# Topic Modeling for Forensics Analysis of Text-Based Conversations

Michael Ippolito
michael.ippolito23@spsmail.cuny.edu

## 1 ABSTRACT

This project evaluated several unsupervised topic models performed on conversation-based textual datasets intending to simulate those extracted from electronic devices during law enforcement or cybersecurity investigations. Four publicly available datasets representative of this type of data were selected. Each dataset was preprocessed, modeled using various topic models, and postprocessed. Preprocessing steps included correcting spelling errors, converting slang and text-speak into conventional English, and adding synonyms, hypernyms, and keyphrases. Topic models included bag-of-words models, embeddings-based models, and transformer-based models. Post-processing was performed by using a text-generation large language model. Results were evaluated using coherence metrics, a semantic quality survey of human workers, and a binary value indicating relevance of the topic representation to the conversation text. Word2Vec and latent Dirichlet allocation (LDA) models preprocessed with keyphrases performed the best in terms of topic coherence and semantic quality. Word2Vec models preprocessed with synonyms and hypernyms were best for topic relevance, though preprocessing in this manner required higher runtimes and the survey data used to model the data was of questionable quality. Models preprocessed with keyphrases and modeling using Word2Vec or LDA were fast and yielded topic representations of high semantic quality and, therefore, are good candidates to use in investigations involving text-based conversation data.[1]

[1] Code for the project is here: https://github.com/mmippolito/cuny_data698_capstone

## 2 INTRODUCTION

In the fields of law enforcement and cybersecurity, investigators often seize devices such as mobile phones, computers, tablets, hard drives, flash drives, memory, and other electronic devices related to the event being investigated, be it a crime or other malicious activity. These devices routinely contain digital evidence that can lead to the prosecution of suspects, clues to the whereabouts of missing persons or property, or the expansion of an investigation to include additional persons of interest.

The data collected from these devices can be in virtually any format: videos, images, audio, instant messages, emails, PDF documents, and HTML documents. Further, the amount of information is often massive and can amount to gigabytes, terabytes, or even petabytes of data. Moreover, the data can be in different languages, contain spelling mistakes and colloquialisms, and, in their raw form, generally present a significant challenge to investigators to derive meaningful insight from them.

These challenges have led to the adoption of a litany of automated methods for approaching the task of forensic analysis, notably in the early stages of investigation in which exploratory data analysis (EDA) is conducted.

In particular, many methods exist to perform EDA on text-based documents. Indeed, textual analytical methods have received much attention for their potential investigative value. Typically, relevant forensic evidence exists in the form of human-to-human, text-based conversations, for example in instant messages, emails, or forum posts. As a first step in exploring this type of data, it is often beneficial to perform keyword searches or topic modeling to identify forensically interesting documents or to cluster semantically similar documents (de Waal, 2008).

While keyword searching is straightforward and requires only a generic technical ability, topic modeling can benefit from the application of machine learning techniques, specifically in the realm of natural language processing (NLP). Many such techniques exist, and the literature is full of examples. However, the vast

majority of these techniques are not specific to the investigation of human-to-human, text-based conversations. Those that are generally exhibit deficiencies in two key areas.

First, the topics generated by traditional topic models may not be of much (or any) forensic value to the investigator. Seized electronic devices will often contain a great deal of content that an investigator isn't necessarily interested in, such as a text message confirming the perpetrator's Uber Eats order or an argument with a significant other over WhatsApp. Being able to quickly identify these topics as uninteresting would help an investigator zero in on the remaining conversations that are more forensically relevant.

Second, the topics resulting from most models may not be semantically meaningful to the investigator. These topics are commonly referred to as the "topic representations" and are often simply lists of terms whose vectors are mathematically closest to each other. As an example, a topic model run against a series of text messages describing an impending cocaine bust might result in a topic representation containing words such as "blow," "nose_candy," "airport," "po_po," and "turn_around" (note the inclusion of slang, along with phrases like "airport" and "turn_around" which, by themselves, aren't readily identifiable as related to a drug bust). Arguably, a more interpretable topic representation would be, simply, "drug bust."

Because of these two deficiencies, it may be difficult for criminal intelligence analysts to find relevant information in forensic data extractions. Often, they want to simply answer the question, "Which conversations are relevant to my case?" and have a narrowly defined set of criteria they are looking for. This might be confined to a single topic (e.g., discussions about a drug sale) or multiple topics (discussions about the hiring of accomplices in addition to those involving the drug sale). The conversations not pertaining to these narrowly scoped topics may not have any relevance to the investigation, and the analyst may simply wish to discard them.

To improve upon these deficiencies, we propose a staged approach in which the data is preprocessed in a novel way before traditional, unsupervised topic

modeling is performed; following topic modeling, the topics are postprocessed to yield semantically meaningful topics and which are relevant to the investigation at hand.

## 3 PRIOR RESEARCH

Extensive research has been performed in the area of topic modeling. Latent semantic indexing (LSI) was first introduced in 1998 (Papadimitriou et al., 1998), followed by probabilistic latent semantic analysis (PLSA) in 1999 (Hofmann, 1999). In 2002, latent Dirichlet allocation (LDA) was introduced (Blei et al., 2003), is still commonly used today, and forms the basis for other more rigorous approaches.

Since its inception, LDA as a topic modeling algorithm has been extended by a number of researchers. Li and McCallum introduced "Pachinko allocation" (Li and McCallum, 2006). Other researchers proposed variations on LDA (discriminative LDA, maximum margin LDA, and others). McAuliffe and Blei (2010) introduced supervised topic models in 2008, while Hughes et al. (2018) describe a "semi-supervised prediction-constrained" topic model.

Beyond LDA, other methods employ term frequency-inverse document frequency (TF-IDF) and various distance-based methods to model topics. One interesting variation on this approach involves using named entity recognition (NER) to weigh topics more heavily when such entities are present (Krasnashchok and Jouili, 2018). Another TF-IDF-based model uses correlation explanations (CorEx) to "seed" the model with lists of anchor words prior to modeling (Alnusyan et al., 2020).

More recently, methods that use text embeddings have emerged as challengers to earlier approaches. In contrast to classic "bag of words" (BoW) models that represent text using sparse matrices, text embeddings represent words, sentences, or documents as dense vectors. These have been shown to be computationally more suitable for NLP tasks.

Embeddings can be generated in one of several ways (Brownlee, 2019). First, embeddings can be learned concurrently with a neural network-based model for an NLP function (such as topic modeling). Second, embeddings can be generated

based on a corpus of documents. Alternatively, pre-trained embeddings exist which fall under the category of large language models (LLMs) and which can be publicly downloaded (Brownlee, 2019). Examples include Word2Vec (Mikolov et al., 2013), global vectors for word representation (GloVe) (Pennington et al., 2014), and FastText (Bojanowski et al., 2016). As an extension of the concept of word and sentence embeddings, Angelov (2020) introduced topic vectors (Top2Vec) which attempts to generate semantically meaningful topics using a clustering technique based on centroids.

And finally, since 2018, transformer-based LLMs have given rise to models such as generative pre-trained transformers (GPTs) and Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) (Motro, 2023). These models represent the current state-of-the-art in NLP tasks. Specific to topic modeling, BERT spawned the development of BERTopic as a framework for building topic models (Grootendorst, 2022).

Evaluation of the quality of topics has also been extensively researched. Several topic coherence metrics have been proposed (Hadiat, 2022). First, UCI is based on the pointwise mutual information (PMI) calculation of pairs of words (Newman et al., 2010). Second, UMass, proposed by Mimno et al. (2011), is based on the "co-document frequency" between word pairs. Aletras and Stevenson (2013) introduced normalized PMI (NPMI), a modification to UCI. And finally, Röder et al. (2015) proposed a metric called Cv (context vector) based on cosine similarity between vectors.

Research into topic modeling particular to law enforcement has been sparse. De Waal et al. (2008) outline the use of topic modeling for forensics investigations but don't go beyond LDA in their methodology. Trenquier (2018) examined the now-famous Enron email dataset, again using an LDA topic model. But Trenquier's work also included an evaluation of topic quality by computing the similarity of topic representations using a vector-based (word2vec) approach. Trenquier reasoned that the intra-topic similarity of topic representations should be as high as possible while minimizing the inter-topic similarity.

## 4 HYPOTHESIS

To improve upon the deficiencies inherent to most unsupervised approaches to topic modeling, we propose a modification to the usual methodology. First, we propose additional preprocessing be performed prior to topic modeling in an attempt to generate topics that have more forensic value to an investigator. We hypothesized that topic coherence could be improved by first spell-checking the text, converting text-speak and other colloquialisms to standard English, adding synonyms and hypernyms to the tokenized text, and reducing the tokenized text to it most pertinent keywords. Section 6 describes this methodology in detail.

Second, the topic representations generated by the topic model were postprocessed in two ways. The topic representations were fed into a transformer-based text generator model to produce more semantically meaningful topics. Then, synonyms and hypernyms were generated for these topics, after which a second round of text generation was performed. Details for this process are provided in Section 6.

Quantitatively, the improvement introduced by this new methodology was evaluated in three ways. First, using industry-standard measures of coherence, topic coherence was numerically compared among the various models tried.

Conversely, semantic quality posed more of a challenge since it is subjective by definition. Therefore, we chose to gauge semantic quality using survey questions posed to human respondents using Amazon's Mechanical Turk service. Respondents were presented with topics generated by the proposed method and those generated by existing methods, after which they were asked to rank the topics numerically in terms of quality. In this way, a numerical representation of semantic quality was calculated by averaging the rankings of the combined survey results.

A third metric was used to evaluate topic relevance. A simple binary value of "yes" or "no" was attached to each hand-labeled topic-conversation pair based on the semantic quality score. Then a series of classification models was performed upon the labeled dataset, followed by predictions against the entire dataset, thereby

estimating which combinations of preprocessing, topic modeling, and postprocessing yielded the most relevant topics.

## 5 DATASETS

Because the goal was to generate topics from human-to-human interactions, the datasets useful in this context were obtained from existing text-based conversation datasets. We used the following open-source, freely downloadable datasets:

- Topical-Chat: 10,784 conversations between Amazon Mechanical Turk workers (Karthik, et al., 2019)
- Chit-Chat dataset: 7,168 conversations from the BYU Chit-Chat Challenge (Myers et al., 2020)
- Ubuntu Dialogue Corpus v2.0: 449,071 multi-turn chat dialogues (Lowe et al., 2015)
- Enron Email Dataset: email messages between about 150 Enron employees prior to the company's collapse (Enron Corp et al., 2015)

The first three datasets are well structured and contain a specific number of conversations in which humans take turns talking to one another in a series of messages. Each message can contain chats, which in turn contain sentences and words.

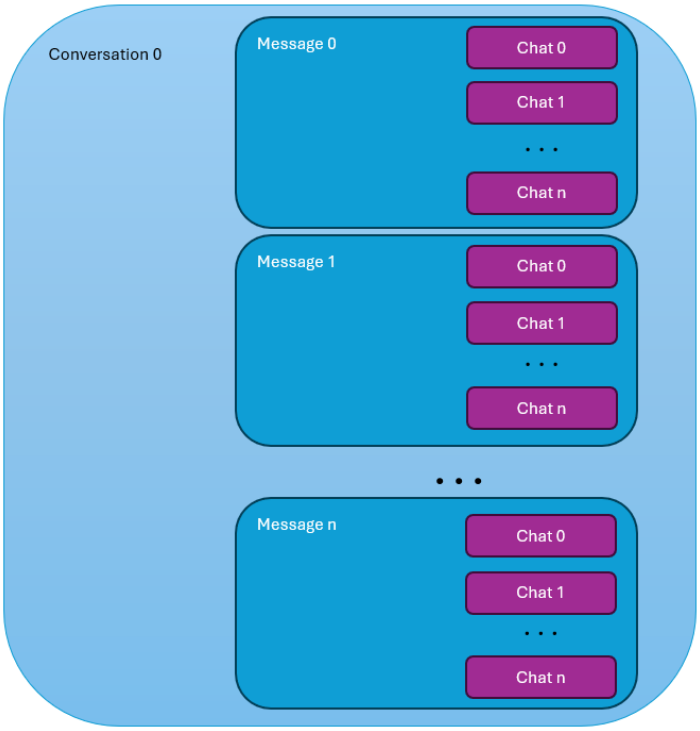


*Figure 1*—Conversation structure.

The Enron email dataset was different in that it wasn't practical to aggregate the messages into specific conversations, as each message could have more than one recipient. Therefore, each message was treated as a separate conversation for analysis purposes.

Preprocessing of the datasets was performed during tokenizing, which included lemmatizing and removing punctuation and stopwords, words with fewer than three characters, and any word that wasn't a noun, proper noun, or verb. Words with these parts of speech can be considered as having a higher semantic value than words such as adjectives and adverbs.

We also discarded conversations that included fewer than five exchanges between participants. This was necessary because a number of conversations were observed in which only one participant had joined, and the conversation consisted only of that participant asking if anyone was there.

Additional stopwords were removed from the Enron dataset, which included email headers containing words such as "forwarded" and location abbreviations such as "HOU" (presumably Houston). The following figure illustrates the keyword "HOU" in context using textacy's keyword-in-context (KWIC) functionality:

```
-------------------- Forwarded by Phillip K Allen/  HOU  /ECT on 10/16/2000  01:42 PM ----------
lation hurdles face at small percent. > > For 6-8   hou  rs a day  Microturbine run time: > Gas requirement
nd transport cost (firm or > interruptible). > > S  hou  ld you have additional questions, give me a call.
-------------------- Forwarded by Phillip K Allen/  HOU  /ECT on 10/09/2000  02:16 PM ----------
urchfield 10/06/2000 06:59 AM To: Phillip K Allen/  HOU  /ECT@ECT cc: Beth Perlman/HOU/ECT@ECT  Subject: Co
 To: Phillip K Allen/HOU/ECT@ECT cc: Beth Perlman/  HOU  /ECT@ECT  Subject: Consolidated positions: Issues
t;  the need for a single set of requirements. Alt  hou  gh the meeting with Keith,  on Wednesday,  was inf
e extremely difficult and time  consuming.  Throug  hou  t the meeting on Wednesday, Keith alluded to the
---------------- Forwarded by Richard Burchfield/  HOU  /ECT on 10/06/2000  08:34 AM ----------
verude 10/05/2000 06:03 PM To: Richard Burchfield/  HOU  /ECT@ECT cc: Peggy Alix/HOU/ECT@ECT, Russ Severson
To: Richard Burchfield/HOU/ECT@ECT cc: Peggy Alix/  HOU  /ECT@ECT, Russ Severson/HOU/ECT@ECT, Scott  Mills/
ECT@ECT cc: Peggy Alix/HOU/ECT@ECT, Russ Severson/  HOU  /ECT@ECT, Scott  Mills/HOU/ECT@ECT, Kenny Ha/HOU/E
/ECT@ECT, Russ Severson/HOU/ECT@ECT, Scott  Mills/  HOU  /ECT@ECT, Kenny Ha/HOU/ECT@ECT  Subject: Consolida
n/HOU/ECT@ECT, Scott  Mills/HOU/ECT@ECT, Kenny Ha/  HOU  /ECT@ECT  Subject: Consolidated positions: Issues
bility to revalue all options incrementally throug  hou  t the trading day.   Approximate delta changes bet
-------------------- Forwarded by Phillip K Allen/  HOU  /ECT on 10/09/2000  02:00 PM ----------
urchfield 10/06/2000 06:59 AM To: Phillip K Allen/  HOU  /ECT@ECT cc: Beth Perlman/HOU/ECT@ECT  Subject: Co
 To: Phillip K Allen/HOU/ECT@ECT cc: Beth Perlman/  HOU  /ECT@ECT  Subject: Consolidated positions: Issues
```

*Figure 2*—Keyword-in-context for the keyword "HOU".

Due to the large number of records in these datasets and limited computing resources available, we chose only a subset of the messages from each dataset. First, conversations with fewer than five exchanges were discarded. Second, only a limited number of conversations were kept in total, based on the constraints of loading the tokenizing model in memory. From this limited set of conversations, 250 conversations were chosen at random from each dataset. The following table shows the summary statistics for each dataset:

*Table 1*—Summary statistics per dataset.

| Dataset | Conversations | Exchanges | Avg Exch/Conv | Sentences | Avg Sent/Conv | Words | Avg Words/Conv |
|---|---|---|---|---|---|---|---|
| **Chitchat** | 2963 | 132554 | 44.736416 | 151156 | 51.014512 | 2610900 | 881.167735 |
| **Topical Chat** | 3000 | 65563 | 21.854333 | 118536 | 39.512000 | 1511902 | 503.967333 |
| **Ubuntu Dialogue** | 10000 | 99988 | 9.998800 | 61606 | 6.160600 | 1281429 | 128.142900 |
| **Enron Email** | 5000 | 5000 | 1.000000 | 47077 | 9.415400 | 1996747 | 399.349400 |

The conversations in the Chitchat dataset were, by design, fluid in nature. This is reflected in the top 20 most frequent words. Since the chat participants were

associated with Brigham Young University, many of the words are consistent with topics a student might discuss with other classmates (e.g., "school," "work," "class," and "like").

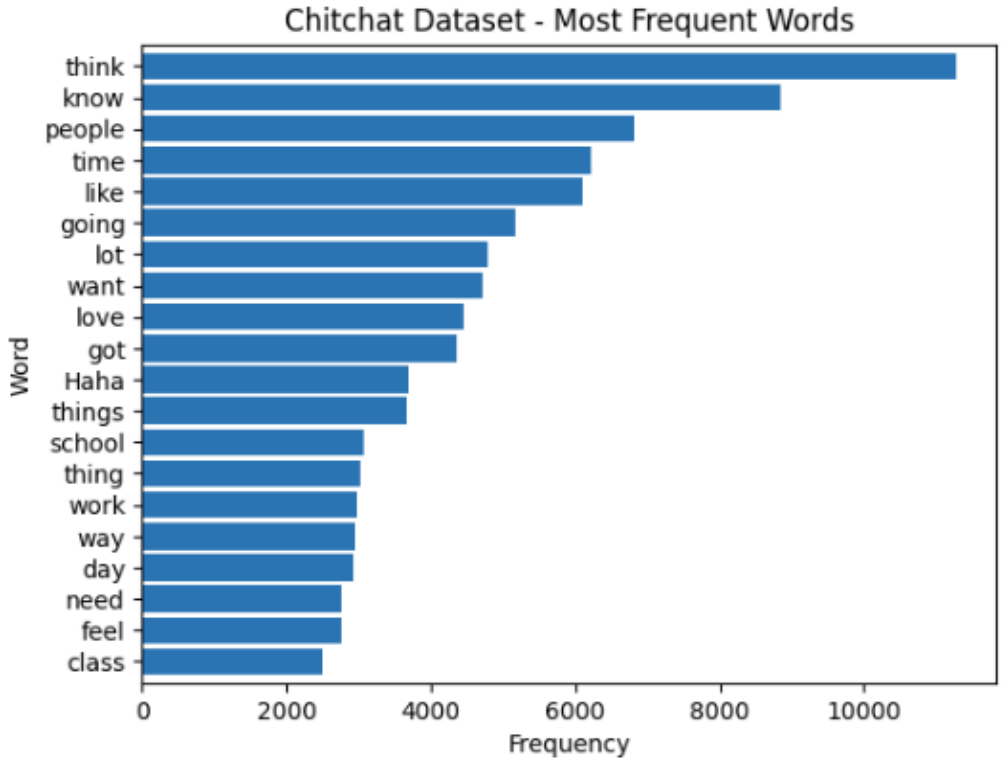


*Figure 2*—Chitchat dataset most frequent words.

Similar words within a corpus can also be visualized using t-distributed stochastic neighbor embedding (t-SNE) graphs. Using this technique, the top 10 most frequently used words were extracted from each dataset; the five words most similar to these top 10 words were then found based on their Word2Vec embedding vectors. Using example code from Sarkar (2019), the dimensionality of these vectors was then reduced to two-dimensional space and graphed, as shown below for the Chitchat dataset.

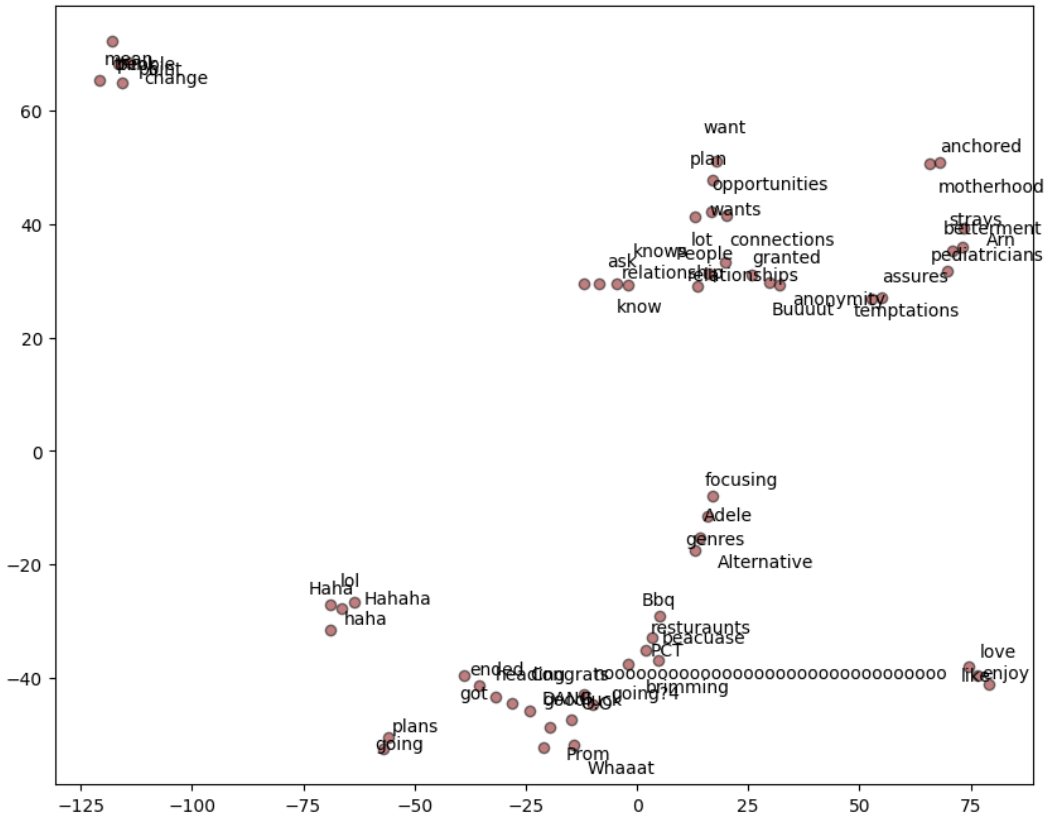


*Figure 3*—Chitchat dataset t-SNE graph.

As shown, clusters of similar words are evident, demonstrating the efficacy of the use of word embeddings to cluster documents. For example, “plans” and “going” are close to each other, as are “love,” “like,” and “enjoy.” Similarly, the words “Bbq” and “restaurants” are physically close, as are “lol” and variations on “haha.”

By comparison, the Topical Chat dataset was seeded with a topic question. This is somewhat reflected in the most frequent words in the dataset. For example, some topics dealt with movies, others with music, as shown below.

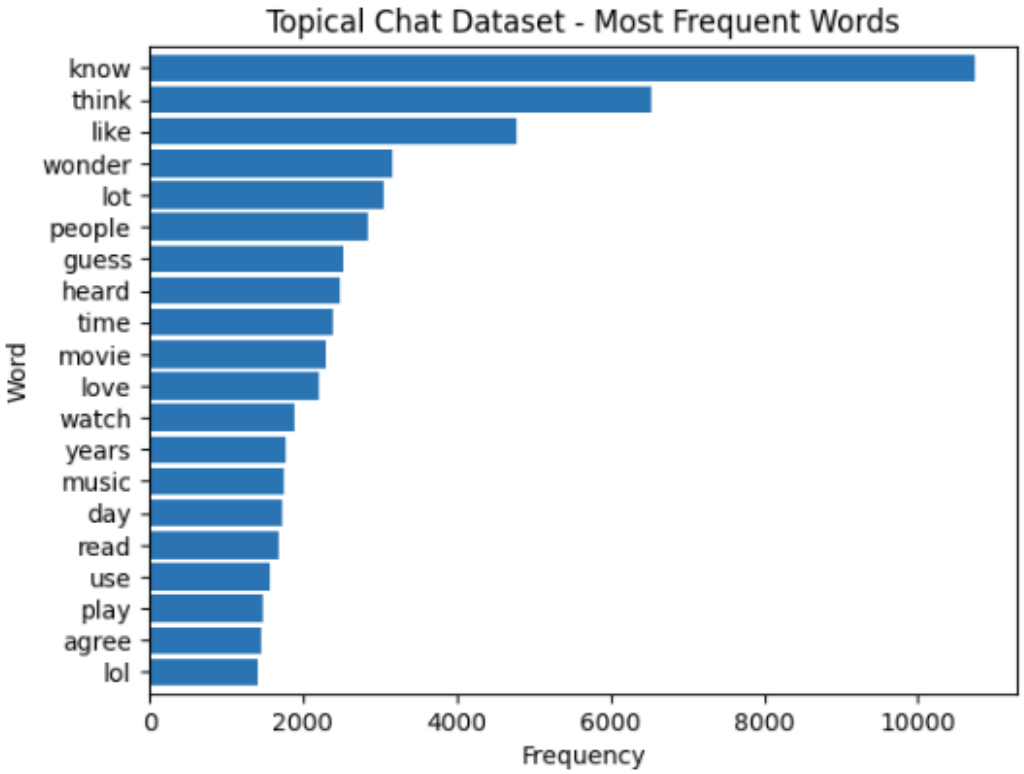


*Figure 4*—Topical Chat dataset most frequent words.

As with the Chitchat dataset, a t-SNE graph was constructed to illustrate clusters of similar words. “Film” and “movie” are close to each other, as are “eiffel” and “France.” Notably, the same three words in the Chitchat dataset (“love,” “like,” and “enjoy”) are clustered similarly in the Topical Chat dataset.

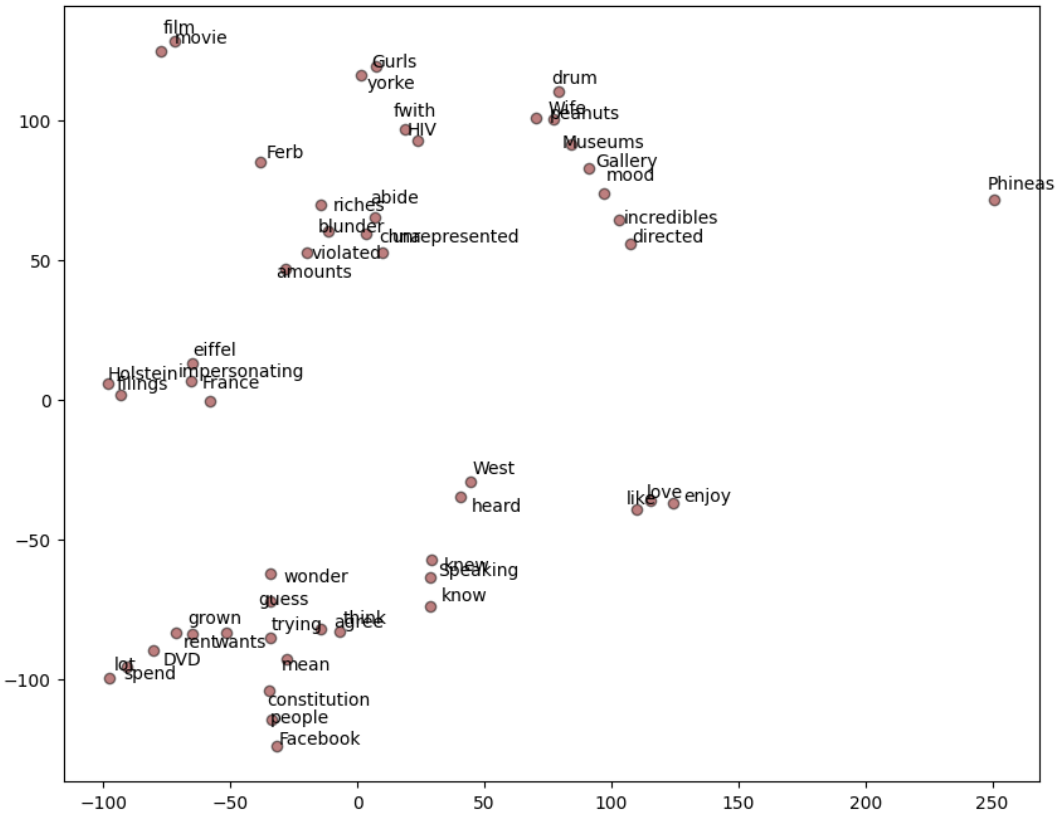


*Figure 5*—Topical Chat dataset t-SNE graph.

Unremarkably, the Ubuntu Dialogue dataset reflects the fact that the conversations revolved around the Ubuntu operating system.

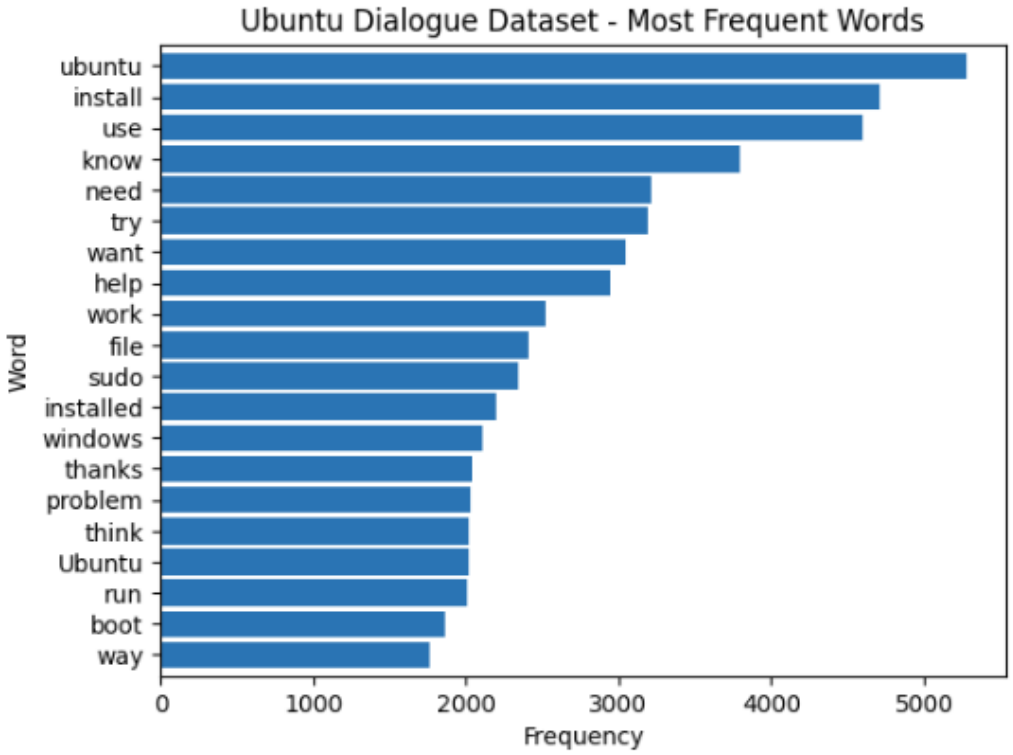


*Figure 6*—Ubuntu Dialogue dataset most frequent words.

In the t-SNE graph for the Ubuntu Dialogue dataset (Figure 7), it is observed that words like "distro" and "ubuntu" are clustered nearby, as are "installed" and "installing." It is perhaps noteworthy to point out that "Windows7" and "frustrated" are very close to each other, while "macbookpro" is only slightly distant from "frustrated." Although we only have anecdotal evidence to support the claim, this observation is consistent with the idea that Linux users have negative feelings towards Windows and Macintosh operating systems.

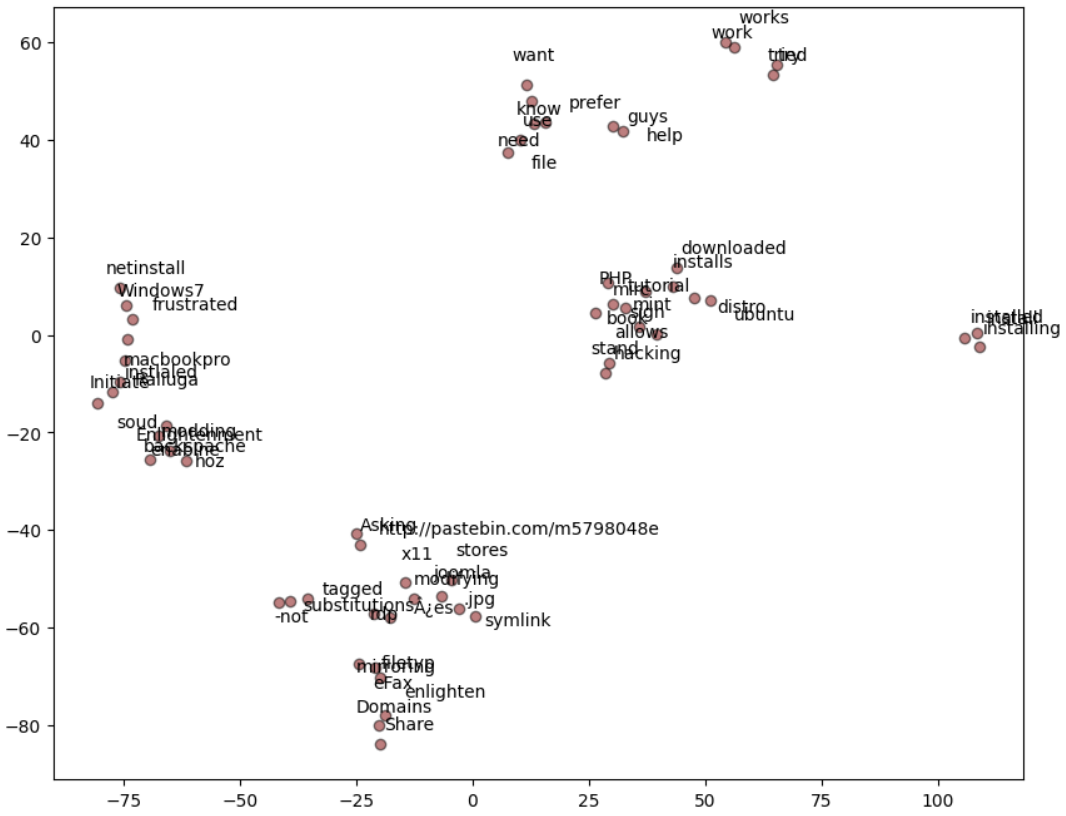


*Figure 7*—Ubuntu Dialog dataset t-SNE graph.

Finally, the most frequent words in the Enron email dataset include names of Enron employees as well as references to topics such as “gas” and “market,” expected given the former company’s dealings. When preprocessing the Enron dataset, words typically found in email headers were removed (for example, “subject,” “encoding,” “forwarded,” and “charset”).

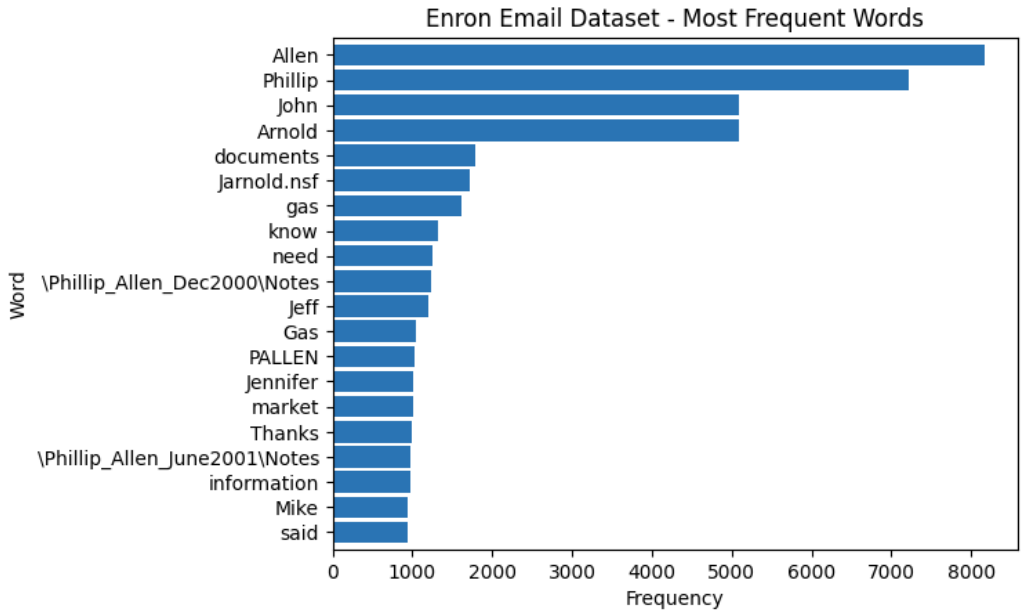


*Figure 8*—Enron email dataset most frequent words.

The t-SNE graph for the Enron Email dataset contains the most clearly delineated word clusters. One cluster includes phrases consistent with email headings and folders (e.g. "Jan," "Mar," "Folders\Discussion," "documents"), while another reflects verbs such as "help," "produce," "like," "need," and "know." A third cluster has to do with Enron employee John Arnold ("John," "Arnold", "jarnold.nsf"), while the last cluster can be construed to be associated with the economics of energy. This cluster includes words such as "prices," "gas," "storage," and "QFs"; QFs is interpreted to mean "qualifying facilities", i.e. alternative energy sources such as wind farms for which Enron was accused of receiving financial benefits despite having divested itself of these investments.

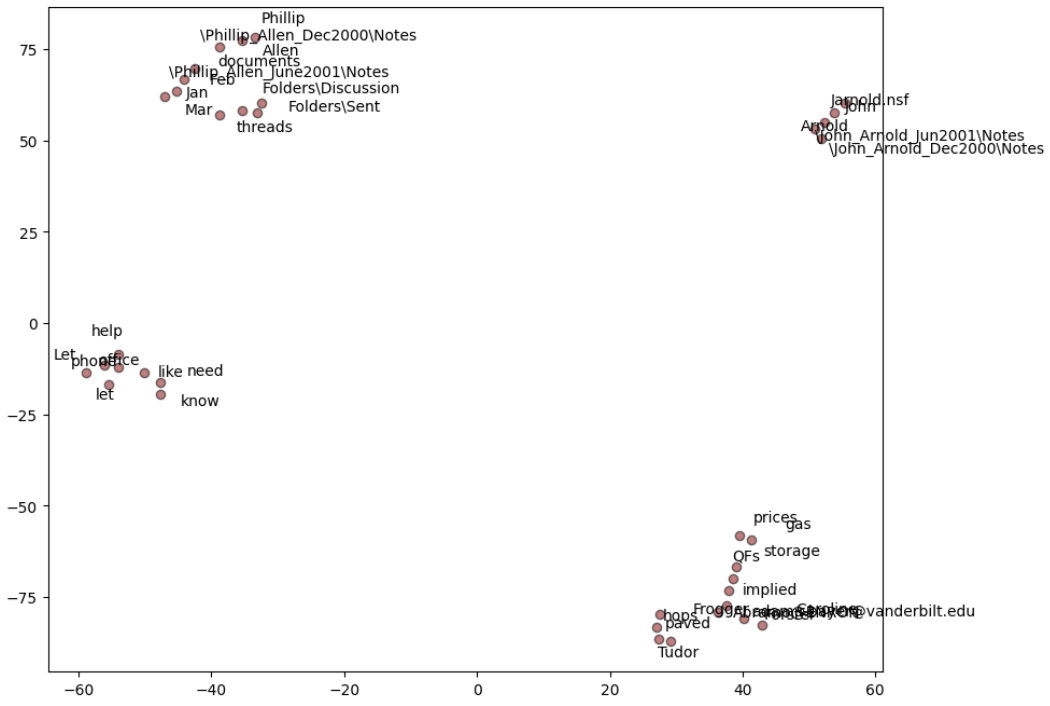


*Figure 9*—Enron Email dataset t-SNE graph.

In total, 20,963 conversations were processed, which included 303,105 exchanges between participants, 378,375 sentences, and approximately 7.4 million words.

## 6 STATISTICAL METHODS AND MODELS

### 6.1 Methods

As stated, we focused on topic modeling of textual, human-to-human conversations similar to those that remain as forensics artifacts stored on seized electronic devices. Specifically, we sought to improve upon the two previously stated deficiencies common in unsupervised topic models: deriving topics that are relevant to a cybersecurity or criminal intelligence analyst as it relates to a forensic investigation, and representing these topics in a semantically meaningful way.

To accomplish this, we explored a novel approach. Typically, a topic modeling solution consists of several steps. First, documents are preprocessed, usually by removing stop words, tokenizing, and lemmatizing. The documents are then converted into numerical format, for example as sparse or dense vector

representations. The documents are then clustered, and topics are generated based on the topics detected in each cluster. We modified this approach, first during preprocessing, and second after topics are generated by the topic modeling algorithm.

### 6.1.1 *Preprocessing*

During preprocessing, additional steps were performed to supplement the corpus's existing tokens. During this step, the following additional actions were included as model parameters:

- Tokens were corrected for spelling errors using the pyspellchecker Python library (Barrus, 2024).
- Text-speak and slang was converted to standard English using a slang converter available on Github (Verma, 2017).
- Tokens were lemmatized using SpaCy's native lemmatizer.
- For bag-of-words models, tokens were left in sparse representation.
- Dense representations were generated for embeddings-based models, using either the Word2Vec, GloVe, or Fasttext models.
- For bag-of-words and embeddings-based models, synonyms of tokens were added. For example, if the word "car" appeared in the corpus, the word "automobile" was also added. The Natural Language Toolkit (NLTK) includes utilities for generating synonyms. This step was omitted from transformer-based models, as transformers rely on the context inherent to full-text documents.
- Hypernyms of tokens were also added. Hypernyms are words that define a general category of an object. For example, "vehicle" is a hypernym of the words "car" and "truck." This step was expected to aid in entity categorization. The NLTK includes a utility to generate hypernyms. As with synonyms, generation of hypernyms was excluded from transformer-based modeling.
- Keyphrases were extracted from each conversation using a transformer model downloaded from huggingface.co (Ml6team, 2022). This model was based on papers by Kulkarni et al. (2021) and Sahrawa et al. (2020) that

describe keyphrase extraction from text using dense embeddings. For the model trials in which keyphrases were extracted, the entire conversation was replaced with these keyphrases in an effort to maximize the semantic value of generated topics.

- Control sets were also maintained in a non-preprocessed state to evaluate the efficacy of preprocessing.

### 6.1.2 *Topic modeling*

Following preprocessing, a battery of unsupervised topic models was executed using the tokens extracted both with and without the supplemental preprocessing. Section 6.2 includes a discussion of the models evaluated.

### 6.1.3 *Postprocessing*

Each set of topic representations was post-processed after unsupervised topic modeling was performed.

- An attempt was also made to generate interpretable, semantically meaningful topic representations using an LLM-based solution with text-generation capabilities. For this, two approaches were employed. First, we used BERTopic's KeyBERTInspired model (Grootendosrt, 2022). KeyBERT is a python library that can be loaded from BERTopic to extract keywords from topics using BERT-based text embeddings. Alternatively, we used Google's flan-t5-base text generation model. With this model, a prompt was used in the format, "I have a topic described by the following keywords: [word list]. Based on the previous keywords, what is this topic about?" The model was run iteratively against all topic representations in the dataset to generate a more semantically meaningful topic description.
- A second round of representations was generated, this time by adding synonyms and hypernyms to the topic representations from the original topic model. These representations were generated in the same way as above using Google's flan-t5-base text generation model.
- Control sets were maintained to evaluate the efficacy of postprocessing.

### 6.2 Topic Modeling

During the topic modeling stage, a total of 400 trials were conducted. Table 2 summarizes the combinations of model runs performed.

*Table 2*—Model parameters.

| Stage | Model parameter | Values |
|---|---|---|
| Pre | Spell-checked | yes, no |
| | Text-speak conversion | yes, no |
| | Synonyms | yes, no |
| | Hypernyms | yes, no |
| | Keyphrase extraction | yes, no |
| Model | Dataset | Chitchat, Topical Chat, Ubuntu Dialogue, Enron Email |
| | Model | (see Table 3 below) |
| Post | First-round text generation | yes, no |
| | Second-round text generation | yes, no |

Modeling was performed using the combinations of preprocessing steps listed above, namely checking spelling or not, translating slang or not, and reducing conversations to their keyphrases or not. In addition, models were tried both with and without adding synonyms and hypernyms. For bag of words models, the number of passes was fixed at 15.

During the topic modeling stage, traditional bag-of-words topic models were run, namely LSI, LDA, and NMF. These models require a predefined number of topics to be configured; we used 15 and 30.

For topic models that use text embeddings rather than bags-of-words, we alternately used pretrained Word2Vec, GloVe, and Fasttext embeddings (rather than training our own), with a vector size of 200, a minimum word count of 1, and a clustering algorithm of either K-means or density-based spatial clustering and application with noise (DBSCAN). To remain consistent with bag-of-words models, K-means clustering was executed with either 15 or 30 clusters. We retained the default maximum iterations of 300 and tolerance of 0.0001. DBSCAN clustering required choosing an epsilon and minimum sample size; based on several trial runs, we chose epsilon to be either 0.1 or 1.0 and minimum sample size of either 2 or 3.

Transformer-based models used fewer model parameters; for these models we only varied spell-checking and slang translation, and additionally KeyBERT topic representations.

Parameters used during the modeling stage are summarized in Table 3.

*Table 3*—Model parameters (topic modeling stage).

| Family | Model | Common Parameters | Model-Specific Parameters |
|---|---|---|---|
| Bag of words | LSI | 15 passes | 15 clusters |
| | | | 30 clusters |
| | LDA | | 15 clusters |
| | | | 30 clusters |
| | NMF | | 15 clusters |
| | | | 30 clusters |
| Embedding | Word2Vec | 200 vectors<br>min count of 1<br>300 max iterations<br>0.0001 tolerance | K-means clustering<br>15 or 30 clusters |
| | GloVe | | |
| | Fasttext | 200 vectors<br>min count of 1 | DBSCAN clustering<br>epsilon of 0.1 (3 samples min) or 1.0 (2 samples min) |
| Trans-former | BERTopic | N/A | Native BERTopic representations |
| | | | KeyBERT-based representations |

### 6.3 Topic Coherence

Topics produced by each model were evaluated for topic coherence using four metrics: UCI (Newmann et al., 2010), normalized pointwise mutual information (NPMI) (Bouma, 2009), UMass (Rosner et al., 2014), and Cv coherence (Röder, 2015).

### 6.4 Semantic Quality Survey

Because evaluating the semantic quality of topic representations is subjective by nature, for this purpose we chose to conduct a survey of human workers using

Amazon's Mechanical Turk. To keep costs within a reasonable budget, only a limited dataset was submitted to Mechanical Turk workers for evaluation. Two conversations were selected from each dataset with a word count such that an English-speaking adult could read the document in about two minutes, assuming an average rate of 238 words per minute (Scholar Within, 2024). 36 model parameter combinations were selected for each document and assigned to three Mechanical Turk workers each. Each combination thereof (dataset, document, model parameters, and worker) constituted a single human intelligence task (HIT). A total of 864 HITs were presented to workers.

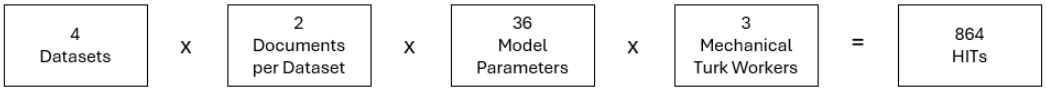


*Figure 10*—Mechanical Turk human intelligence task count.

Each HIT consisted of the full text of the conversation along with the topic representation generated by the corresponding model run. The worker was asked to rate the quality of the topic on a scale from 1 to 5, with 5 being the best.

Human-generated topic word lists for each document were also generated and submitted to workers for evaluation, as were a human-summarized topic word or phrase. In addition, nonsensical topic representations were included as "attention responses" commonly included in surveys to help discern valid from spurious answers.

*Table 4*—Mechanical Turk human intelligence task list.

| | | Preprocessing | | | | Postprocessing | |
|---|---|---|---|---|---|---|---|
| **Model** | **Model Parameters** | **Spellcheck/ Textspeak Conversion** | **Syonyms/ Hypernyms** | **Keyphrases** | **Keybert** | **Flan Text Generation** | **Synonyms/ Hypernyms + Flan Text Generation** |
| Human | Keyword list | - | - | - | - | - | - |
| Human | Single keyword/keyphrase | - | - | - | - | - | - |
| LDA | 15 topics | yes | - | - | - | - | - |
| LDA | 15 topics | yes | yes | - | - | - | - |
| LDA | 15 topics | yes | - | yes | - | - | - |
| Word2Vec | K-means (15 clusters) | yes | - | - | - | - | - |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Word2Vec | K-means (15 clusters) | yes | yes | - | - | - | - |
| Word2Vec | K-means (15 clusters) | yes | - | yes | - | - | - |
| GloVe | K-means (15 clusters) | yes | - | - | - | - | - |
| GloVe | K-means (15 clusters) | yes | yes | - | - | - | - |
| GloVe | K-means (15 clusters) | yes | - | yes | - | - | - |
| Bertopic | - | yes | - | - | - | - | - |
| Bertopic | - | yes | - | - | yes | - | - |
| Bertopic | - | yes | - | yes | - | - | - |
| LDA | 15 topics | yes | - | - | - | yes | - |
| LDA | 15 topics | yes | yes | - | - | yes | - |
| LDA | 15 topics | yes | - | yes | - | yes | - |
| Word2Vec | K-means (15 clusters) | yes | - | - | - | yes | - |
| Word2Vec | K-means (15 clusters) | yes | yes | - | - | yes | - |
| Word2Vec | K-means (15 clusters) | yes | - | yes | - | yes | - |
| GloVe | K-means (15 clusters) | yes | - | - | - | yes | - |
| GloVe | K-means (15 clusters) | yes | yes | - | - | yes | - |
| GloVe | K-means (15 clusters) | yes | - | yes | - | yes | - |
| Bertopic | - | yes | - | - | - | yes | - |
| Bertopic | - | yes | - | yes | - | yes | - |
| LDA | 15 topics | yes | - | - | - | yes | yes |
| LDA | 15 topics | yes | yes | - | - | yes | yes |
| LDA | 15 topics | yes | - | yes | - | yes | yes |
| Word2Vec | K-means (15 clusters) | yes | - | - | - | yes | yes |
| Word2Vec | K-means (15 clusters) | yes | yes | - | - | yes | yes |
| Word2Vec | K-means (15 clusters) | yes | - | yes | - | yes | yes |
| GloVe | K-means (15 clusters) | yes | - | - | - | yes | yes |
| GloVe | K-means (15 clusters) | yes | yes | - | - | yes | yes |
| GloVe | K-means (15 clusters) | yes | - | yes | - | yes | yes |
| Bertopic | - | yes | - | - | - | yes | yes |
| Bertopic | - | yes | - | yes | - | yes | yes |

### 6.5 Topic Relevance Modeling

Based on the survey responses, each topic representation was deemed either relevant or irrelevant; quality scores of 3 or higher were presumed to indicate relevant topics, while scores of 2 or lower were considered irrelevant. Along with the modeling parameters used to generate the topic representation, this binary

indicator of relevance was taken to be a labeled dataset with which further predictions could be made against the unlabeled portion of the dataset.

To aid in modeling, additional topic representations were scored for relevance by the author. In all, 1,347 observations were labeled (456 by the author and 891 by Mechanical Turk workers), of which 689 were relevant and 658 were irrelevant.

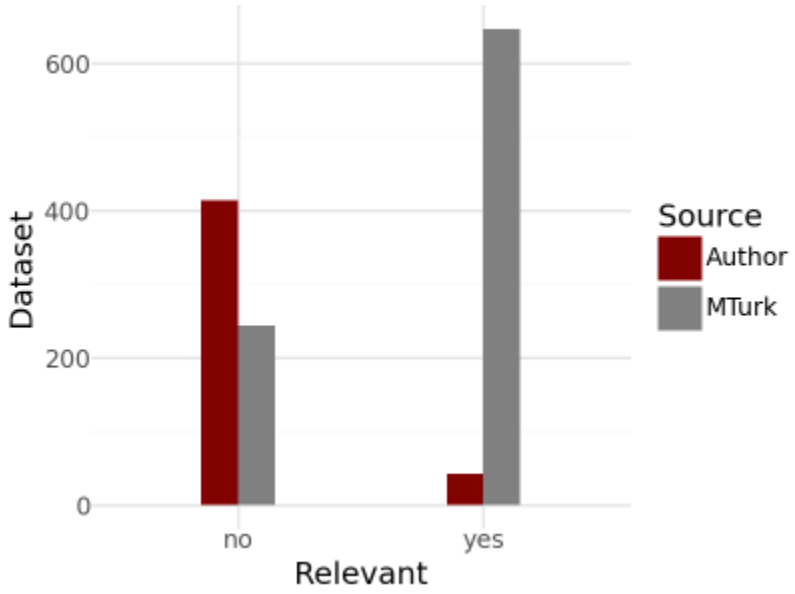


*Figure 11*—Relevance dataset label counts.

It is noted that Mechanical Turk workers rated topic words as relevant at a much higher volume than did the author. This may be indicative of the quality of Mechanical Turk workers' responses and is discussed in Section 7.

As an additional parameter to aid in modeling topic relevance, cosine similarity scores were calculated between each conversation and the topic representation generated from each round:

- Topic words from the original topic model
- first round of representations generated by the LLM
- second round of LLM-generated representations

The following table summarizes the features used for topic relevance modeling.

*Table 5*—Modeling feature summary.

| Feature | Description |
|---|---|
| Cosine_similarity | Cosine similarity between topic representation and conversation text |
| Dataset | Chitchat, Topical Chat, Ubuntu Dialogue, Enron Email |
| Hypernyms | True/False |
| Keyphrases | True/False |
| Model | LDA, word2vec, GloVe, Bertopic |
| Model_family | Bag of words, embedding, transformer |
| Model_run | Additional model-specific parameters |
| Num_topics | Number of topics |
| Relevant | True/False (label) |
| Representation | Topic words, LLM-generated (round 1), LLM-generated (round 2) |
| Source | Author/Mechanical Turk workers |
| Spell_checked | True/False |
| Synonyms | True/False |
| Text_speak conversion | True/False |

A series of classification models were run against the labeled dataset using five-fold cross-validation, reserving 30% of the observations for validation. The models run were linear discriminant analysis (LDA), support vector machines (SVM), logistic regression, naïve Bayes, K-nearest neighbors (KNN), tree-based, random forest, and gradient boosting. The validation set was then scored for accuracy, and the best model was chosen based on the highest accuracy score.

Using the best model, a receiver operating characteristic (ROC) curve was generated and a probability threshold selected to balance the true positive rate (TPR) against the false positive rate (FPR). The best model's parameters were used to predict relevance for the remaining (unlabeled) portion of the dataset delineated by the selected probability threshold. A full discussion of the models and results is included in Section 7.

## 7 DISCUSSION OF RESULTS

Each model run yielded a set of three topic representations: word lists produced directly by the topic model, a more concise topic representation generated by Google's flan-t5-base LLM, and another concise representation generated by the same LLM but after the original topic word list was postprocessed using synonyms and hypernyms.

For example, a snippet from one of the conversations in the Chitchat dataset includes the following text:

> **UserA**: I would travel the world, go scuba diving, and maybe explore places I've never been before.
>
> **UserB**: Where would you go to first?
>
> **UserA**: I would probably go to Saipan. It's a place in the Marianna Islands that has some really cool scuba sites with WWII wrecks. What about you?
>
> **UserB**: I would go to Korea first because my really old great grandma lives there and I want to visit her

Examples of topic representations resulting from various models are given in Table 6.

*Table 6*—Example topic representations for various model types.

| Topic Model | Preprocessing | Postprocessing | Example Topic Representation |
|---|---|---|---|
| LDA, 15 topics | None | None | people, class, world, remember, end, bit, year, come, sound, send |
| LDA, 15 topics | None | First-round LLM | People |
| LDA, 15 topics | None | Second-round LLM | kinfolk |
| Word2Vec/K-means, 15 clusters | Keyphrases | None | China, India, Africa, Europe, Germany, Greece, Italy, Korea, Brazil, Albania |
| Word2Vec/K-means, 15 clusters | Keyphrases | First-round LLM | World |
| Word2Vec/K-means, 15 clusters | Keyphrases | Second-round LLM | World |
| BERTopic w/KeyBERT | None | None | fun, things, work, would, about, go, think, going, ive, do |
| BERTopic w/KeyBERT | None | First-round LLM | Work |
| BERTopic w/KeyBERT | None | Second-round LLM | field of study |

As shown in the table, the non-postprocessed representations include a list of words while the first- and second-round LLM-generated topics are represented by a more concise word or phrase. The second-round LLM topic appeared to choose a phrase that could be considered a stretch to represent the conversation as a topic; this is likely due to the effect of adding synonyms and hypernyms, which tended

to place undue importance on the added words; it is possible that weighting the original topic words more heavily than they synonyms and hypernyms may have aided in the second-round LLM-generated topics, but this was left for future study.

It should be noted that modeling results using embedding-based models with DBSCAN clustering yielded an abundance of empty clusters. Consequently, these topic representations were discarded, and the results discussed below exclude models run using DBSCAN clustering. It is likely that the high dimensionality of the data combined with a one-size-fits-all epsilon and minimum sample size specified as parameters prevented clusters from being formed. Additionally, epsilon is typically specified on a per-dataset basis and does not work well when generalized over many datasets of different sizes. Since the aim of the study was to compare results across all datasets, DBSCAN was found to be less suitable as a clustering technique than K-means.

As described in Section 6, topic modeling results were evaluated in three ways: coherence, semantic quality, and relevance of the topic representations.

### 7.1 Coherence

Four coherence metrics were calculated for each model run. Each metric had its benefits and drawbacks as a means of evaluating topic coherence. However, one of the metrics (Cv score) had to be discarded because it was discovered that the metric's author admitted to finding discrepancies between calculations in his paper and those performed by peers using the same data (Röder, 2018).

Of the three metrics remaining, UMass scores were problematic in that the UMass formula is based on how often the top N words occur together within a corpus; since keyphrases use n-grams rather than single words, the UMass scores appeared to be artificially low.

Both the UCI and NPMI scores are based on pointwise mutual information (PMI) and exhibited very similar results, the difference being that UCI is based on a sliding window of words within a corpus, while NPMI is based on the entire corpus. Using PMI as a coherence metric generated artificially high values for

conversations preprocessed with keyphrases; because PMI relies on the co-occurrence of topic representations within a corpus, and because keyphrase preprocessing involved replacing the entire conversation with keyphrases, it is natural that PMI would be higher than other metrics.

In general, topics preprocessed with keyphrases and synonyms/hypernyms exhibited the best coherence scores, noting the caveat regarding low UMass scores for keyphrases. Figure 12 illustrates this discrepancy.

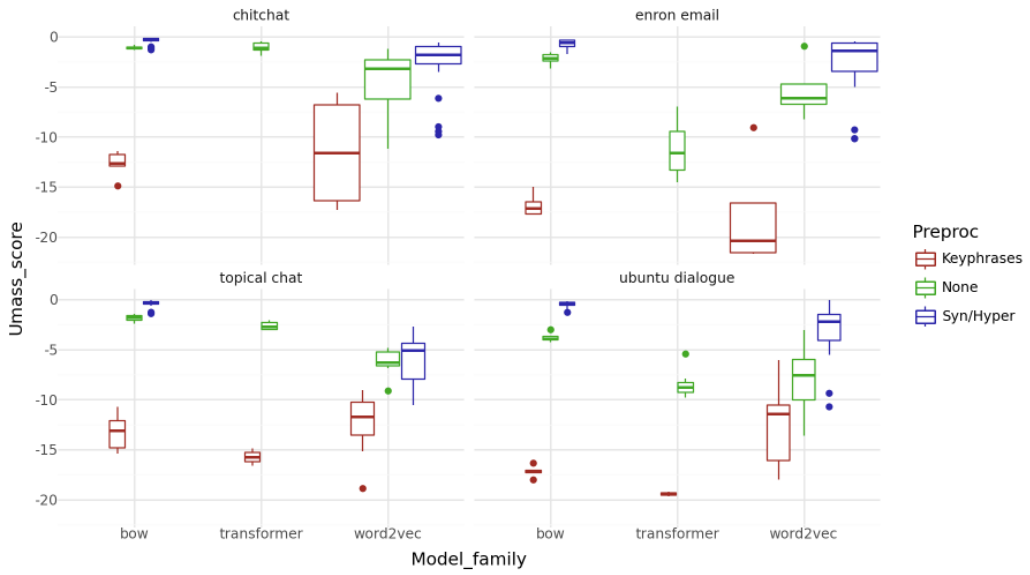


*Figure 12*—UMass coherence scores.

However, in the case of preprocessing with synonyms and hypernyms, both UMass and the PMI-based metrics exhibited the best scores. Figures 13 and 14 illustrate this for UCI and NPMI, respectively.

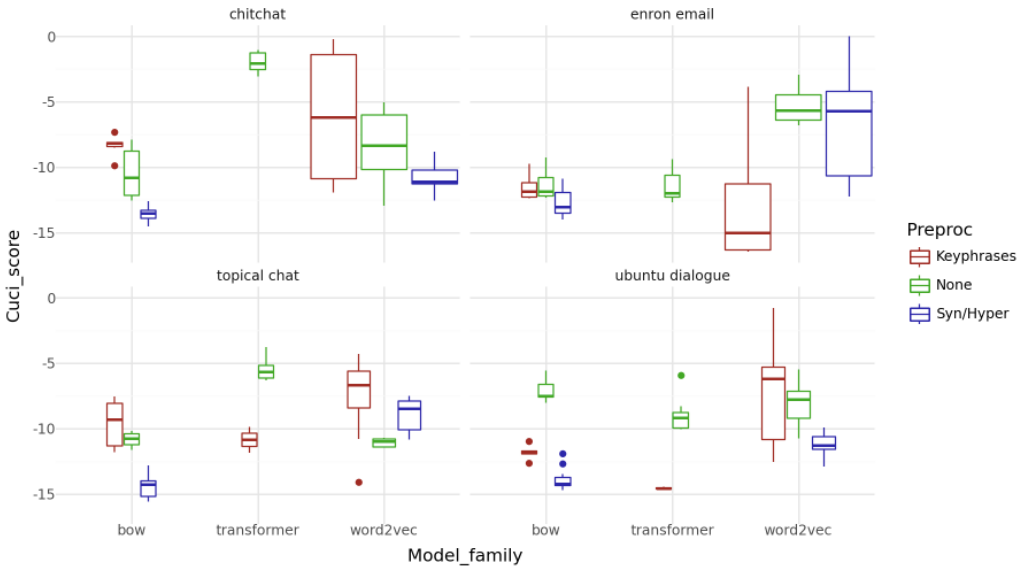


*Figure 13*—UCI coherence scores.

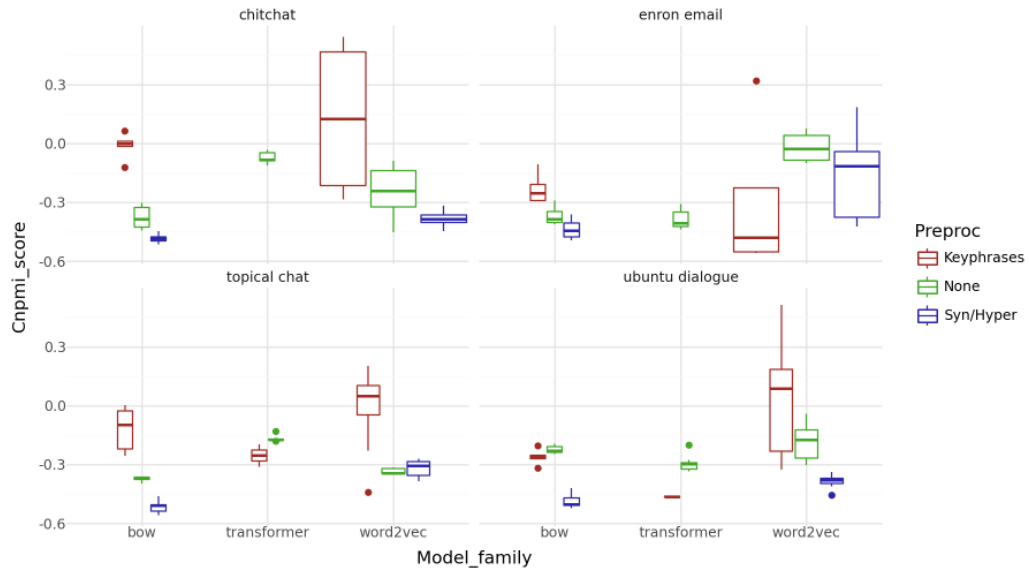


*Figure 14*—NPMI coherence scores.

UCI scores relative to NPMI were very similar; only the bag-of-words scores for the Enron Email dataset exhibited a visually obvious difference between scores, though the keyphrase median score remained higher in both cases. Therefore, these scores will be generalized in the discussion that follows as "PMI-based" or simply "PMI" scores.

PMI-based scores were split depending on the dataset and model family. Transformer-based models yielded the highest PMI scores for the Chitchat and Topical Chat datasets, while embeddings-based models exhibited better results for the Ubuntu Dialogue and Enron Email datasets. Bag-of-words models performed poorly among all four datasets in terms of PMI scores. These results may reflect the fact that both the Chitchat and Topical Chat datasets included a wide variety of topics, while the Ubuntu and Enron Email datasets included a much narrower focus. From this it is reasonable to conclude that transformer-based models may perform better on datasets that include a wide variety of topics, while the opposite would be true for datasets in which a narrow range of topics is delineated in a more nuanced fashion using embeddings-based models.

It is noteworthy that bag-of-words UMass scores were generally higher across all datasets. This is consistent with the fact that bag-of-words representations include a list of topic words, which are computationally more comparable against a corpus than a topic transformed into a shorter representation, albeit one that might be more semantically meaningful to a human.

In summary, it was found that coherence scores varied significantly depending on the dataset, the model family, and the coherence metric chosen. In general, we found that coherence scores had limited value in determining topic quality, but remained viable as a measure of how well individual topic words or phrases were represented within a corpus.

### 7.2 Semantic Quality

Semantic quality was gauged based on the results of the aforementioned Mechanical Turk survey of human workers, along with topics scored by the author. As mentioned, the quality of the Mechanical Turk results is somewhat questionable, as shown in Figure 11 which compares topics scored by the author to those scored by Mechanical Turk workers. An attempt was made to discard spurious results from the survey, first by detecting impossibly fast readers and second by examining how workers answered the "attention responses" included in the survey, i.e. those that were of either obvious good quality or obvious bad quality.

In the first case, the number of words in the conversation was divided by the worker's response time to yield a word-per-minute rate. If the worker appeared to read at a rate of at least five times the average rate of 238 words per minute, the worker was labeled as suspect. If the worker answered more than one question at this rate, all HITs from that worker were discarded. Only one worker was considered impossibly fast; 12 HITs from this worker were discarded and resubmitted for other workers to complete.

Similarly, HITs were discarded from workers who did not answer more than one attention response correctly. Responses were deemed to be incorrect if a topic representation of obvious poor quality was ranked 5 out of 5 or if a topic representation of obvious good quality was ranked 1 out of 5. Four workers were found to exhibit this pattern, and a total of 94 HITs were discarded from the survey and resubmitted for different workers to complete.

In all, 106 HITs out of 997 were discarded, a rate of 10.6%. Since this is a non-negligible rate—and in an effort to improve the quality of results—scores from both the Mechanical Turk workers and the author were factored when evaluating the semantic quality of topic representations.

It should be noted that workers who completed all the attention responses correctly at a reading rate consistent with an average human reader were issued a small monetary bonus. Also of note, Amazon Turk workers are often paid poorly for their work and must spend long hours completing HITs for small reward. For this reason, payment per HIT for this survey was calculated based on the average conversation length in words, the average reading rate, plus an allotment of ten seconds to respond to the question, such that the hourly rate per worker amounted to about $10 USD per hour.

Comparing the semantic quality of topic representations, it was unsurprising and expected that author-generated topics clearly scored the highest. These are shown in Figure 15 as "human_keywords" (list of author-generated keywords) and "human_friendly_topic" (a single, concise keyword or keyphrase describing the topic).

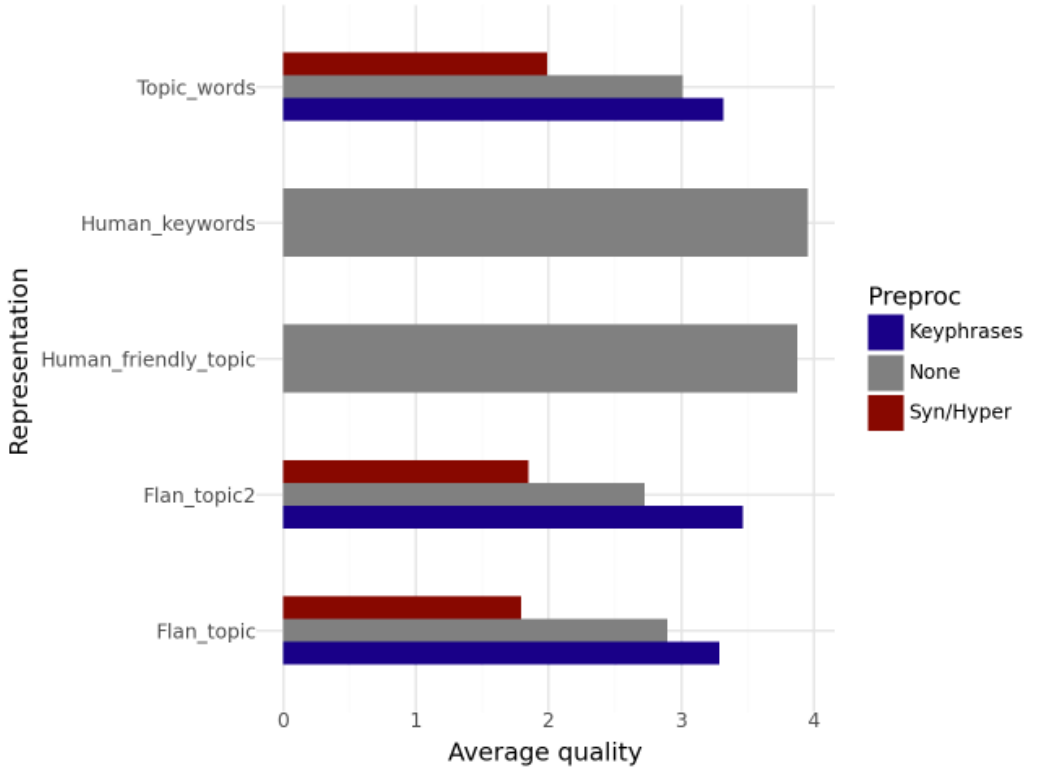


*Figure 15*—Semantic quality by topic representation.

Topic representations consisting of topic word lists or first- and second-round LLM-generated topics exhibited similar scores, with word lists not post-processed by an LLM performing slightly better. Analysis of variance (ANOVA) calculations yielded p-values that demonstrate the differences are statistically significant, as shown in Table 7.

*Table 7*—ANOVA results (topic representations).

| | Representation | Test_statistic | p-value |
|---|---|---|---|
| **0** | Flan_topic | 45.677318 | 1.046328e-18 |
| **1** | Flan_topic2 | 46.653834 | 4.560618e-19 |
| **2** | Topic_words | 37.623315 | 8.705815e-16 |

Conversely, conversations preprocessed with keyphrases were clearly superior in quality, with second-round LLM post-processing having the highest mean).

Perhaps surprisingly, conversations preprocessed with synonyms and hypernyms yielded the lowest-quality representations.

The best-performing family of models were generally the bag-of-words, followed by transformer-based and embeddings-based models (Figure 16). Across all model families, those that were preprocessed using keyphrases exhibited the highest semantic quality. As with topic representations, models preprocessed using synonyms and hypernyms performed relatively poorly.

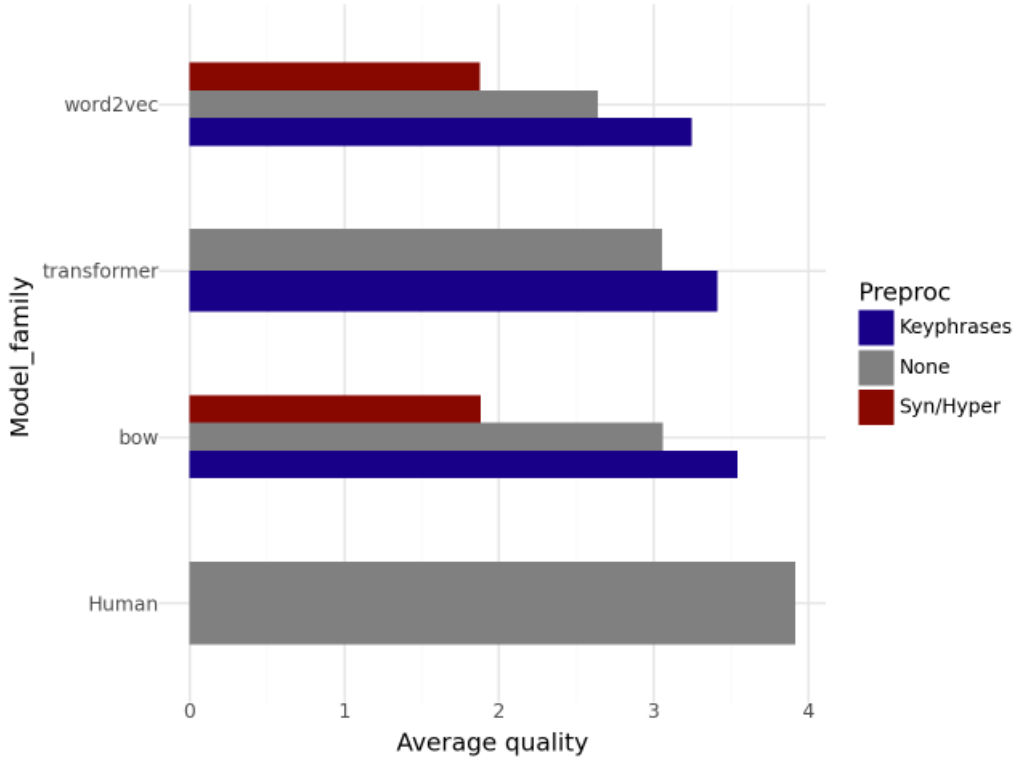


*Figure 16*—Semantic quality by model family.

ANOVA tests performed against the means indicate the differences are statistically significant (Table 8).

*Table 8*—ANOVA results (model family).

| | Model_family | Test_statistic | p-value |
|---|---|---|---|
| **0** | bow | 46.692614 | 1.391263e-18 |
| **1** | transformer | 2.840189 | 9.324405e-02 |
| **2** | word2vec | 58.974028 | 1.881218e-24 |

As shown in Figure 17, the time-tested LDA model performed the best on average, only slightly edging out Bertopic and Word2vec with K-means clustering. Again, conversations preprocessed with keyphrases yielded better-quality topic representations than those with no preprocessing. Conversations preprocessed with synonyms and hypernyms again performed poorly.

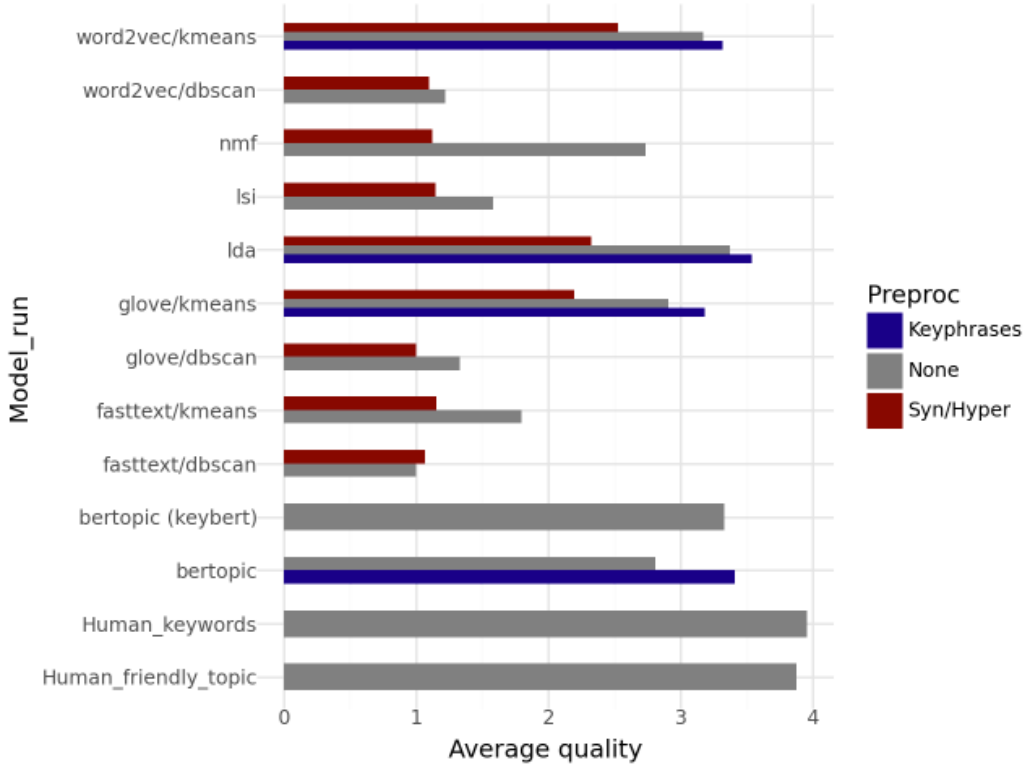


*Figure 17*—Semantic quality by model type.

ANOVA testing confirmed that the differences are statistically significant (Table 9).

*Table 9*—ANOVA results (model type).

| | Model_run | Test_statistic | p-value |
|---|---|---|---|
| **0** | bertopic | 6.990564 | 9.077359e-03 |
| **1** | fasttext/dbscan | 0.816327 | 3.716657e-01 |
| **2** | fasttext/kmeans | 7.897992 | 6.737341e-03 |
| **3** | glove/dbscan | 14.166667 | 6.335855e-04 |
| **4** | glove/kmeans | 13.372173 | 2.812509e-06 |
| **5** | lda | 20.436541 | 6.259944e-09 |
| **6** | lsi | 4.064094 | 5.110891e-02 |
| **7** | nmf | 43.336919 | 3.853563e-08 |
| **8** | word2vec/dbscan | 0.787939 | 3.793420e-01 |
| **9** | word2vec/kmeans | 8.651889 | 2.329921e-04 |

The semantic quality of topics did not appear to vary based on the number of topics generated by the model (Figure 18). This is expected in models which self-select the number of topics, but bag-of-words models and embedding models generated using K-means clustering were run using predefined topic counts of 15 and 30. This result might be explained by the fact that topic counts from most self-selecting models resulted in fewer than 15 topics. As shown in the figure, no datasets exhibited topic counts numbering being 15 and 30. Therefore, it is reasonable to conclude that our selection of 30 topics was too high to adequately delineate topics in most cases.

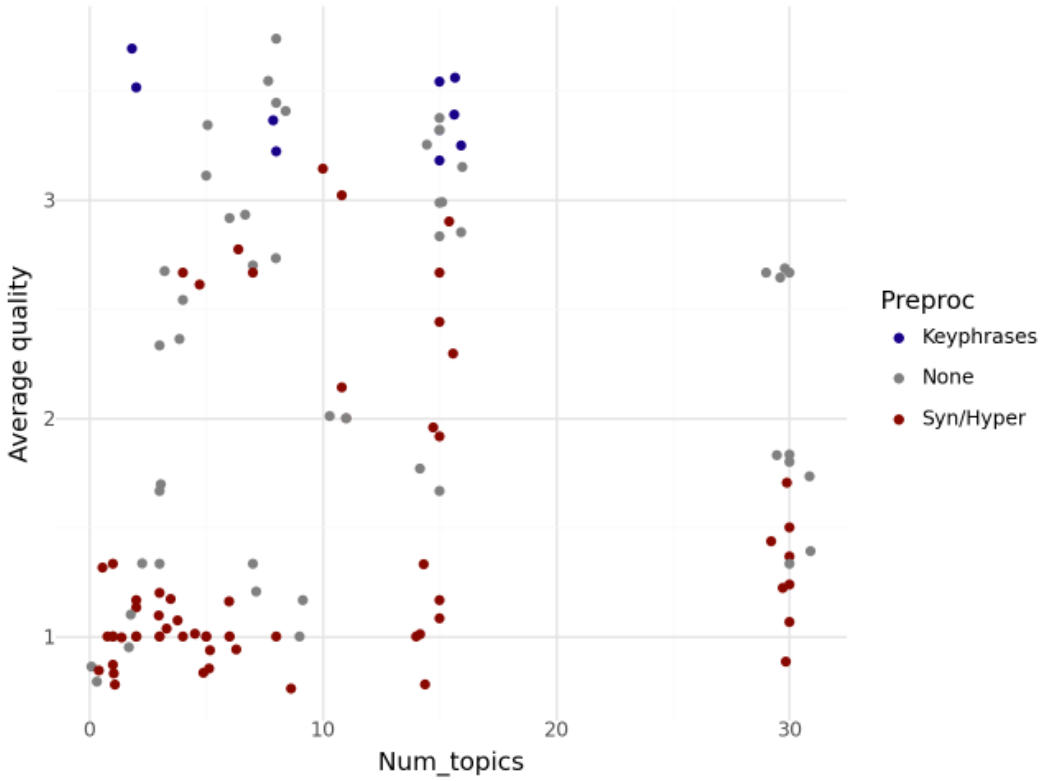


*Figure 18*—Semantic quality by number of topics.

### 7.3 Topic Relevance

The relevance of a topic representation to a conversation was gauged using the results of the Mechanical Turk survey along with author-scored results. Quality scores of 3 or greater were deemed relevant, while scores below that were considered irrelevant. This constituted a labeled dataset from which relevance could be predicted against the remaining (unlabeled) observations.

Various classification models were executed against the test set using 5-fold cross-validation, and accuracy scores were calculated (Table 10).

*Table 10*—Topic relevance modeling results.

| | Model | Best_params | Train_accuracy | Test_accuracy |
|---|---|---|---|---|
| **2** | lda | {'shrinkage': 0.0001, 'solver': 'lsqr'} | 0.776008 | 0.809877 |
| **4** | svm | {'C': 6.309573444801943, 'gamma': 0.01} | 0.778132 | 0.809877 |
| **0** | logistic | {'penalty': 'l2'} | 0.774947 | 0.797531 |
| **1** | naive_bayes | {'alpha': 0.0001, 'force_alpha': True} | 0.737792 | 0.775309 |
| **3** | knn | {'n_neighbors': 15} | 0.756900 | 0.767901 |
| **5** | tree | {'ccp_alpha': 1.0002302850208247, 'max_depth': 1, 'min_samples_leaf': 1, 'min_samples_split': 2} | 0.511677 | 0.511111 |
| **6** | random_forest | {'ccp_alpha': 1.0002302850208247, 'max_depth': 1, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100} | 0.511677 | 0.511111 |
| **7** | gradient_boost | {'ccp_alpha': 1.0002302850208247, 'learning_rate': 0.0, 'max_depth': 1, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100, 'subsample': 0.1} | 0.511677 | 0.511111 |

While both SVM and LDA yielded the highest identical accuracy score against the test set (81.0%), LDA was selected due to its far shorter run time.

Based on the binary relevance values scored for the test set, a ROC curve was generated (Figure 19). The area under the curve (AUC) was calculated to be 0.860.

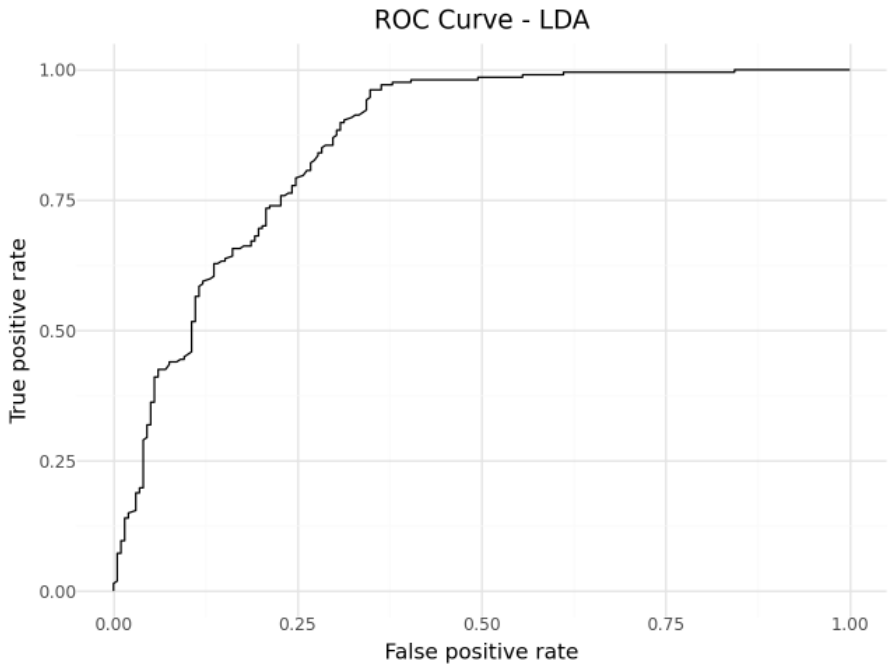


*Figure 19*—ROC curve for topic relevance LDA model.

To select an appropriate probability threshold to distinguish relevant from irrelevant topics, an approach was taken that would yield the same approximate number of relevant topics as were represented in the training dataset. This constituted about 51.2% of the labeled observations. Figure 20 shows the percentage of relevant observations in the entire dataset versus probability threshold.

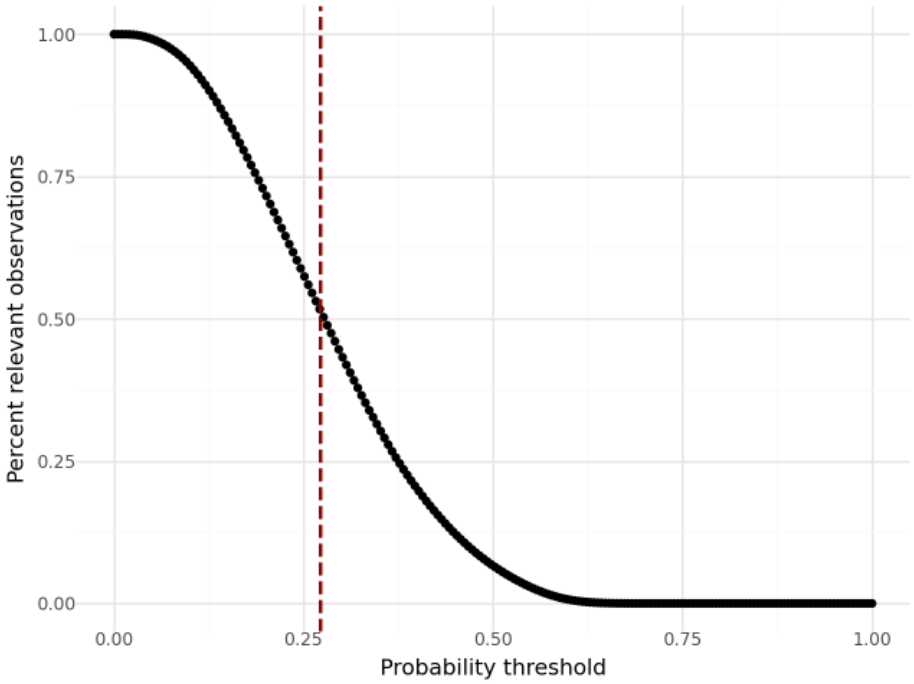


*Figure 20*—Percentage of relevant observations versus probability threshold.

This threshold yields a true positive rate (TPR) of approximately 96.5%, and a false positive rate (FPR) of about 35.0%. Along with the accuracy score of 81.0%, the TPR and FPR were considered adequate for further predictions, as was the AUC of 0.860. Relevance scores were predicted for the remaining (unlabeled) portion of the dataset using a probability threshold of 0.2725, based on 51.2% of observations labeled as relevant.

Predicted relevance results varied significantly from those from the semantic quality survey. Figure 21 illustrates this by showing the probability of relevance by topic representation, grouping by the type of preprocessing performed. In every case, conversations preprocessed with synonyms and hypernyms were more likely to yield relevant topic representations than those preprocessed with keyphrases or those that were not preprocessed.

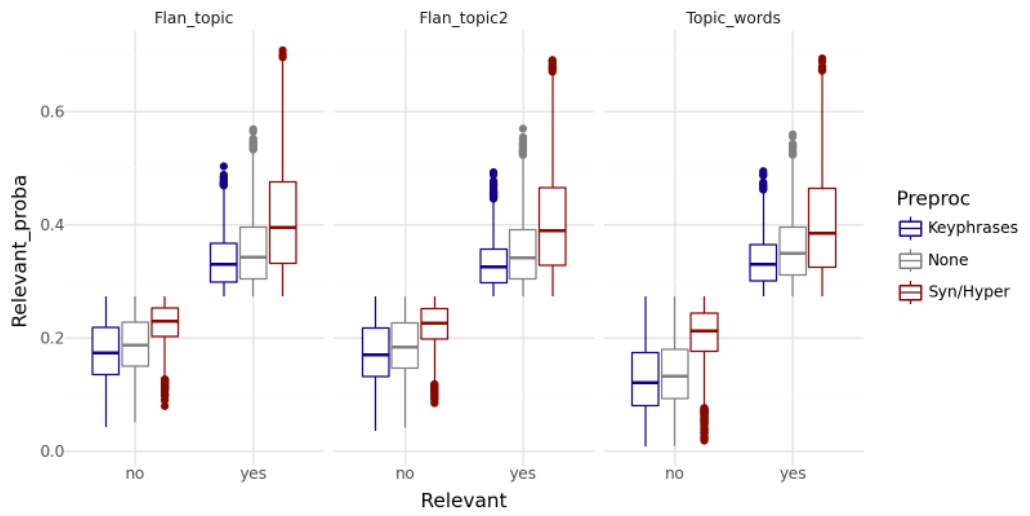


*Figure 21*—Probability of relevance vs. topic representation.

Similarly, synonym/hypernym preprocessing outperformed other preprocessing methods across all model types (Figure 22), with the exception of transformer-based models in which no synonym/hypernym preprocessing was performed.

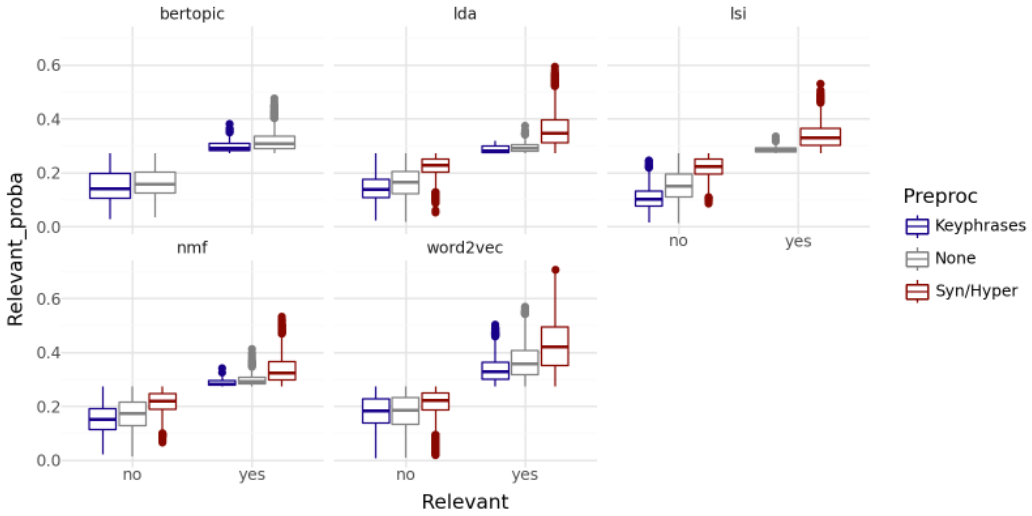


*Figure 22*—Probability of relevance vs. model type.

The same trend was true across all four datasets, as shown in Figure 23.

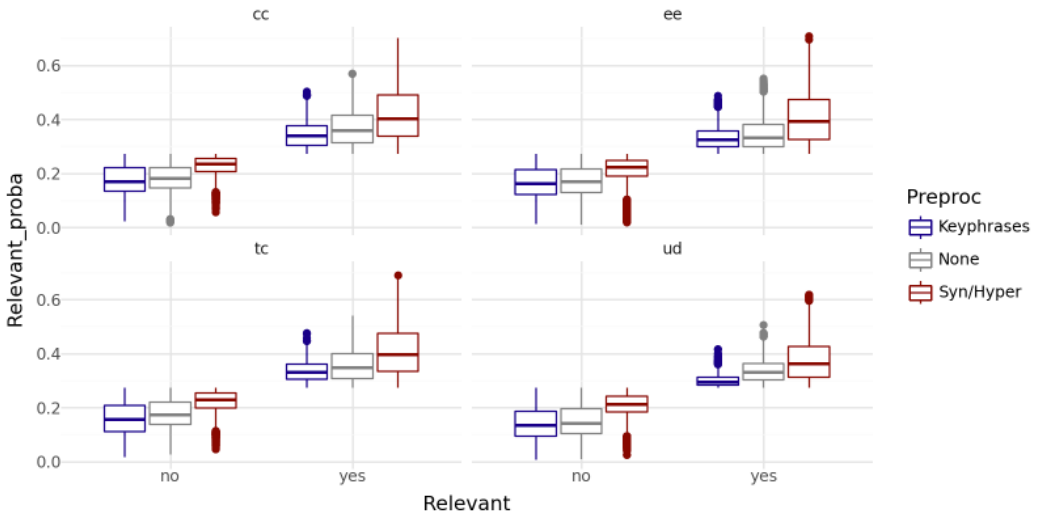


*Figure 23*—Probability of relevance vs. dataset.

While the difference in results between topic relevance and semantic quality may appear counterintuitive, the cause of this discrepancy can be accounted for in several ways. First, the conversations selected for the semantic quality survey in comparison with the rest of the corpus of conversations may have underrepresented the entire corpus of conversations. Despite an attempt being made to choose representative conversations, the wide range of topics and conversation lengths made the success of this endeavor difficult to gauge. Additionally, resource constraints led to the inclusion of only a small subset of models in the survey. As a result, many combinations of model runs were excluded (for example, Fasttext, LSI, and NMF). The exclusion of these observations from the survey undoubtedly led to results biased in favor of those that were in the survey. Further, it was earlier noted that the Mechanical Turk survey results themselves may be questionable, as shown in Figure 11 which compares Mechanical Turk semantic quality scores with those manually scored by the author. Finally and probably most significant was the fact that cosine similarity was included as a model feature; this had the effect of overemphasizing the importance of similar words added to conversations during preprocessing by synonyms and hypernyms, which were then later compared with topic representations that likely included some of these added words.

Despite the difference, several conclusions can be made based on the modeling results. First, topics that were postprocessed with first-round LLM-based text generation performed slightly better than those with second-round or no postprocessing (Figure 21). Second, embedding-based models performed notably better than transformer or bag-of-words models (Figure 22). Third, there was no significant difference in performance among datasets, with the exception of the Ubuntu Dialogue dataset, against which models performed only slightly more poorly (Figure 23).

### 7.4 Model Runtime

Runtimes in seconds were recorded for each model run, including preprocessing and topic modeling. Postprocessing was not included in the runtime calculation as this step was performed separately. Figure 24 shows runtime durations by model type, split by how the dataset was preprocessed.

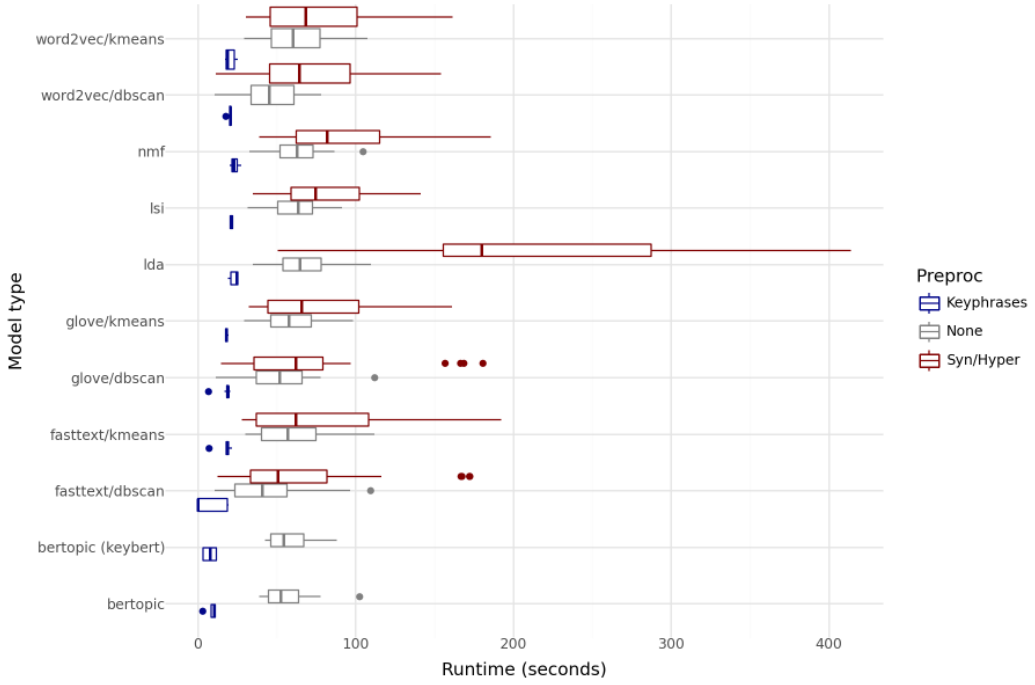


*Figure 24*—Runtime vs. model type.

As shown, adding synonyms and hypernyms greatly increased the runtime, especially for LDA model runs. Preprocessing with keywords actually reduced

runtimes, but this was expected since this type of preprocessing tends to reduce the overall size of each conversation within a corpus. Embeddings-based models without preprocessing ran somewhat faster than bag-of-words models; however, it should be noted that loading large pretrained models into memory was time-consuming and was not factored into the runtime calculations shown above.

## 8 CONCLUSIONS

Several conclusions can be made based on the coherence metrics, semantic quality survey, and relevance modeling results.

In terms of topic coherence, PMI-based scores (UCI and NPMI) appear to be the most appropriate coherence scores to use due to its reliance on PMI as a metric. Word2Vec models preprocessed with keyphrases yielded the best overall PMI scores.

The best model in terms of semantic quality was LDA preprocessed with keyphrases, with no postprocessing performed.

Word2Vec-based models preprocessed with synonyms and hypernyms and post-processed with a single round of LLM text generation exhibited the best topic relevance.

Comparing results across datasets, coherence results varied somewhat depending on whether the topic range was wide (as in the Chitchat and Topical Chat datasets) or more narrowly focused (as in the Enron Email and Ubuntu Dialogue datasets). There was only slight variation in topic relevance for the Ubuntu Dialogue dataset, while the rest yielded consistent results.

As expected, preprocessing with synonyms and hypernyms was time-consuming and resource intensive, as was postprocessing with LLM-based text-generated representations. In addition, embeddings-based topic models executed in the unsupervised modeling stage took the longest to load into memory, followed by LLM-based models. Conversely, traditional bag-of-words models ran in a relatively short timeframe, especially those preprocessed with keywords.

Therefore, these factors should be taken into consideration if time and/or machine resources are a concern.

Because each evaluation technique yielded different results, it is difficult to recommend a single best approach that would apply to every investigation. However, disregarding the results of topic relevance, a reasonable conclusion can be made that preprocessing using keyphrases yields the best results overall, along with either LDA or Word2Vec topic modeling without postprocessing. If time and machine resources pose a constraint, LDA might be preferred over Word2Vec with only a slight degradation in coherence metrics.

To refine these results, several areas of further study are possible. First, additional topic modeling might be performed using a greater number of parameters. For example, the topic or cluster size might include others besides the 15 or 30 run in this study. For embedding-based models, the number of vectors might be varied (we used 200). Additionally, the embeddings could be trained on the corpus of conversations rather than using pretrained embeddings as we did in this study. The parameters used with DBSCAN clustering might also be refined to work with all datasets such that fewer empty clusters were formed, obviating the need to discard the results, as was the case in this study.

To improve the semantic quality results, the Mechanical Turk survey could be expanded to include more model runs, though at a higher monetary cost. Further study is also warranted to evaluate the quality of Mechanical Turk survey results in general, and other ways to gauge the quality of workers' answers might be employed. Services other than Mechanical Turk might also be investigated.

Because of the confidential nature of most criminal investigations and the lack of publicly available datasets of this nature, it was only possible to perform this study using datasets that approximate the type of forensic evidence obtained during criminal investigations. As such, it would be useful to evaluate the results of this study against datasets from real-world criminal investigations.

## 9 REFERENCES


1. Aletras, N. & Stevenson, M. (2013). Evaluating Topic Coherence Using Distributional Semantics. In Proceedings of the 10th International Conference on Computational Semantics (pp. 13–22).
2. Alnusyan, R., Almotairi, R., Almufadhi, S., Shargabi, A., & Alshobaili, J. (September 2020). A Semi-Supervised Approach for User Reviews Topic Modeling and Classification. Institute of Electrical and Electronics Engineers Inc. 2020 International Conference on Computing and Information Technology, ICCIT 2020.
3. Angelov, D. (August 2020). Top2Vec: Distributed representations of Topics.
4. Barrus, T. (2024). Pyspellchecker 0.8.1. Python Package Index, https://pypi.org/project/pyspellchecker/.
5. Blei, D., Ng, A., Jordan, M., & Lafferty, J. (January 2003). "Latent Dirichlet allocation". Journal of Machine Learning Research. 3: 993–1022.
6. Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2016). Enriching Word Vectors with Subword Information. In arXiv:1607.04606 [cs.CL].
7. Bouma, G. (2009). Normalized (Pointwise) Mutual Information in Collocation Extraction.
8. Brownlee, J. (August 2019). "What Are Word Embeddings for Text?". Deep Learning for Natural Language Processing, https://machinelearningmastery.com/what-are-word-embeddings/.
9. Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171-4186, Minneapolis, Minnesota. Association for Computational Linguistics.
10. de Waal, A., Venter, J., & Barnard, E. (2008). Applying Topic Modeling to Forensic Data. In: Ray, I., Shenoi, S. (eds) Advances in Digital Forensics IV. DigitalForensics 2008. IFIP — The International Federation for Information Processing, vol 285. Springer, Boston, MA.

11. Enron Corp & Cohen, W. W. (2015) Enron Email Dataset. United States Federal Energy Regulatory Commissioner, comp [Philadelphia, PA: William W. Cohen, MLD, CMU] [Software, E-Resource] Retrieved from the Library of Congress, https://www.loc.gov/item/2018487913/.
12. Google, "FLAN-T5-base". (2024). Hugging Face, https://huggingface.co/google/flan-t5-base.
13. Gopalakrishnan, Karthik, et al. (2019). "Topical-Chat: Towards Knowledge-Grounded Open-Domain Conversations.", Proc. INTERSPEECH, https://github.com/alexa/Topical-Chat.
14. Grootendorst, M. (2022). KeyBERT. https://maartengr.github.io/KeyBERT/.
15. Grootendorst, M. (2022). BERTopic: Neural topic modeling with a class-based TF-IDF procedure. In arXiv preprint arXiv:2203.05794.
16. Hadiate, A. (2022). Topic Modeling Evaluations: The Relationship Between Coherency and Accuracy. Rijksuniversiteit Groningen, https://fse.studenttheses.ub.rug.nl/28618/.
17. Hofmann, T. (1999). "Probabilistic Latent Semantic Indexing". Proceedings of the Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrieval.
18. Hughes, M., Hope, G., Weiner, L., McCoy, T., Perlis, R., Sudderth, E., & Doshi-Velez, F. (2018). Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics, PMLR 84:1067-1076.
19. Krasnashchok, K. & Jouili, S. (2018). Improving Topic Quality by Promoting Named Entities in Topic Modeling. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 247-253.
20. Kulkarni, M., Mahata, D., Arora, R., and Bhowmik, R. (2021). "Learning Rich Representation of Keyphrases from Text." arXiv preprint arXiv:2112.08547.
21. Li, W. & McCallum, A. (2006). "Pachinko allocation: DAG-structured mixture models of topic correlations". Proceedings of the 23rd international conference on Machine learning - ICML '06. pp. 577–584.

22. Lowe, R., Pow, N., Serban, I., & Pineau, J. (September 2015). The Ubuntu Dialogue Corpus - A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems. Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 285-294.
23. McAuliffe, J.D. & Blei, D., Supervised Topic Models (March 2010).
24. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. In arXiv:1301.3781 [cs.CL].
25. Mimno, D., Wallach, H., Talley, E., Leenders, M., & McCallum, A. (2011). Optimizing Semantic Coherence in Topic Models. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (pp. 262–272).
26. Ml6team, "Keyphrase Extraction Model: KBIR-inspec" (2022). Hugging Face, https://huggingface.co/ml6team/keyphrase-extraction-kbir-inspec.
27. Motro, Y. (August 2023). "The Current State of Large Language Models (LLM)". tas.ai, https://www.tasq.ai/blog/large-language-models/.
28. Myers, W., Etchart, T. & Fulda, N. (2020). Conversational Scaffolding: An Analogy-Based Approach to Response Prioritization in Open-Domain Dialogs.
29. Newman, D., Lau, J. H., Grieser, K., & Baldwin, T. (2010). Automatic Evaluation of Topic Coherence. In the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Pages 100–108.
30. Papadimitriou, C., Raghavan, P., Tamaki, H., & Vempala, S. (1998). "Latent semantic indexing". Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems - PODS '98. pp. 159–168.
31. Pennington, J., Socher, R., & Manning, C.D. (2014). GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing ({EMNLP}).
32. Röder, M. (2018). "Not being able to replicate coherence scores from paper." Github, https://github.com/dice-group/Palmetto/issues/13.

33. Röder, M., Both, A., & Hinneburg, F. (February 2015). Exploring the Space of Topic Coherence Measures. WSDM '15: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, pp. 399-408.
34. Rosner, F., Hinneburg, F., Röder, M., Nettling, M., & Both, A. (March 2014). Evaluating topic coherence measures. Presented at the "Topic Models: Computation, Application and Evaluation" workshop at the "Neural Information Processing Systems" conference 2013.
35. Sahrawat, D., Mahata, D., Zhang, H., Kulkarni, M., Sharma, A., Gosangi, R., Stent, A., Kumar, Y., Ratn Shah, R., and Zimmermann, R. (2020). "Keyphrase extraction as sequence labeling using contextualized embeddings." In European Conference on Information Retrieval, pp. 328-335. Springer.
36. Sarkar, D (2019). *Text Analytics with Python: A Practitioner's Guide to Natural Language Processing*. Apress, Bangalore, India.
37. Scholar Within, "Average Reading Speed by Age and Grade Level." (2024). Scholar Within, https://scholarwithin.com/average-reading-speed.
38. United States Securities and Exchange Commission v. Andrew S. Fastow (2002). United States District Court, Southern District of Texas, Houston Division, October 2, 2002. https://www.sec.gov/litigation/complaints/comp17762.htm.
39. Verma, R. (2017). SMS/Message_Slang_Translator, Github, https://github.com/rishabhverma17/sms_slang_translator.