CS 5604 Spring 2015

Classification

Xuewen Cui
Rongrong Tao
Ruide Zhang

May 5th, 2015

Overview

- Background
- Algorithm
- Implementation
- Result and Evaluation
- Future Work
- Conclusion

Background

- Feature Extraction
  - Words extracted as features
- Feature Selection
  - Feature vectors generated from Apache Mahout
- Classification – enforce Solr search engine
  - Naive Bayes

Algorithm

- Text Mining
  - Naïve Bayes
  - Random Forest
  - Support Vector Machine (SVM)

    Only serial implementation provided by Mahout

Algorithm - Naive Bayes

Task: Classify a new instance D based on a tuple of attribute values $D = \langle x_1, x_2, \ldots, x_n \rangle$ into one of the classes $c_j \in C$

$$c_{MAP} = \underset{c_j \in C}{\operatorname{argmax}} P(c_j \mid x_1, x_2, \ldots, x_n)$$

$$= \underset{c_j \in C}{\operatorname{argmax}} \frac{P(x_1, x_2, \ldots, x_n \mid c_j) P(c_j)}{P(x_1, x_2, \ldots, x_n)}$$

$$= \underset{c_j \in C}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_n \mid c_j) P(c_j)$$

MAP = Maximum Aposteriori Probability

Implementation

- Data labeling
- Data source: cleaned data from reducing noise team.
- Collection: small collection, large collection.
- Each collection has tweets and webpages.
- Each has 8 topics(come from 8 teams).
- Each topics: 100 positive,100 negative
- Train set: 80% samples
- Measurement of Accuracy: 5-fold Cross-Validation

Implementation

- Apache Mahout Naive Bayes

```
mahout seqdirectory
  -i ${WORK_DIR}/20news-all
  -o ${WORK_DIR}/20news-seq
  -ow
```

```
mahout seq2sparse
  -i ${PATH_TO_SEQUENCE_FILES}
  -o ${PATH_TO_TFIDF_VECTORS}
  -nv
  -n 2
  -wt tfidf
```

```
mahout trainnb
  -i ${PATH_TO_TFIDF_VECTORS}
  -o ${PATH_TO_MODEL}/model
  -li ${PATH_TO_MODEL}/labelindex
  -ow
  -c
```

```
mahout testnb
  -i ${PATH_TO_TFIDF_TEST_VECTORS}
  -m ${PATH_TO_MODEL}/model
  -l ${PATH_TO_MODEL}/labelindex
  -ow
  -o ${PATH_TO_OUTPUT}
  -c
  -seq
```

Class label prediction for new data is not supported

Implementation

❖Apache Mahout Naive Bayes

```
mahout seqdirectory
  -i ${WORK_DIR}/20news-all
  -o ${WORK_DIR}/20news-seq
  -ow
```

```
mahout seq2sparse
  -i ${PATH_TO_SEQUENCE_FILES}
  -o ${PATH_TO_TFIDF_VECTORS}
  -nv
  -n 2
  -wt tfidf
```

```
mahout trainnb
  -i ${PATH_TO_TFIDF_VECTORS}
  -o ${PATH_TO_MODEL}/model
  -li ${PATH_TO_MODEL}/labelindex
  -ow
  -c
```

```
mahout testnb
  -i ${PATH_TO_TFIDF_TEST_VECTORS}
  -m ${PATH_TO_MODEL}/model
  -l ${PATH_TO_MODEL}/labelindex
  -ow
  -o ${PATH_TO_OUTPUT}
  -c
  -seq
```

Learning Apache Mahout Classification

work for Hadoop 1.x

Implementation

- pangool: 100% compatible with different versions of Hadoop
  - Naive Bayes classification with MapReduce

- Jose Cadena from Hadoop Team
  - made our program able to read in .avro file and write to .avro file

Example

Training: .txt file

POSITIVE	content of tweets or webpages
NEGATIVE	content of tweets or webpages

New data file stored in HDFS

All results (labeled new data) loaded into HBase by Hadoop team

Result and Evaluation

❖Tweets: Small Collection

Team	Collection Topic	Average Accuracy	Filesize	Runtime
Classification	Plane Crash	92%	90M	13s
LDA	Suicide Bomb Attack	89%	13M	8.6s
Hadoop	Jan. 25	73%	214M	18.6s
Solr	Election	86%	298M	34s
Reducing Noise	Charlie Hebdo	85%	64M	8.4s
NER	Storm	100%		

Arabic

negative tweets come from other collections

Result and Evaluation

❖ Tweets: Large Collection

Team	Collection Topic	Average Accuracy	Filesize	Runtime
Classification	Malaysia Airlines	78%	270M	32s
Hadoop	Egypt	81%	3.1G	136s
Reducing Noise	Shooting	73%	7.4G	112s
LDA	Bomb	75%	5.8G	110s

Result and Evaluation

- Webpages: Small Collection

Team	Collection Topic	Average Accuracy	Filesize	Runtime
NER	storm	83%	65M	4.7s
Classification	Plane Crash	87%	24M	9s

Result and Evaluation

❖Webpages: Large Collection

Team	Collection Topic	Average Accuracy	Filesize	Runtime
Hadoop	Egypt	79%	140M	36s
Clustering	Diabetes	77%	305M	40s

Future Work

- For performance improvement
  - Use larger training set
  - Try using tf-idf value instead of word count in NaiveBayesGenerate.java
  - Use more representative features

Conclusion

- Learned feature selection and classification algorithm on Apache Mahout

- Found a package to predict class label using Naive Bayes classifier generated from Apache Mahout
  - Make it compatible with higher versions of Hadoop, instead of just 1.1.1

- Found a package for M/R Naive Bayes classifier
  - Can be updated later for performance improvement

Thanks

- We give our thanks to
  - Instructor: Dr. Fox
  - TA: Mohamed Magdy and Sunshin Lee
  - Classmates
    - Jose Cadena from Hadoop Team
    - Reducing Noise Team

Acknowledgements

We would like to thank our instructor Dr. Edward A. Fox, who brought us into this interesting project. We would like to thank our TAs, Mohamed Magdy and Sunshin Lee, for their continued support and valuable suggestions throughout the project. We would also give special thanks to Jose Cadena from the Hadoop team, who helped us with input and output formatting problems. Further, we thank the Reducing Noise team, which provided cleaned tweets and webpages for us to work on. Finally, thanks go to the support of NSF grant IIS - 1319578, III: Small: Integrated Digital Event Archiving and Library (IDEAL).

Questions