E-Comm 911 Capstone Report | Natural Language Processing to Help Save Lives and Protect Property

Frank Lu, Jaekeun Lee, Jacky Ho

Advisor: Varada Kolhatkar

Project partner: E-Comm 911

Executive Summary

Call takers at E-comm 911 have a very intense work environment. While every second counts, they rely on the instructions in SOP documents to take proper actions. However, SOP documents have been one of the bottlenecks in 911 call procedures due to the redundancy in the instructions and the inefficient design of the lookup system. In this project, we adopted NLP (Natural Language Processing) techniques to identify similar SOP instructions for consolidation. We also generate corresponding procedure flowcharts to provide a clear visual guide to users. On top of the NLP models and procedure flowcharts, we also created a web app through which users can select a desired subset of SOP documents and view the flowcharts in real-time.

Introduction

E-Comm 911's operators handle tons of emergency calls every day for different kinds of events, such as dog bites and domestic violence, in various jurisdictions/agencies; for each event, they have to follow a set of standard operating procedures (SOP) to designate the appropriate actions. Each SOP document handles one event type for one agency, depicting its associated scenarios and their corresponding actions. One event type could have many SOP documents for different agencies.

The main problems E-Comm 911 is facing regarding its SOP documents are redundancy and haphazardness. SOP documents from different agencies for the same event could be remarkably similar with trivial differences. For example, two agencies can handle firearm incidents the same way semantically but have separate SOP documents with slightly divergent wordings. Moreover, the operators can conceptualize the documents through their groupings and relationships. Due to these problems, there is ample room for efficiency improvement in call handling, and there is a high turnover rate in operators.

This project aims to address these issues with various data science techniques. The main goal is to identify similarities and differences among documents for potential consolidation and better

visualize documents for more condensed understanding.

Description of the Data

For our analysis, E-Comm 911 provides us with 2021 SOP documents from 18 police agencies. These documents concern over 280 events, such as assault, missing persons, and domestic violence. These documents are all Microsoft Word files. (Fig 1)

Data Science Methods

The data science methods are divided into two portions: modelling and visualization. The modelling portion goes through the texts and determines which documents are similar enough to be placed in the same cluster. However, the resulted clusters can contain many documents and are hard to interpret, so we visualize them by converting them into map-like procedural flowcharts for easier understanding.

Modeling

We have looked into document clustering using LDA, HDP, and KMeans. LDA and HDP are topic modelling algorithms. The idea is that, when several documents are said to be under the same topic, the wordings of these documents should be similar to each other. This can be a valuable starting point for text consolidation. Kmeans, on the other hand, has long been used for clustering tasks. We tuned hyper-parameters of these 3 models based on their objective measurements, and the 3 tuned models will be compared based on human interpretation .

LDA from Gensim was the first algorithm we tried. In order to find the best LDA model, we tuned two of the hyper-parameters:

- alpha: document level distribution of topics
- num_topics: number of topics

Here, coherence score was used as the objective measurement of model performance. Coherence score has many variants, and we chose “c_v” and “npmi” because they tend to correlate more with human judgment by several research papers [1,2]. As depicted in Fig 2, both “c_v” and “npmi” agree that the desired range of num_topics for this model should fall between 80 and 90.

HDP from Gensim is also a topic modeling algorithm and an extension of LDA. HDP can infer the number of topics and doesn’t need the num_topics hyperparameter. However, HDP has more hyperparameters to be considered:

- alpha: document level distribution of topics
- gamma: corpus level distribution of topics
- T: upper bound of number of topics
- K: upper bound of number of words inside a topic

Because the number of topics is inferred by HDP, we tried tuning other hyperparameters, and they don't seem to make a significant change in coherence scores.

KMeans is a conventional clustering algorithm, and we use the implementation from Scikit-Learn. For KMeans, we mainly focus on the n_clusters hyperparameter, which controls the number of clusters. There are two metrics that we use to tune n_clusters objectively:

- inertia: the distance between cluster members and cluster centroid
- Silhouette score: the distance between different clusters

As displayed in Fig 3 and Fig 4, the inertia and Silhouette scores both indicate that a range between 80 and 90 is desirable for n_clusters.

After obtaining 3 tuned models, we use human interpretation to compare the results of clustering. With the goal of finding similar texts for consolidation, KMeans generates more reasonable clusters and is selected as the final model. The output of clustering is a data frame similar to Fig 6. Please note a sample data frame is displayed here because the real SOP texts are confidential.

Visualization

Although the clustering models generate reasonable results, the resultant table flattens out the hierarchy of SOP instructions. To highlight the hierarchy of 911 call procedures, we convert the result of clustering to graph-like flowcharts, which is shown in Fig 2.4. We also apply TF-IDF and cosine similarity to merge similar nodes in the flowcharts, so they don't look crowded.

Data Product

The final data product consist of two parts: pipeline and dashboard.

Pipeline

In order to make conversion of docx files automatic and the analysis reproducible, our team came up with a pipeline that identifies the highly hierarchical pattern of E-comm's SOP documents. The pipeline conversion of SOP documents will be viable as long as they have a similar structure. The caveat here is that SOP documents need to be structurally aligned to produce a consistent result. The pipeline has three features in total. First, it converts SOP documents into a machine readable format. The processed documents are stored as comma separated version (csv) files so that they can be used later for other purposes. Then, the pipeline processes the csv files into TF-IDF vectors and performs K-Means Clustering. The results of clustering is also stored as csv files. Finally, SOP situations and event types that have been clustered together are merged as visualized flowcharts.

Dashboard

One of the difficulties of unsupervised learning is interpretability and evaluation of model. Since

the data is not labeled, it is challenging to assess the quality of clusters, especially when the model outputs a large number of them. As our team took advantage of the nested hierarchical structure in the SOP documents, we decided to generate a graph-like flowchart to represent the procedure of 911 calls, which is shown in Fig 2.4. To present and combine all the results, we devised a dashboard which has two sections: clustering results with visualized flowchart and search bar to look up SOPs using keywords.

In some clusters, more than one event type exists, suggesting the possibility of consolidation between the two event types. We hope our partners can discover the redundant elements within SOPs using this section. A search bar was appended at the bottom of the dashboard to facilitate SOP document search. Using the keyword, users can locate the filename and contents of SOP document that has the largest cosine similarity.

Limitation and Recommendation

Our data product has several limitations.

First of all, some of the clustering results lack interpretability. Most clusters make sense, but some clusters don't really make sense semantically. Because we have a large number of clusters, it is hard to evaluate them one by one. Finding the ones that lack interpretability should be manually done by browsing through each cluster numbers.

Secondly, we haven’t been able to parse actions/ conditions and questions in the sop instruction part. Most documents adhere to a consistent and hierarchical structure in terms of sentence formats and their implications, but some stray away from it. Since the parsing and analysis heavily relied upon the hierarchy of contents, inconsistent structure was a huge problem that couldn't be resolved. This problem gets more complex when it involves questions, reference to other resources, because we couldn't ensure definite order or hierarchy in these instructions.

Finally, some of the large complex SOP flowcharts lack visibility. The large nodes that contain multiple line of sentences, makes the graph size too big to be contained in one page. This problem is relevant to the previous limitation of being unable to parse the instructions on the SOP to granular levels

In order to overcome this limitations, we propose the following approaches. Instead of making use of the document structure to parse texts, we can identify named entities in the text and find similar documents based on the named entities. The team developed a rule-based entity extraction feature within the pipeline, so the E-comm team can expand the work from there.

Appendix

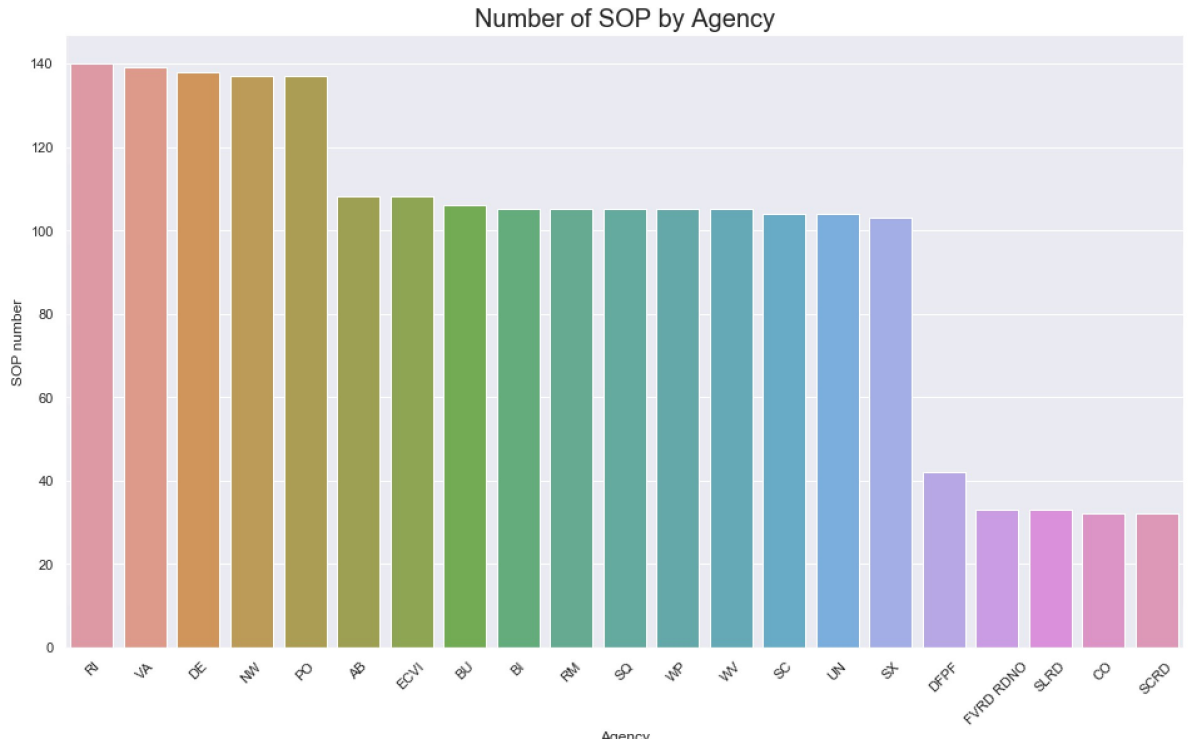

Fig 1 - Number of SOPs by agency

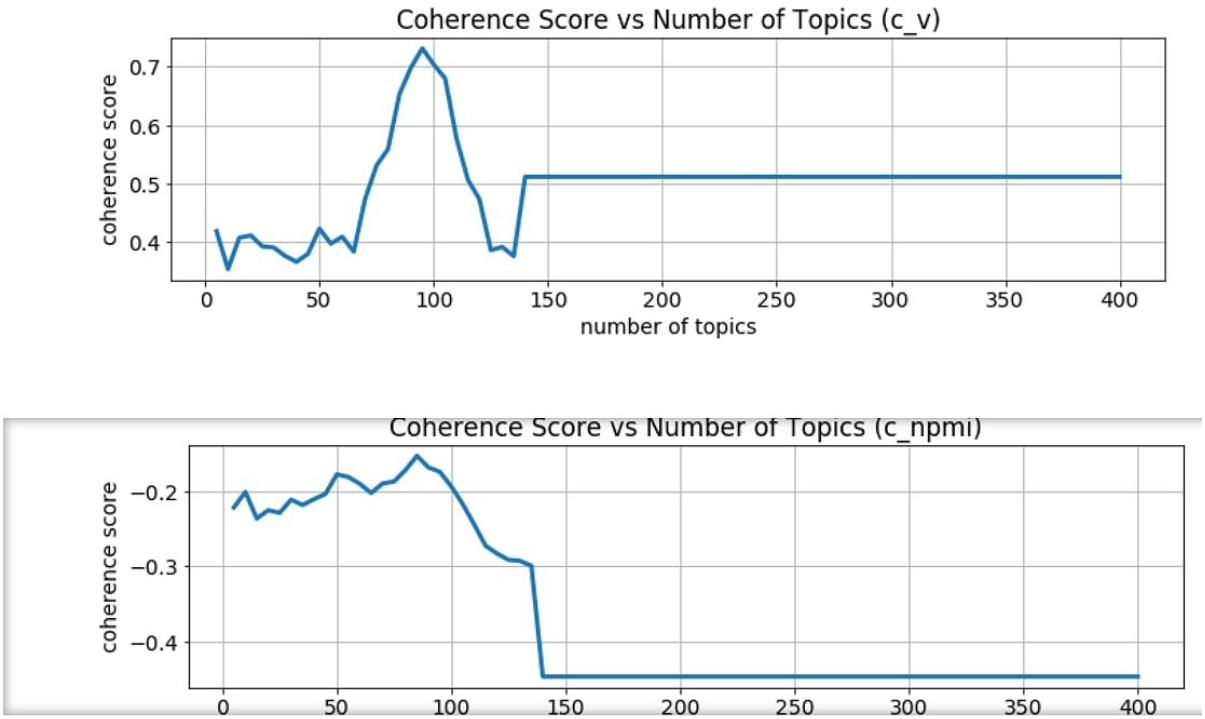

Fig 2 - Coherence scores vs num_topic of LDA model

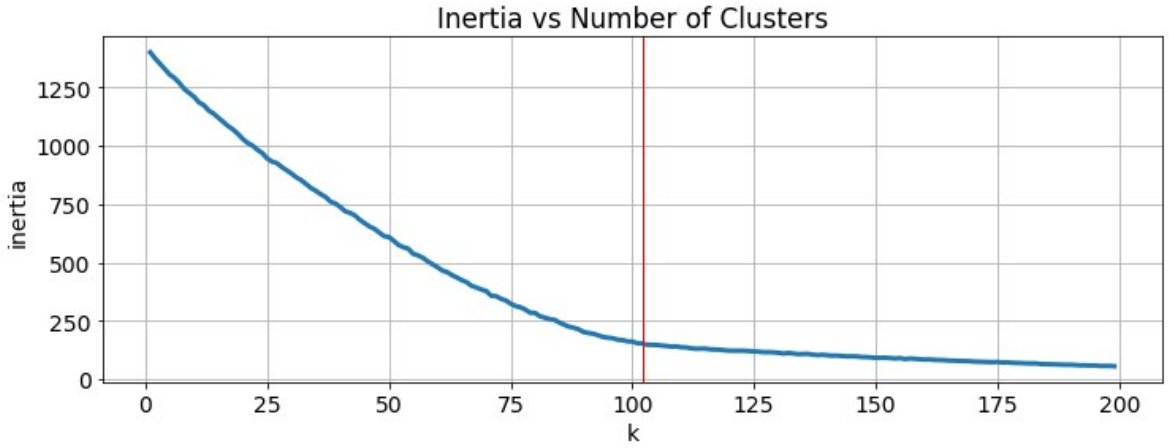

Fig 3 - Inertia Score vs n_clusters of K-Means

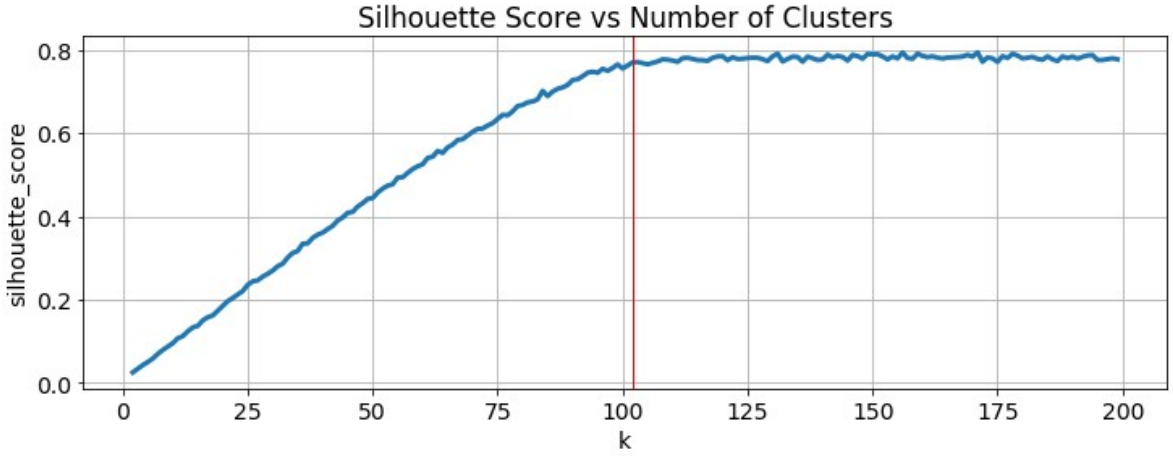

Fig 4 - Silhouette Score vs n_clusters of K-Means

Type	Juri	Situation	Sop	filename	cluster
ICRM	LL	ice cream drop	create a call	ICRM-LL.docx	27
ICRM	WL	ice cream robbery	create a call	ICRM-WL.docx	27
ICRM	DL	melted ice cream	refer to clean up	ICRM-DL.docx	27
ICRM	SL	ice cream soldout	refer to clean up	ICRM-SL.docx	27

Fig 5 - Sample output of clustering

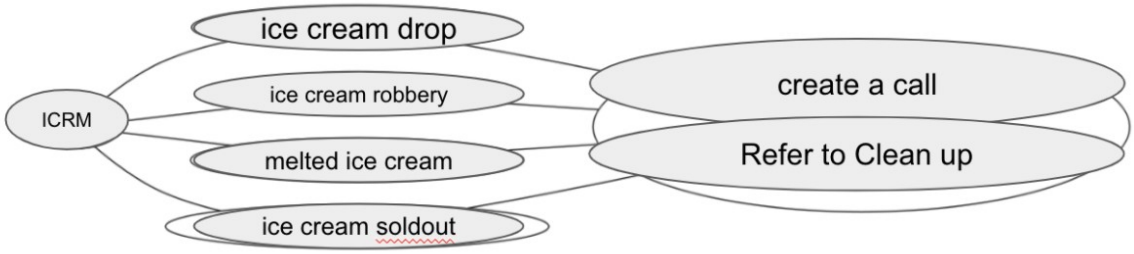

Fig 6 - Sample procedure flowchart

Reference

1. Linzi Xing, Michael J. Paulz, and Giuseppe Carenini. (2019). Evaluating Topic Quality with Posterior Variability. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3471-3477. https://www.aclweb.org/anthology/D19-1349.pdf

2. Michael Roder, Andreas Both, and Alexander Hinneburg. (2015). Exploring the space of topic coherence measures. *Eighth ACM International Conference on Web Search and Data Mining*, pages 39–408. https://dl.acm.org/doi/pdf/10.1145/2684822.2685324