Nikola Mrksic, Trinity College

Semi-supervised learning methods for data augmentation

Supervisor: Dr Sean Holden

Overseers: Dr Simone Teufel, Prof Jon Crowcroft

Semi-supervised learning

- **Classic Supervised Learning**:

  Training data - labelled examples of ***all*** n classes.

- **Positive-unlabeled(PU) learning**:

  Training data contains examples of a positive class ***P*** and a set of ***unlabeled*** examples ***U,*** of which some belong in ***P*** and the rest belong to negative classes.

- Key feature of ***PU learning***: no labelled negative training data is available!

PU Learning and Data Augmentation

- Many **data-scarce** application areas have **small sets of positive examples** and a **huge body of mixed examples** available (it might be hard/costly/time consuming to manually find more positives).

- The goal of the project is to apply **different** *PU learning* algorithms to such application areas and examine how they can improve (*by augmenting their training data set*) the performance of subsequent supervised learning approaches used.

PU learning methods implemented

- **Bayesian Sets** [Ghahramani and Heller , 2006]
- Inspired by *Google Sets'* approach to set expansion.
- Treat entity set expansion as a Bayesian inference problem:
- Result: A ranking of examples in ***U*** by their likelihood of belonging to ***P****.*
- *A*ssumes that all features are independent in order to obtain a *tractable* solution.
- Given binary features, reduces to a single matrix multiplication!

Entity Set Expansion via Bayesian Sets

- MovieLens data set:
- Set to be expanded: {Empire Strikes Back, Return of the Jedi Indiana Jones and the Last Crusade}
- Ranking produced:

181	9.58508	181|Return of the Jedi (1983)
50	9.58508	50|Star Wars (1977)
172	9.34084	172|Empire Strikes Back, The (1980)
271	7.99097	271|Starship Troopers (1997)
498	7.14069	498|African Queen, The (1951)
897	5.20461	897|Time Tracers (1995)
450	5.20461	450|Star Trek V: The Final Frontier (1989)
449	5.20461	449|Star Trek: The Motion Picture (1979)
380	5.20461	380|Star Trek: Generations (1994)
373	5.20461	373|Judge Dredd (1995)
230	5.20461	230|Star Trek IV: The Voyage Home (1986)
229	5.20461	229|Star Trek III: The Search for Spock (1984)
228	5.20461	228|Star Trek: The Wrath of Khan (1982)
227	5.20461	227|Star Trek VI: The Undiscovered Country (1991)
222	5.20461	222|Star Trek: First Contact (1996)
82	5.20461	82|Jurassic Park (1993)
62	5.20461	62|Stargate (1994)
121	5.01092	121|Independence Day (ID4)

PU learning methods implemented

The other two methods are based on a *two-step* strategy:

1. Identifying Reliable Negatives(**RN**);
2. Building a sequence of classifiers and selecting the optimal one.

- **Spy-EM** algorithm [Liu et al, 2002]

  Uses **Naive Bayesian**(**NB**) classifiers to identify RNs and the **Expectation Maximization**(**EM**) algorithm to build the classifiers.

- **Roc-SVM** algorithm [Li et al, 2003]

  Uses the **Rocchio classifier** and **k-means clustering** to identify RNs. **Support Vector Machines(SVM)** are used to build the classifiers.

Results achieved

- **Data augmentation on *text classification data* (Reuters dataset)**:

- SVM classification performance: [precision, recall, f-score]:

- Prior to augmentation: **0.9590 0.7919 0.8675**
- Post augmentation:

Bayesian sets:	0.6601	0.8613	0.7474
Spy-EM:	0.8717	0.9645	**0.9158**
Roc-SVM:	0.9580	0.8468	**0.8990**

- In traditional problems such as this one, Spy-EM and RocSVM are vastly superior to Bayesian Sets(finding appropriate **cutoff** is hard).

Results achieved

- **Data augmentation on *biomedical data* (KDDCup 2001, Task 1)**:

- Very hard dataset: 1:20 positive to negative examples ratio.
- Computationally heavy: 140000 features, 2000 entries!

- SVM classification performance: [precision, recall, f-score]:

- Prior to augmentation: **0.1579 0.0200 0.0355**
  Post augmentation:
  Bayesian sets: **0.1667 0.1133 0.1349**
  Spy-EM: 0.1183 0.0733 0.0905

- In our experiments with this dataset, Bayesian Sets **consistently** outperform the other two methods (and execute **much** faster!)

Further work

- Data augmentation on theorem prover statements.
- Utilising the disclosed information about negative examples in the training data in order to boost data augmentation further.
- **Iterative (bootstrapped) Bayesian Sets**: using newly extracted positives to repeatedly refine the rankings produced.
- A consideration of the proof of stability of the Bayesian sets algorithm, warranting its applicability even when its assumption of independent features is violated.