**REMOVE the FIRST Page – It is just for our reference.**

Code Submission Guideline:

1. Code 🡺 Ensure the code is clean and well commented/documented.
2. Related Data set
3. Documentation (README file + Other required docs)
4. Presentation Video 🡺 UIUC Mediaspace or YouTube and then supply URL in the Documentation. If the file is directly uploaded to GitHub, then all good. But limitation of the file size in GitHub is 100MB.

Can schedule demo or if demo video can explain everything, it’s fine to ignore that. Mention the challenges with Stemmer package installation. Dependency on the Visual Studio C++ package.

Documentation:

**1) An overview of the function of the code (i.e., what it does and what it can be used for).**

**2) Documentation of how the software is implemented with sufficient detail so that others can have a basic understanding of your code for future extension or any further improvement.**

**3) Documentation of the usage of the software including either documentation of usages of APIs or detailed instructions on how to install and run a software, whichever is applicable.**

**4) Brief description of contribution of each team member in case of a multi-person team.**

**Project Overview:**

In this project, we have tried to reproduce the model and results from the following published paper on Pattern Annotation.

**Qiaozhu Mei, Dong Xin, Hong Cheng, Jiawei Han, and ChengXiang Zhai. 2006. Generating semantic annotations for frequent patterns with context analysis. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD 2006). ACM, New York, NY, USA, 337-346. DOI=10.1145/1150402.1150441**

The goal is to annotate a frequent pattern with in-depth, concise, and structured information that can better indicate the hidden meanings of the pattern. This model will automatically generate such annotation for a frequent pattern by constructing its context model, selecting informative context indicators, and extracting representative transactions and semantically similar patterns.

This general approach has potentially many applications such as generating a dictionary like description for a pattern, finding synonym patterns, discovering semantic relations, ranking patterns, categorizing and clustering patterns with semantics, and summarizing semantic classes of a set of frequent patterns.

**Implementation Approach:**

Here, the general approach taken to automatically generate the closed frequent patterns and structured annotations for them by the following steps:

1) Derive the closed frequent patterns from the Database. PrefixSpan algorithm has been used in this step to find the closed frequent itemsets for the co-authorship.

2) Define and model the context of a pattern; select context units and design a strength weight for each unit to model the contexts of frequent patterns. FP-Growth Algorithm has been used in this step.

3) Apply weights to context indicators based on their strength to indicate pattern semantics. Design similarity measures for the contexts of two patterns, and for a transaction and a pattern context. **Mutual Information** /Cosine Similarity approach has been used to measure the strength/weight of the contexts.

4) Rank transactions and semantically similar patterns based on context similarity analysis. For a given frequent pattern, extract the most significant context indicators, representative transactions and semantically similar patterns to construct a structured annotation.

The title words are stemmed by Krovertz stemmer, which converts the morphological variations of each English word to its root form. Also, excluded all the stopwords i.e., very common words appears in the vocabulary to derive only the meaningful words and terms.

The model has been experimented on DBLP Dataset to show how the algorithm is effective for generating semantic pattern annotations and can be applied to various real-world tasks.

**<<Include the algorithms used for each function, related supports, threshold etc.>>**

**Installation and Usage:**

pip install -i https://test.pypi.org/simple/ krovetz

pip install -U prefixspan

pip install re, csv, pandas, numpy, mlxtend, sklearn

Here the input dataset (DBLP Dataset) is in a specific format. Each transaction/record has 3 fields – id (numeric), title (String) and MergedAuthors (string). The authors and co-authors associated with the title has been merged into a single column for easy analysis.

Once the program **patternAnnotation.py** is executed, it takes the **DBLP\_Dataset.csv** as the input and generates the output.txt file in the same path where source code exists. This output file contains all the closed frequent patterns and their most representative Context Indicators, most representative transactions (capped as 4), Semantically Similar Patterns (SSPs) as per co-author patterns and title term patterns. Each record in the output file represents one closed frequent pattern and their associated details.

**Experiment and Result:**

We propose the novel problem of semantic pattern annotation (SPA) generating semantic annotations for frequent patterns. A semantic annotation consists of a set of strongest context indicators, a set of representative transactions, and a set of semantically similar patterns (SSPs) to a given frequent pattern. We define a general vector-space context for a frequent pattern. We propose algorithms to exploit context modeling and semantic analysis to generate semantic annotations automatically. The context modeling and semantic analysis method we presented is quite general and can deal with any types of frequent patterns with context information. The method can be coupled with any frequent pattern mining techniques as a postprocessing step to facilitate interpretation of the discovered patterns.

We evaluated our approach on three different dataset and tasks. The results show that our methods can generate semantic pattern annotations effectively. As shown in our experiments, our method can be potentially applied to many interesting real-world tasks through selecting different context units and focusing on candidate patterns for SSPs.

We consider two types of patterns: (1) frequent co-authorship, each of which is a frequent itemset

of authors and (2) frequent title terms, each of which is a frequent sequential pattern of the title words. The goal of experiments on this dataset is to show the effectiveness of the SPA to generate a dictionary-like annotation for frequent patterns.

**<Attach Screenshot of the pivoted result file >**

Our experiments are designed as follows:

1) Given a set of authors/co-authors, annotate each of them with their strongest context indicators, the most representative titles from their publications, and the co-authors or title patterns which are most semantically similar to them.

2) Given a set of title terms (sequential patterns), annotate each of them with their strongest context indicators, the most representative titles, the most similar terms, and the most representative author/co-authors.

In both experiments, we use the tools FP-Growth and PrefixSpan to generate closed frequent itemsets of co-authors and closed sequential patterns of title terms respectively. The title words are stemmed by Krovertz stemmer, which converts the morphological variations of each English word to its root form. We set the minimum support for frequent itemset **as 12 and** sequential patterns as which outputs **<<64>> closed sequential patterns.**

**<<Explain the result received:>>**

In the supplied screenshot, we selectively show the results of semantic pattern annotations. We see that the SPA system can automatically generate dictionary-like annotations for different kinds of frequent patterns. For frequent itemsets like co-authorship or single authors, the strongest context indicators are usually their other co-authors and discriminative title terms that appear in their work. The semantically similar patterns extracted also reflect the authors and terms related to their work. However, these SSPs may not even co-occur with the given pattern in a paper.

**Disclaimer:**

While reproducing the model and the result presented in the paper, few of the steps have not been followed the same way due to time constraints.

a) To calculate the strength weighting of the context units, instead of Mutual information approach, we have implemented a new weighting method.

b) Redundancy removal in closed frequent pattern by using Micro-Clustering technique has not been applied.

c) While the published paper has experimented the model and algorithm on 3 different datasets, we have done that on only the first one i.e., DBLP dataset (a subset of around 12k transactions/titles papers from the proceedings of 12 major conferences in Data Mining; around 1k latest transactions from each of such conference).

**References:**

**Qiaozhu Mei, Dong Xin, Hong Cheng, Jiawei Han, and ChengXiang Zhai. 2006. Generating semantic annotations for frequent patterns with context analysis. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD 2006). ACM, New York, NY, USA, 337-346. DOI=10.1145/1150402.1150441**

**DBLP Dataset:** 12 Major Conferences on Data Mining. 1000 latest titles from each conference.

**ACL** - Annual Meeting of the Association for Computational Linguistics (<https://dblp.uni-trier.de/db/conf/acl/>)

**ADBIS** - Symposium on Advances in Databases and Information Systems (<https://dblp.uni-trier.de/db/conf/adbis/>)

**CIKM** - International Conference on Information and Knowledge Management (<https://dblp.uni-trier.de/db/conf/cikm/>)

**ECIR** - European Conference on Information Retrieval (<https://dblp.uni-trier.de/db/conf/ecir/>)

**ICDE** - IEEE International Conference on Data Engineering (<https://dblp.uni-trier.de/db/conf/icde/>)

**ICDM** - IEEE International Conference on Data Mining (<https://dblp.uni-trier.de/db/conf/icdm/>)

**KDD** - Knowledge Discovery and Data Mining (<https://dblp.uni-trier.de/db/conf/kdd/>)

**PAKDD** - Pacific-Asia Conference on Knowledge Discovery and Data Mining (<https://dblp.uni-trier.de/db/conf/pakdd/>)

**SDM** - SIAM International Conference on Data Mining (<https://dblp.uni-trier.de/db/conf/sdm/>)

**SIGIR** - Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (<https://dblp.uni-trier.de/db/conf/sigir/>)

**WSDM** - Web Search and Data Mining (<https://dblp.uni-trier.de/db/conf/wsdm/>)

**WWW** - The Web Conference (<https://dblp.uni-trier.de/db/conf/www/>)

**<<Need to provide the references for all the packages used?? >**

**Work Distribution:**

**<<Who did what>>**