERTa, distilBERT, distilRoBERTa
Hyperparameters	**Levels**	**Values Used**
Batch size	2	16, 32
Learning rate	4	2e-5, 3e-5, 4e-5, 5e-5
Number of epochs	4	2, 3, 4, 5
Total Experiments	**128**	

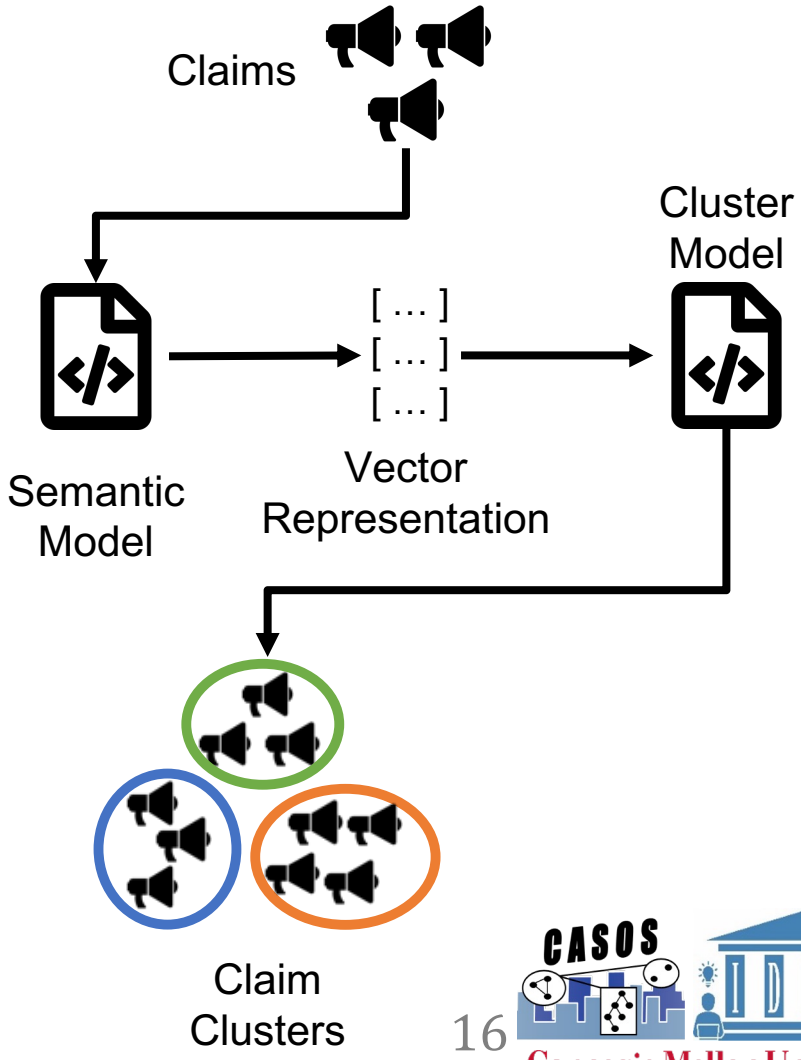
NLP 1: Claim Extraction (Sequence Tagging)

- Input = social media text containing a claim
- Output = span of text (sequence) with only the claim
- Training Data:
  - Text with known claim → labeled text span with only the claim
- Modeling:
  - Stage 1: **task-tune** several **transformer** models (distilBERT, BERT, RoBERTa, distilROBERTa) to extract claim spans using **token labeling**. If message classification study shows DAPT improvements, can re-use those DAPT models here.
  - Stage 2: **task tune** encoder-decoder model (T5) to test against the transformer baseline.
  - Stage 3: **prompt-engineer** for few- and zero-shot learning with generative LLMs to test performance.
  - Stage 4: evaluate potential **ensemble approach**, with the understanding this will likely be computationally infeasible (and tends not to work as well with span extraction)



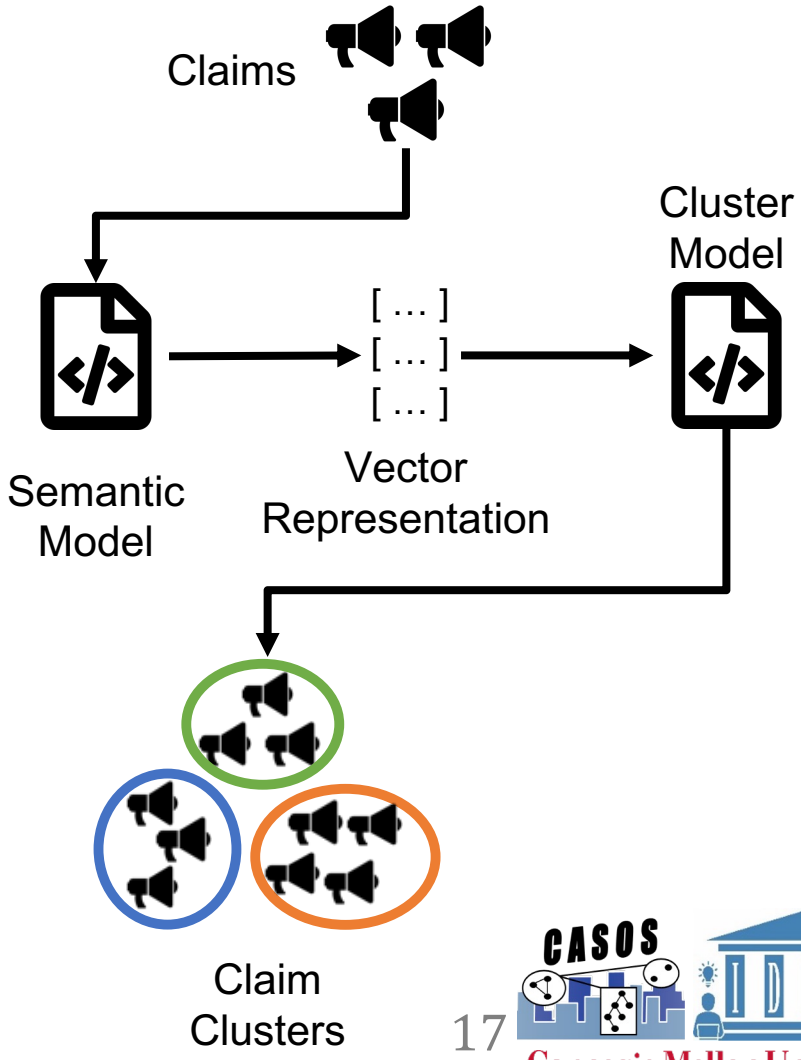
NLP 2: Claim Clustering (Overview)

- Goal = group claims that are about similar topics
- Data = same as previous study
- Method overview:
  - R1: Directly cluster claims by converting into vector space using semantic similarity and clustering.
  - R2: Extract entities from claims, create a bipartite graph of claims to entities, and cluster.
- Notable existing work:
  - BERTopic – generalized topic detection using semantic embeddings
  - SciClops – clustering scientific claims using hybrid embedding/network methodology
- Contribution = unsupervised claim clustering that generalizes across topics.



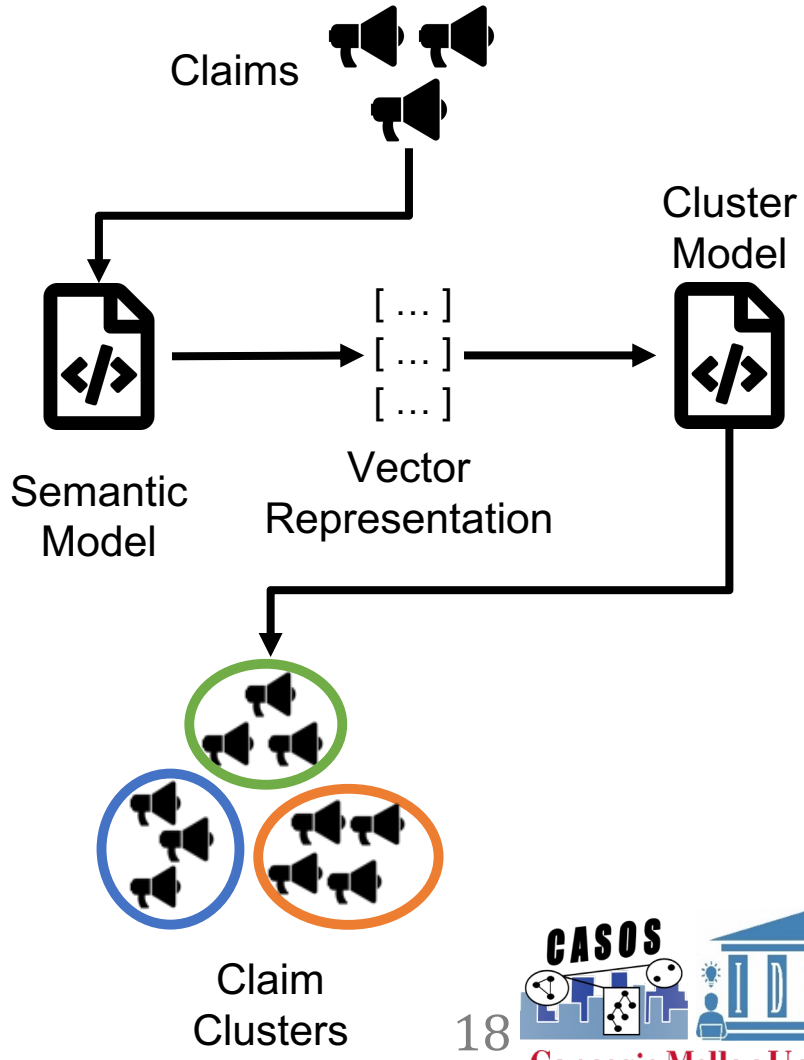
NLP 2: Claim Clustering (Semantic Clustering)

- Input = set of claims
- Output = clustered groups of claims based on topic
- Modeling:
  - Stage 1: represent claims as vectors (experiment with different embedding models)
  - Stage 2: reduce dimensionality with UMAP or PCA (test to see which is more performant)
  - Stage 3: cluster vectors using k-means (tuning k with the elbow method) or HDBSCAN, depending on the data size
  - Stage 4: represent clusters using TF-IDF words or summaries generated from generative LLMs



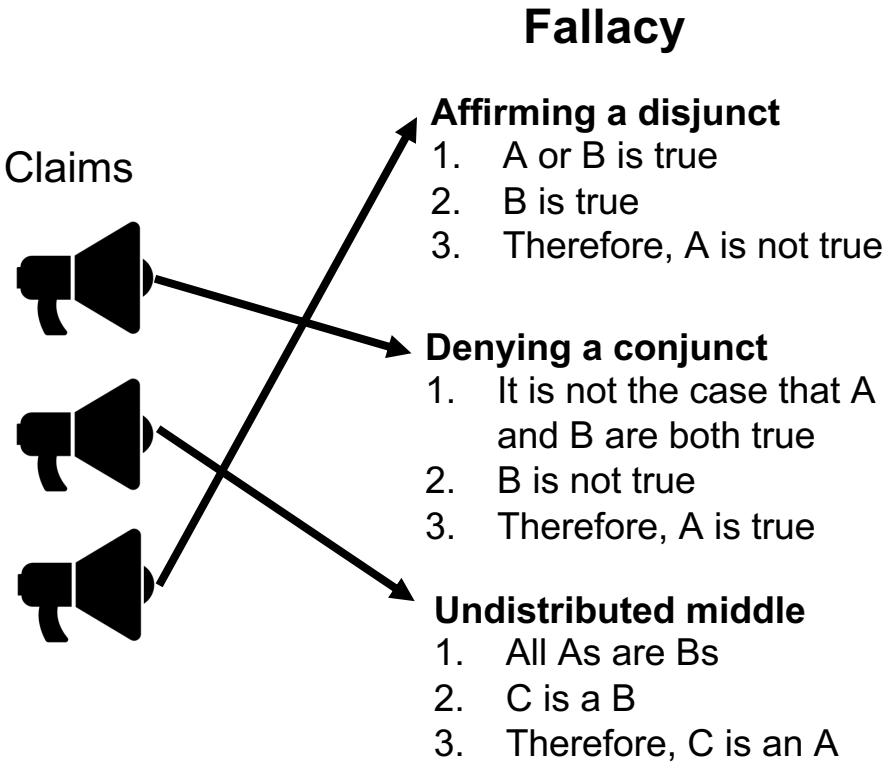
NLP 2: Claim Clustering (Entity Graph)

- Input = set of claims
- Output = clustered groups of claims based on topic
- Modeling:
  - Stage 1: extract entities from claims using named entity extraction (NER)
  - Stage 2: create a weighted bipartite graph of entities to claims
  - Stage 3: cluster the claims (evaluate existing community detection algorithms that operate on bipartite graphs)



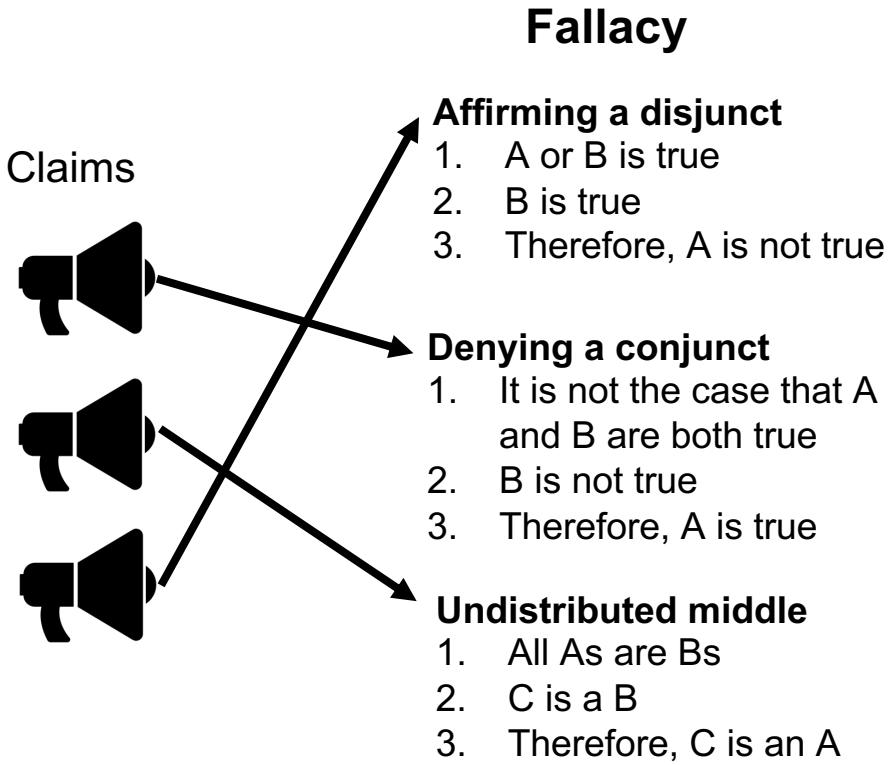
NLP 3: Logical Fallacies (Overview)

- Goal = detect the presence of logical fallacies in a claim
- Data = raw social media text containing a claim, labeled as containing a fallacy or not (result of claim classification study).
- Method Overview:
  - Binary classification of claims (contains fallacy or does not) using encoder models (BERT), encoder-decoder models (T5), and generative models (GPT).
  - Fine-tune on social media and determine which is the best cost/performance tradeoff
- Notable Existing Work:
  - Study demonstrated performance of different transformer models, but many new NLP techniques have emerged since this work.
- Contribution = Reframes fallacy detection from a multi-classification problem (identifying a specific fallacy) to a binary classification model that finds claims containing any fallacy. Also, generalizes across topics.



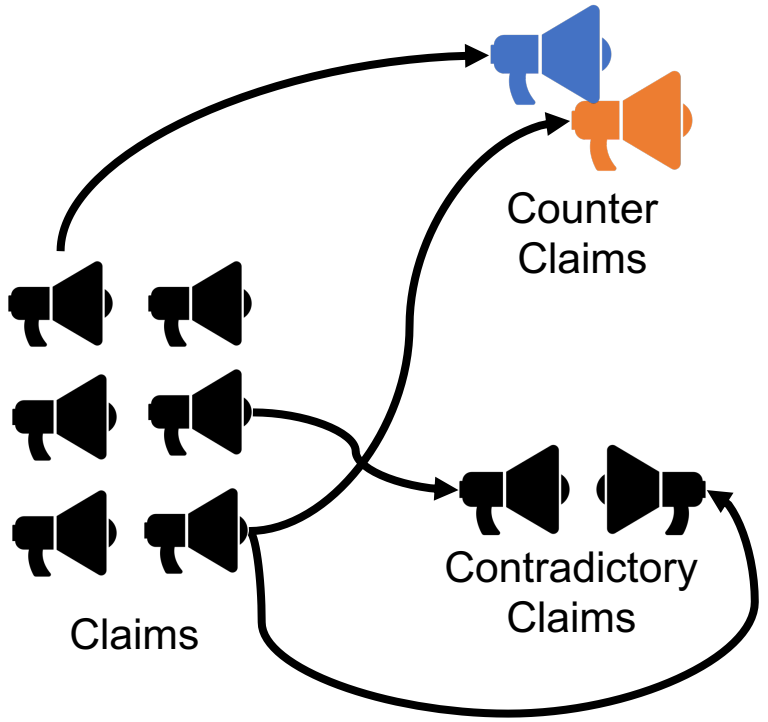
NLP 3: Logical Fallacies (Classification/Detection)

- Input = social media claims, some contain logical fallacies
- Output = binary label (contains fallacy or not)
- Modeling:
  - Stage 1: domain adaptive pre-training (DAPT) transformer models (distilBERT, BERT, RoBERTa) on large repositories of claims (adapting from models tuned on regular text).
  - Stage 2: task-tune models for binary classification of text containing logical fallacies.
  - Stage 3: task/domain-tune encoder-decoder model (T5) for binary classification of logical fallacy.
  - Stage 4: few and zero-shot prompt engineering for generative LLMs.
  - Stage 5: evaluate each method in terms of computational cost/performance.



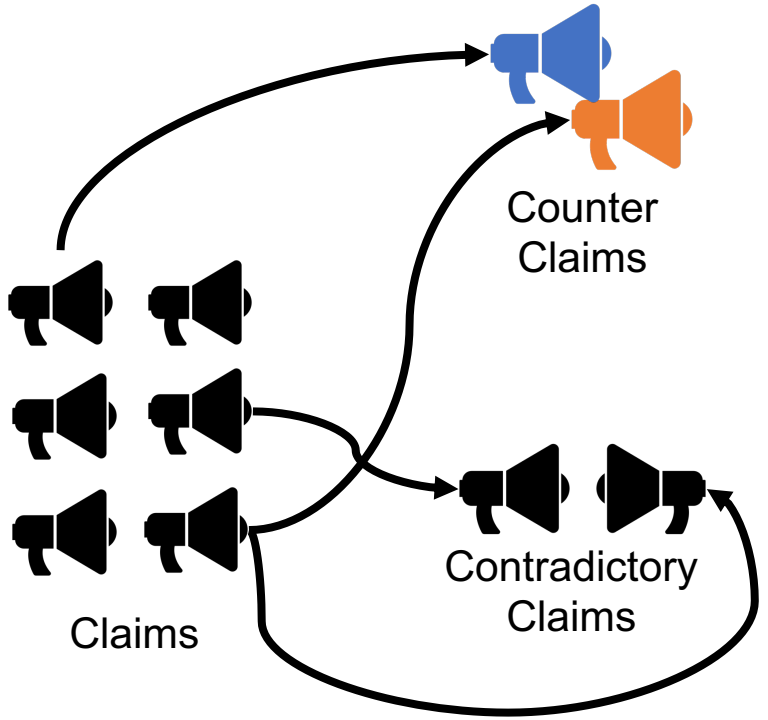
NLP 4: Counter and Contradictory Claims (Overview)

- Goal = detect counter and contradictory claims
- Data = pairs of claims tagged as counter, contradictory, or neither.
- Method Overview:
  - R1: build semantic networks of claims, use network features to detect opposing claims.
  - R2: train encoder model (BERT) and/or encoder-decoder model (T5) to classify pairs of claims as contradictory.
- Notable Existing Work:
  - Several studies demonstrate that BERT models are more effective than older neural network-based techniques for finding contradictory claim pairs, but their methods are out-of-date.
  - Study demonstrates how semantic networks can be used to detect contradictions, but their model is bogged down by also attempting to extract claims from text.
- Contribution:
  - A 2-pass approach using semantic networks and fine-tuned transformer models to identify pairs of counter and contradictory claims.



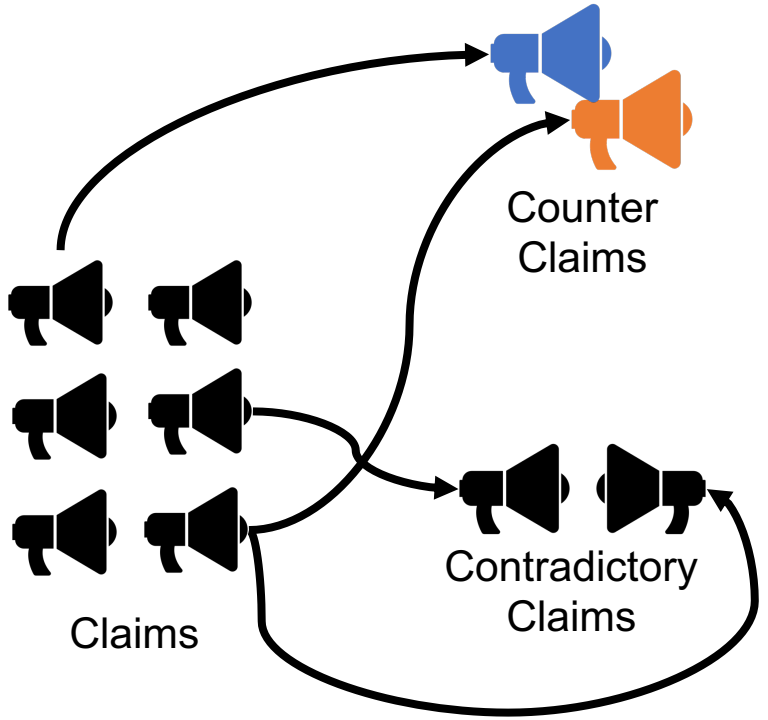
NLP 4: Counter and Contradictory Claims (Semantic Networks)

- Input = pairs of claims, extracted from social media text
- Output = labeled pairs of claims denoting if they are counter-claims, contradictory, or neither
- Modeling:
  - Stage 1: create semantic network from claims (using Netmapper)
  - Stage 2: manually identify subgraphs that correspond to counter and contradictory claims
  - Stage 3: identify unique network features between these subgraphs.
  - Stage 4: algorithm to search for subgraph features.



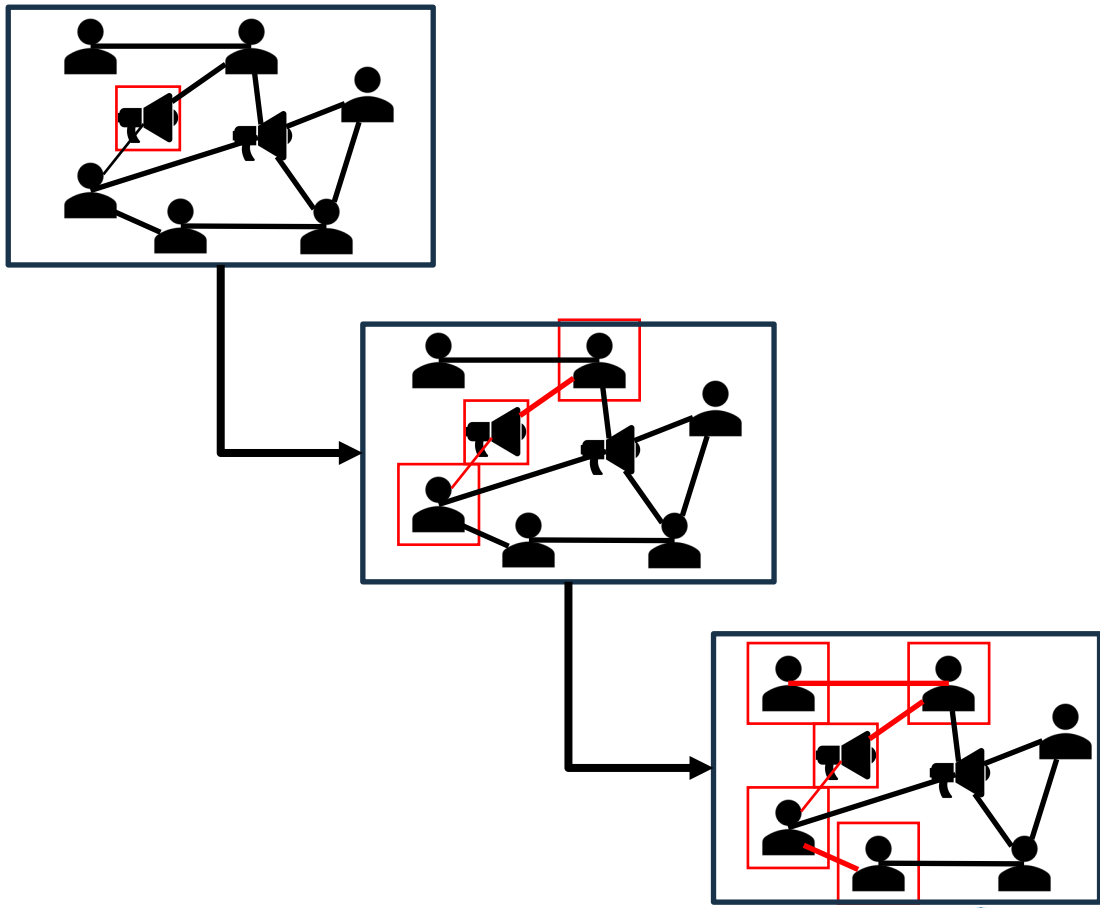
NLP 4: Counter and Contradictory Claims (Multi-Classification of Claim Pairs)

- Input = pairs of claims, extracted from social media text
- Output = labeled pairs of claims denoting if they are counter-claims, contradictory, or neither
- Modeling:
  - Stage 1: use repository of claims (preferably data from outside of the training data) to perform DAPT on transformer models to prepare them for use with claims.
  - Stage 2: task-tune the DAPT models for the multi-classification tasks of identifying counter and contradictory claims (or neither).
  - Stage 3: evaluate against zero- and few-shot training of encoder-decoder models (T5)
  - Stage 4: evaluate against reframing as two separate binary classification problems (counter or zero label; contradictory or zero label)



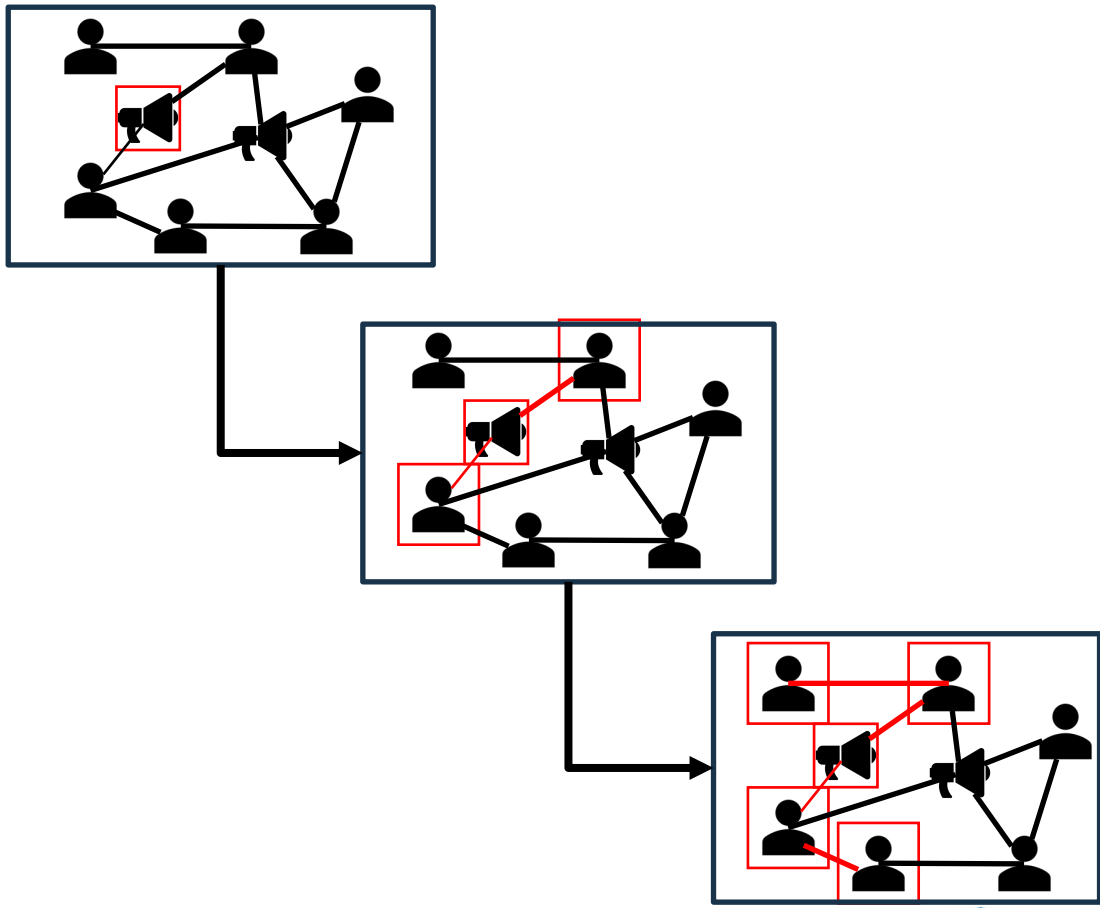
Network 1: Unreliable Source Propagation (Overview)

- Goal = locate users likely to post unreliable information and propagate reliability through social networks.
- Data = Twitter, Reddit, and Telegram data on COVID-19, US Politics, and Russia/Ukraine
- Method Overview:
  - Extract and label URLs in social media according to their reliability (using things like Media Bias Ratings)
  - Infer user reliability based on their relationships to URLs, and propagate to other users using social ties.
- Notable Existing Work:
  - Study demonstrates a methodology to propagate stance through social media using network features.
  - Several studies demonstrate demonstrate a methodology for trust propagation through social networks.
- Contribution:
  - An automated methodology for inferring the reliability of users in social graphs built from varied social media sources.



Network 1: Unreliable Source Propagation (Detection and Propagation)

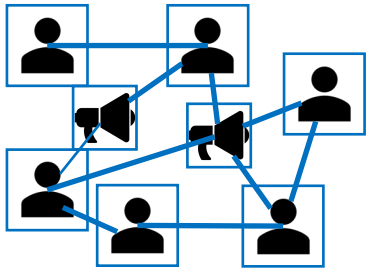
- Inputs = social media data containing at least network information and URLs
- Outputs = user labels for inferred reliability
- Modeling:
  - Stage 1: parse social media into network structure, extracting URLs (done with ORA)
  - Stage 2: crash URLs into repositories of media reliability (e.g., Media Bias Ratings) to generate a reliability score. Seek to use multiple repositories to limit potential bias.
  - Stage 3: propagate reliability from URLs to users using methods similar to the existing stance and trust propagation methodologies.
  - Stage 4: tune the propagation methods to leverage features from specific social media platforms (e.g., ”retweets” are specific to Twitter, so creating a Twitter-specific sub-model that improves performance over the base model).



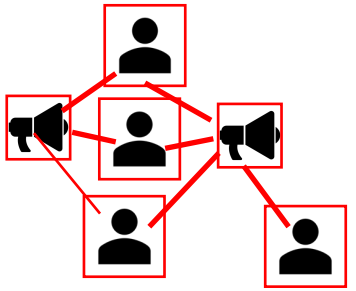
Network 2: Dynamic Network Signature of Misinformation Campaigns (Overview)

- Goal = identify social network features that are indicators of misinformation propagation.
- Data = Twitter, Reddit, and Telegram data on COVID-19, US Politics, and Russia/Ukraine
- Method Overview:
  - Find and label subgraphs where known misinformation is being shared and another set of subgraphs where it is verified that misinformation is not being shared.
  - Compare (using dynamic network analysis) subgraphs with positive and negative labels to find generalizable differences between misinformation networks and those that do not share misinformation.
- Notable Existing Work:
  - Study identified network signatures of misinformation on Twitter during the 2016 US presidential election.
  - Many studies describe network characteristics of communities sharing mis/disinformation, but few compare to similar communities sharing legitimate information.
- Contribution:
  - Generalizable network signatures of misinformation-propagating networks that can be used to identify potential misinformation.

Network without Misinformation



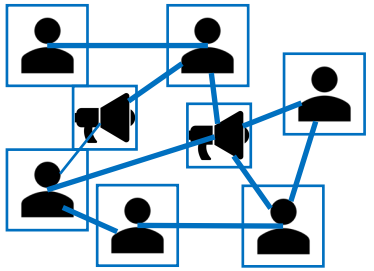
Known Misinformation Network



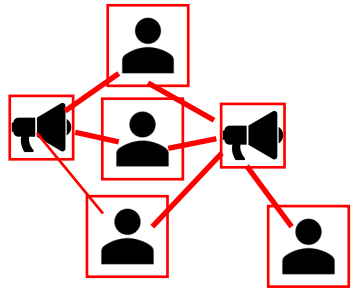
Network 2: Dynamic Network Signature of Misinformation Campaigns (DNA Analysis)

- Inputs = social media data containing at least network information
- Outputs = subgraph labels that indicate the likelihood that misinformation is present, which can be used to infer users' propensity for sharing misinformation
- Modeling:
  - Stage 1: build a dataset consisting of pairs of social networks that share misinformation and do not but are otherwise similar.
    - Ex) Medical misinformation communities compared to scientific discourse communities; or, conspiracy theory communities compared to political communities.
  - Stage 2: perform a comparative dynamic network analysis on the network sets, looking for metrics (and composite metrics) that differ.
  - Stage 3: develop algorithm to find network signatures of misinformation communities and apply it to data beyond what was used in Stage 1 to verify the generalizability of the method.

Network without Misinformation



Known Misinformation Network



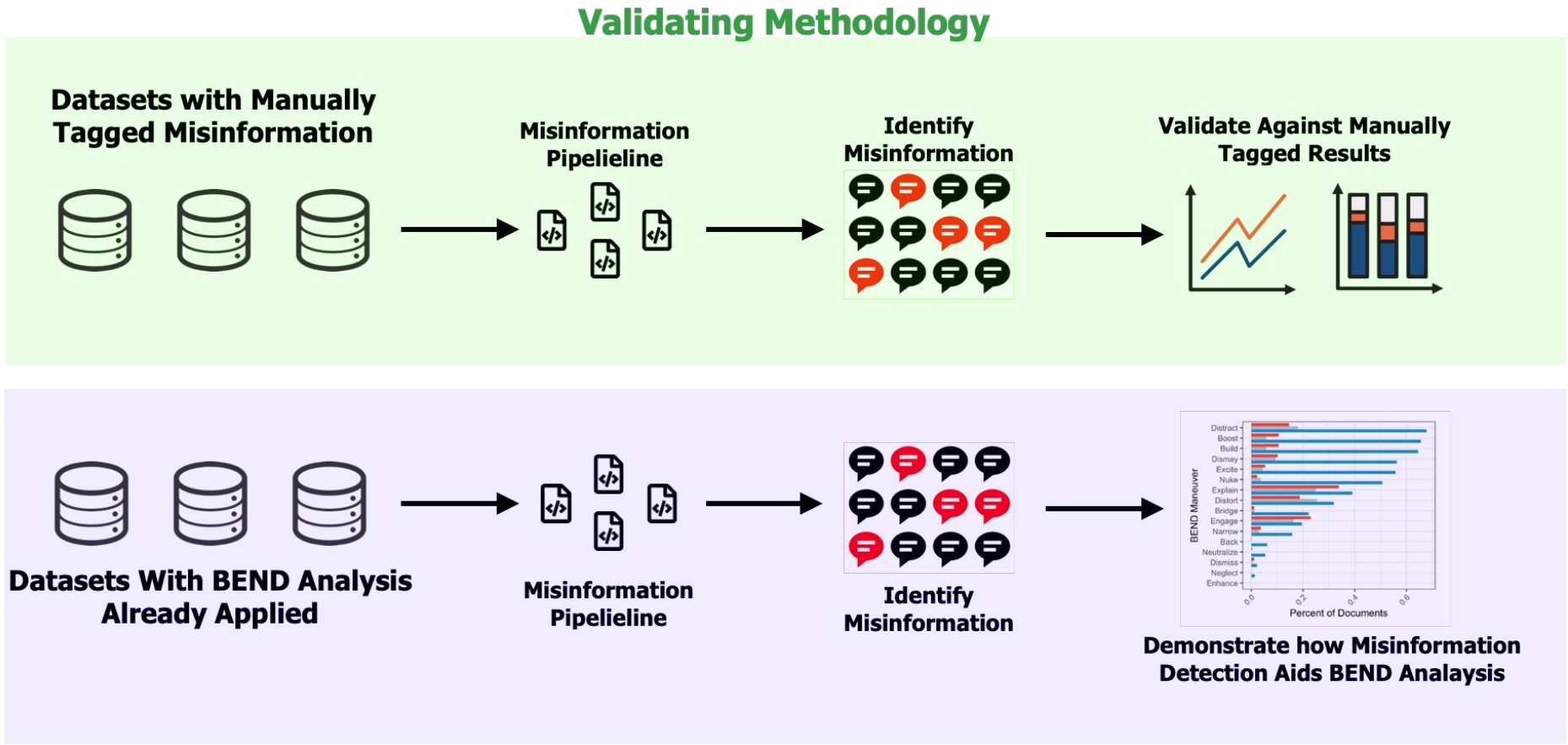
Proposed Detection Pipeline Data (Claim Dataset)

- Platforms: Twitter, Reddit, and Telegram
- Topics: COVID-19, Russian invasion of Ukraine, US Elections
- Existing Twitter data with claims about COVID-19 (500 Tweets) and US Elections (500 Tweets).
  - Add 500 additional Tweets to each existing dataset (1,000 Tweets)
  - Add data about the Russian invasion of Ukraine (1,000 Tweets)
- Create Telegram and Reddit repositories for all 3 topics (6,000 messages).
- Annotations for each message:
  - Claim location (specific claim in bigger message) → can infer claim present (binary)
- Estimated manual tagging: 8,000 messages
- Desired number of annotators >= 2 (including me)

Proposed Detection Pipeline Data

- Logical fallacy dataset:
  - Label positively identified claims as containing logical fallacy or not (binary)
- Claim pairs dataset:
  - Annotate (multi-class) as counter-claim, contradictory, or neither
  - Goal = 1,000 pairs of claims per platform (3,000 total)

Validation Pipelines



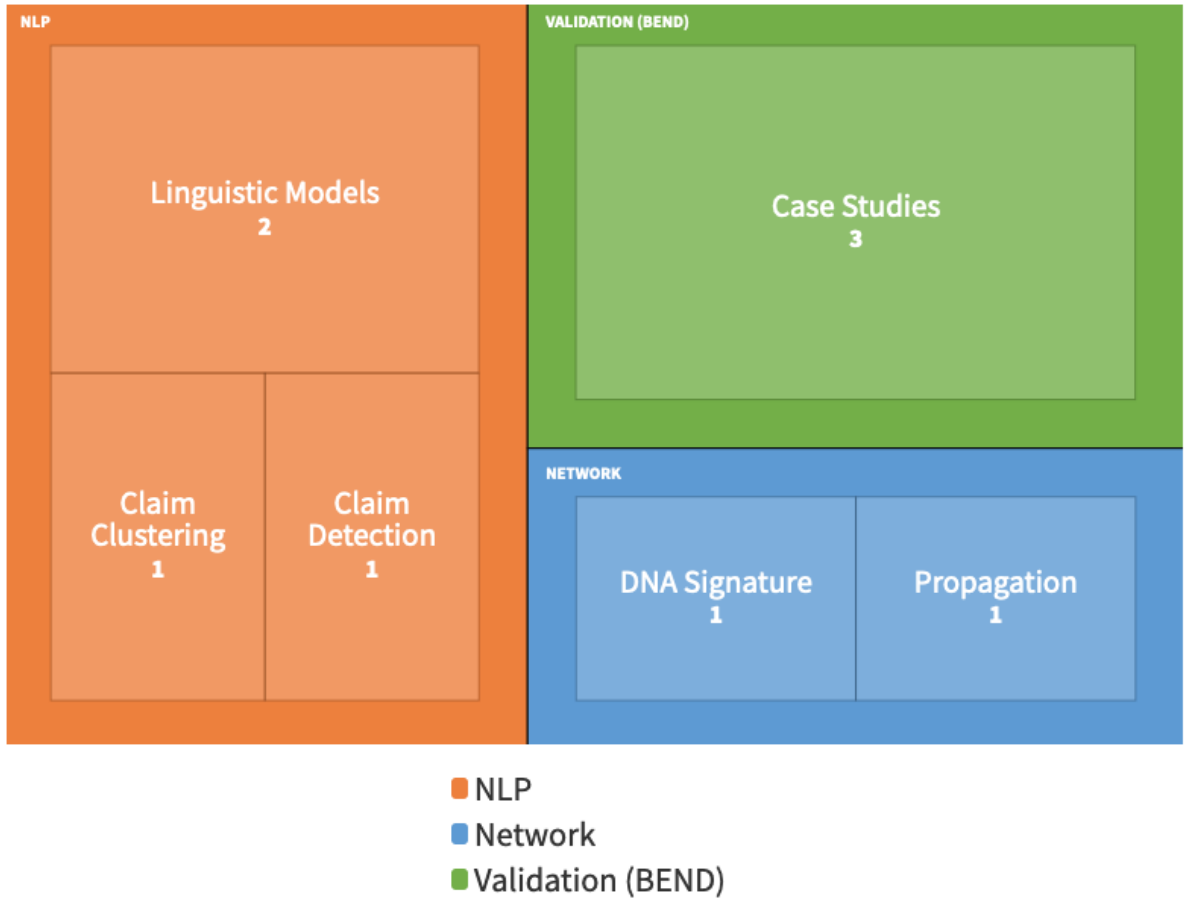
Proposed Validation Data

		Platform		
Domain	**Topic**	**Twitter**	**Telegram**	**Reddit**
Medical	COVID-19	X	X	X
Political	US Elections	X		X
War	Russian Invasion of Ukraine	X	X	

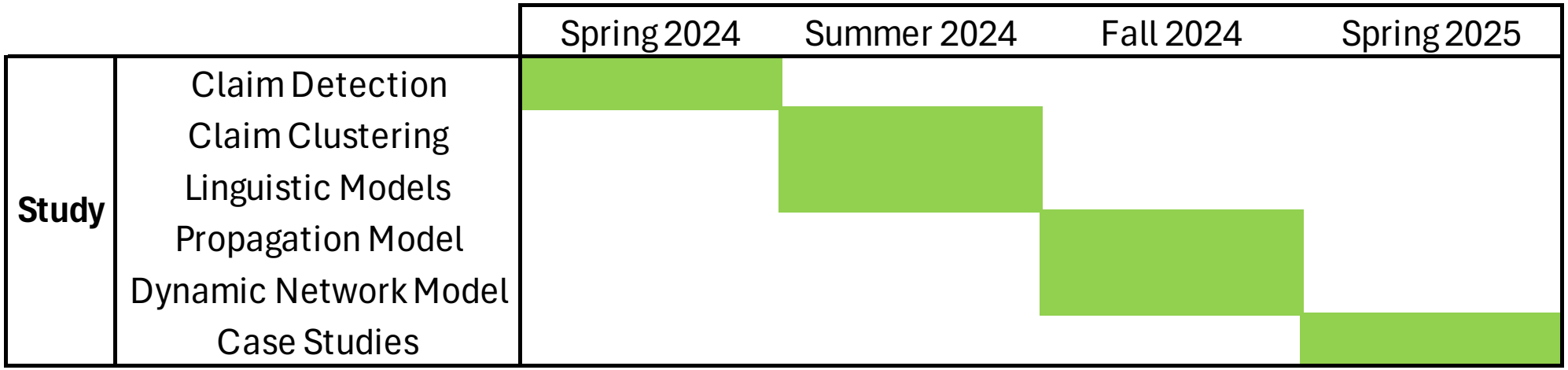
- ”x” denotes data that we already have
- Twitter/X and Reddit data will be predominately from before 2023, as both sources have blocked researcher API access

Research Roadmap (Workload)

- **NLP models**
  - Claim detection = 1 study
  - Claim clustering = 1 study
  - Linguistic models = 2 studies
    - Logical fallacy detection
    - Counterclaim detection
- **Network Models**
  - Propagation model = 1 study
  - DNA signature model = 1 study
- **Validation**
  - Case Studies = 3 studies
- **Total work** = 6 studies, 3 cases



Research Roadmap (Timeline)



- ***None of this work is complete***
  - Claim detection and clustering work is underway, Propagation, Linguistic Models, Dynamic Network model are in literature review stages

Limitations and Boundary Conditions: Language

- Only using English data or data translated into English
  - Better support for NLP methods and better sources on media reliability
  - Can expand to other languages in future as multilingual tooling develops
  - Must then focus on topics with misinformation targeting English speakers

Limitations and Boundary Conditions: Data

- Focus on text-based social media due to reliance on NLP methods
- Available data sources:
  - Telegram (data is accessible, but network features are unideal due to snowball sampling collection)
  - Reddit (API shut down, relying on existing or purchased data)
  - Twitter/X (API shut down, relying on existing or purchased data)
  - Facebook (API shut down, relying on existing or purchased data)

Limitations and Boundary Conditions: Types of Misinformation

- Cannot detect misinformation that isn't a logical fallacy or doesn't have some counter-conversation happening in the data
  - System is not meant to be a fact-checker, but is meant to find **relevant** misinformation
    - We expect that misinformation that is having an effect on the information environment will be met with counter points.
    - Misinformation that is having little effect on the information environment has less research value.
  - System is meant to enhance BEND analysis, which is applied in contested information spaces
    - Contested information spaces are likely to contain counter-conversation to misinformation claims.

Conclusion: Research Contributions

- Methodological contributions
  - Overall = an unsupervised methodology for misinformation detection using both linguistic and network features
  - Other contributions:
    - Improved multi-topic social media claim detection
    - Improved multi-topic claim clustering
    - Improved linguistic models for detecting counter-narratives and logical fallacies
    - Identification of dynamic network signatures of misinformation
    - Improved unreliable-source propagation in social media data
  - Datasets:
    - Expanded Twitter claim detection data (with new topics)
    - Claim detection data for Reddit and Telegram
- Application contributions
  - Enhance the BEND framework by providing another lens for social cybersecurity analysis
  - Better understanding of the tactics employed by bad actors in information operations

Conclusion: Real World Impact

- Misinformation emerges and is employed by bad actors in the information environment…but its true effects are difficult to measure without the ability to detect it at scale
- Understanding how misinformation is used is critical to mitigating any undesirable effects
- Current understanding of misinformation's role in social cybersecurity is primarily based on case studies in specific topic areas

Scalable misinformation detection is critical to addressing these key issues and furthering research into how misinformation is used, its effects, and mitigation strategies.