Concept overview

Objective

Use ML to assist qualitative analysis researchers in efficiently uncovering latent semantic groupings within text datasets that are in strong accord with their notions of relevancy for a given research objective.

Why off the shelf clustering doesn't work

Off-the-shelf clustering algorithms fall short of meeting the above goal primarily because the clustering criteria are not in accord with the researchers' notions of relevancy in a given context. When applied to representations of text, clustering algorithms often surface shallow relationships between examples such as occurrence of specific proper nouns. The researcher, on the other hand, might care about much more subtle relationships. Moreover, off the shelf algorithms are blind to the research context. For example, given a dataset of open-ended survey questions from college students commenting on their remote learning experience, a researcher may be interested in the psychological impact of remote learning on individuals, on the various technological challenges the students faced, or something else entirely. Off-the-shelf algorithms have no mechanism for adapting to these contexts, but produce the same results for each case.

How to incorporate researcher feedback?

To overcome these shortcomings, we introduce a mechanism for incorporating user judgement into the process for representation of text prior to clustering. Since the researcher does not know *a priori* which class labels (schema, codebook, lexicon) are needed to describe the data, this feedback cannot naturally take the form of labeling individual examples. This points to the use of similarity (or dissimilarity) between samples as a possible source of researcher feedback. However, in the context of qualitative analysis, the subjective nature of judgements about text make it difficult to consistently answer questions of the form, "Is A similar to B?". The researcher does not have a well-defined similarity metric with which to answer the query. How similar must A and B be in order to answer "Yes"? For pairwise feedback to work, this ambiguity must be resolved and carefully implemented by all users. Moreover, choices made in order to resolve this ambiguity will tend to be arbitrary and hard to justify especially in contexts where subjective judgements about language are involved. Assuming the criteria can be well-defined, the researcher still has the challenge (or impossibility?) of "calculating" the degree of similarity between new pairs of examples in order to apply the criteria. In some cases, these challenges may not be so onerous, but we expect these cases to also lend themselves to simple *a priori* classification schemes.

Many of the challenges associated with pairwise feedback can be avoided by considering instead triplet feedback as answers to queries of the form, “Is A more similar to B than C?”. The crux of the problem with pairwise feedback was the difficulty in answering the question, “How similar must A and B be in order to be considered `similar’?”. The triplet formulation answers this by saying A must simply be more similar to B than to C. Of course this does not completely remove ambiguity because the user may still not have the knowledge needed to answer the query. However, the presence of the 3rd reference example greatly alleviates the need to precisely define the notion of similarity between two examples in advance. Also, when answering a specific query, the “calculation” the researcher must do is greatly simplified.

Active learning

Ultimately, we would like to arrive at a set of coherent classes that characterize the relevant information contained in the dataset within the given research context. Ordinarily, qualitative analysis researchers would sample the dataset and manually inspect many instances to arrive at a set of codes. For the ML approach to make any sense, it should be efficient in terms of the number of queries that the researchers must answer to arrive at the final set.

Active learning is a technique for sampling examples for coding (in this case sampling triplets for answering) in a smart way based on the current state of the model in order to maximally reduce the model’s uncertainty (vaguely defined for now) at each step. There are many possible ways to sample that will be explored in this study.

Pre-training

Like active learning, pre-training is an efficiency measure. The goal is to find a completely unsupervised representation of the input text prior to the active learning process. By first training an encoder on an unsupervised task such as language modeling or the BERT objectives, we can reduce the burden on the model when attempting to learn the similarity structure.

Schematic

The schematic below shows all of the conceptual pieces of the assisted qualitative analysis pipeline. We will briefly describe each of these components in turn. Fully specifying and implementing these components is the primary objective of this project.

Data: Provide the input corpus consisting of a set of text documents.

Pretrain: Using an unsupervised objective, encode the documents in the input data. The encoder will be used to initialize training at the first iteration of the "Train" step.

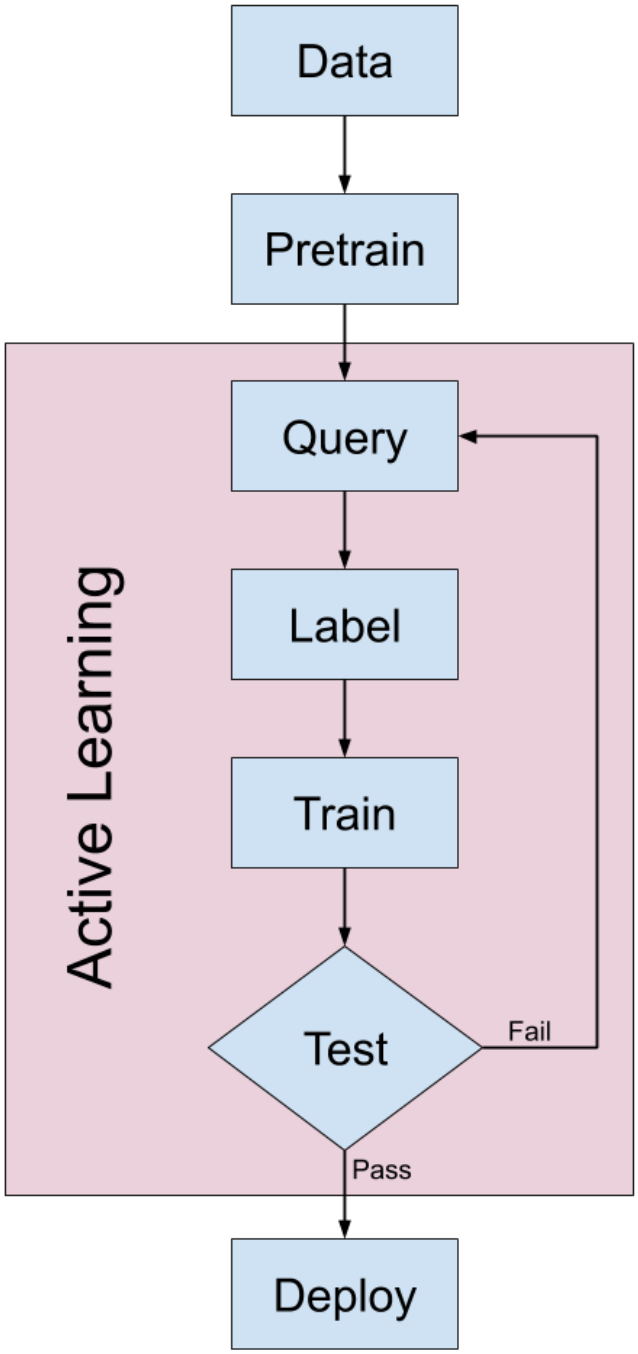
Query: Sample triplets of documents (A, B, C) such that answers to the question, "Is A more similar to B than C?", will maximally reduce model uncertainty according to some measure.

Label: Supply the triplets of documents to the researchers for labeling and receive the results.

Train: Update the encoder in order to satisfy the new triplet constraints.

Test: Examine the current state of the learned representation through visualizations, clustering, word frequency analysis, etc, to determine whether or not the encodings represent the required structure.

Deploy: Using the final encoded representation, systematize the codebook, train a classifier, etc.



Open questions

1. What datasets should we examine?
2. What pre-training objective should we use?
3. What should the query criteria be?
4. How should label interaction with researchers be handled?
5. What training objective should we use?
    a. Should the pre-training objective and triplet objective be optimized simultaneously or not?
    b. If yes to the above, should this be done just for samples involved in triplets or also for untouched samples?
6. What journals/conferences should we target?
7. Are there any grant opportunities we could target?

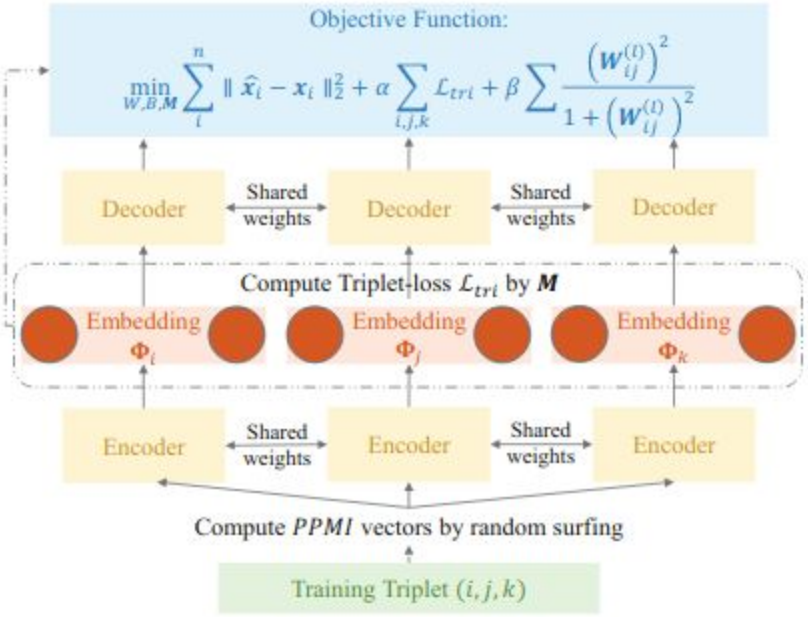
References

Triplets

1. Stochastic triplet embedding
    a. t-SNE paper
    b. Formulation: given dataset Z, and set of triples {(i,j,k), z_i is more similar to z_j than to z_k}, find a representation of Z that approximately satisfies the constraints implied by the triples.
    c. This formulation handles the situation where a distance metric between two examples is not well defined, but it is possible to perform comparisons of the triple form.
    d. Look at t-distribution-based triplet probability measure.
2. Learning a Distance Metric from RelativeComparisons
    a. Like 1. above but explicitly approximate the distance metric
    b. Use SVMs
3. Bayesian representation learning with oracle constraints
    a. Like 1. above but using a Bayesian framework.
    b. “Oracle” refers to a human providing soft supervision of the form A is more similar to B than to C.
4. Deep metric learning using Triplet network
    a. Similar in spirit to all of the above but:
        i. Used deep encoder representation.
        ii. Used classification metric to train triplet network (same class vs out of class)

iii. Note: surely someone has tried using the t-loss from 1 but with a triplet network?

5. Triplet Enhanced AutoEncoder:Model-free Discriminative Network Embedding
6. Train a triplet VAE with an added cost term for triplets in embedded space:



   a. They base triplet loss on pairwise similarity metric Mahalanobis distance but where the covariance matrix is replaced with a learned matrix and the differences are between two points. Not sure why this makes much sense. I suppose if the goal is to learn the distance metric.
7. In Defense of the Triplet Loss for Person Re-Identification
   a. Really nice paper with lots of wisdom about triplet loss functions.

Active Learning for relative comparisons

8. Active Learning from Relative Comparisons
9. Actively learning from triplet supervision
10. Referring to triplet constraints: “Such constraints, when available, have been shown to be useful toward learning tasks such as defining appropriate distance metrics or finding good clustering solutions.”
11. Misc reading notes:
    a. Notion of similarity in this work: instances of the same class are more similar than instances of a different class.
    b. Objective: information theoretic objective that maximizes information gain about class instances in the objective. *Question: not how well triples are modeled?* Later, they mention that their objective is suited to classification applications.
    c. Sampling trick: not much of a trick, but they uniformly sample 100n triples where n is the size of the dataset. They show that this works fine.

d. Useful formulation of query:

> A relative comparison query, denoted by triplet $r_{ijk} = (\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k)$, can be interpreted as a question: "*Is* $\mathbf{x}_i$ *more similar to* $\mathbf{x}_j$ *than* $\mathbf{x}_k$*?*". Given a query r_{ijk}, a oracle/user will return an answer, denoted by $l_{ijk} \in \mathcal{A} \triangleq \{\text{yes, no, dk}\}$, based on the classes to which the three instances belong. In particular, the oracle returns:
>
> - $l_{ijk} = \text{yes}$ if $y_i = y_j \neq y_k$,
> - $l_{ijk} = \text{no}$ if $y_i \neq y_j = y_k$, and
> - $l_{ijk} = \text{dk}$ (*do not know*) for all other cases;

e. They assume that the query answer is independent of the data points given their classes. This is the same as saying all of the information about similarity is contained in the class labels.

12. Semi-supervised Document Clustering via Active Learning with Pairwise Constraints
13. Improving Semi-Supervised Clustering Algorithms with Active Query Selection
14. Bayesian Active Distance Metric Learning