UNIVERSITY OF TEXAS AT DALLAS

News Content Text Classification

Kunal Arora (kxa142230)
Varun Kumar Reddy (vxb150830)
Mohammed Saad Tambe (mxt142830)
12/7/2015

Contents

Problem Description

Classifying the semantic content, or topic, of text is one of the critical problems in natural language processing, information retrieval, artificial intelligence and machine learning more broadly. Newspaper articles provide a particularly good opportunity for learning such classifications, as the semantic content of articles is generally coherent, and large, open source corpuses of labeled news articles exist and are easily accessible.

Here, in our project we are classifying the news articles into one of the three categories:

- Politics
- Business
- Sports

Proposed Solution

We are using Naïve Bayes model along with the following features to classify the news articles

- Lexical Features (**Stop Word Elimination**, **Bi-Gram**, **Stemming**, **Lemmatization**)
- Syntactic Features (**Headwords**, **POS Tags**, **Syntactic Parsing**)
- Semantic Features (**Hypernyms**, **Hyponyms**, **Synonyms**, **Meronym**)

Implementation Details

Baseline Classifier: Random Classifier.

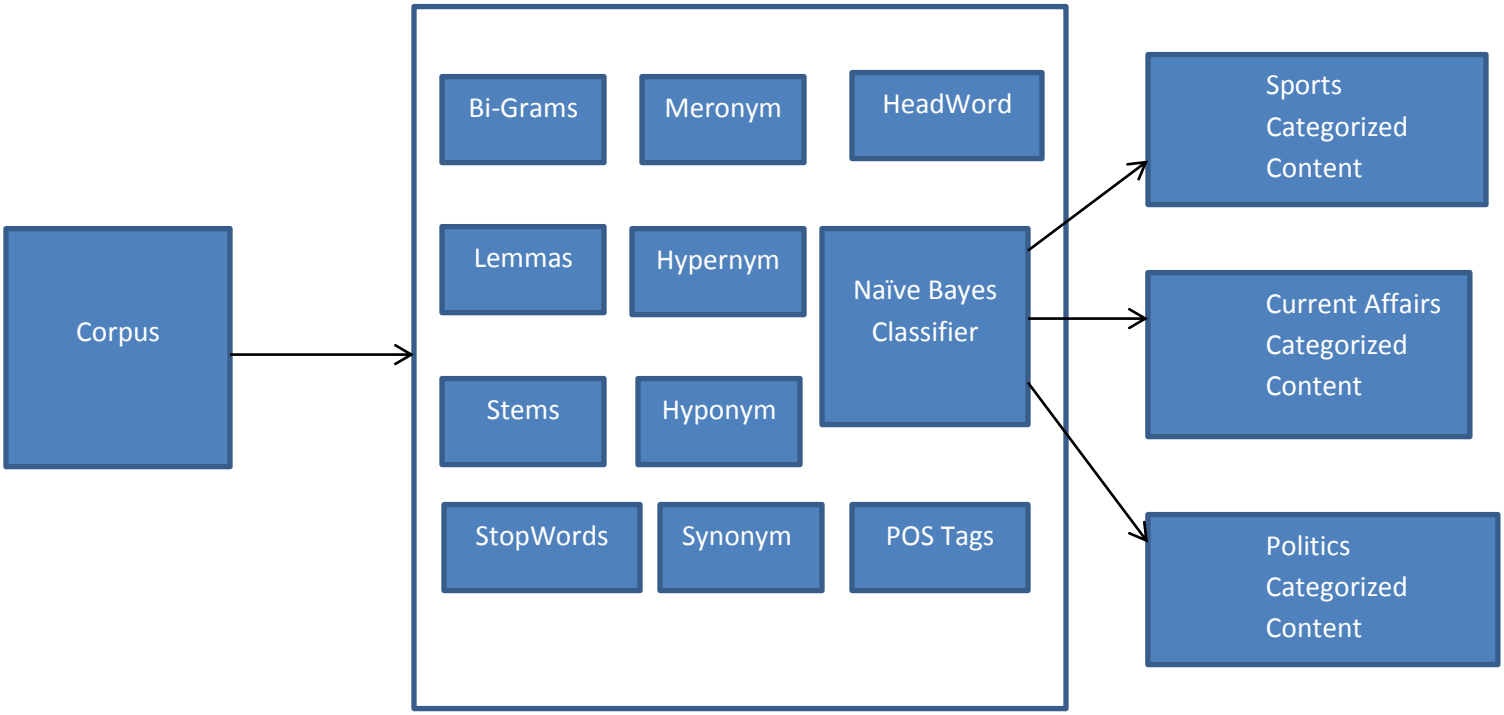
1. The baseline classifier used in the system is a random classifier.
2. Random Classifier is uniform distribution based classifier giving an accuracy of 33.33 % for each of the 3 categories.

Enhanced Classifier

Overview

The system uses Naïve Bayes along with different features to classify the documents. In order to classify documents using Naïve base, each document is represented as a vector of terms. As the number of words in the documents is high, feature dimensionality is an important problem. Therefore dimensionality reduction methods like stemming and stop word removal is applied. By using semantic features for categorization better results are obtained for e.g. Part of speech tags of terms and relations in WordNet such as synonyms, hypernyms, hyponyms, meronyms and topics of terms are used

Architectural Diagram



Details:

Below are details of features implemented

The cumulative score from all the features is used to determine which class the document belongs. For e.g. the score from the below features is added to get the aggregate score of a document for each class and then score is compared to find the result class.

Lexical Features:

1. **Bigram**
   A bigram is every sequence of two adjacent elements in a string of tokens, which are typically words.
   E.g.: American government increased its security at all of its sea banks.
   Chase, an American private bank releases its annual business details.

2. **Stemming and Lemmatization**
   Stemming and Lemmatization is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form. Here we used the stem, base or root of the word as feature instead of just word itself as a feature. The drawback of just using the word itself as feature is it treats a word in two different tenses as two different words.
   E.g.: Obama announced the war on ISIS.
   Obama have thoughts of announcing the war on ISIS.

Syntactic Features

1. **Headword**

   Headword is a word which tells us main theme of a sentence. In other we can say it as headword is the most important word in the sentence in most of the cases. This features incorporates bag of headwords model in the system to compute the category score.

   Workers **dumped** sacks into a bin.

2. **POS Tags**

   A word tagged with its parts of speech will be more unique in nature and also it solves the problem of ambiguity of word senses up to some extent.

   The/DT ***back/JJ*** door/NN

   On/IN my/PRP ***back/NN***

   Win/VB the/DT voters/NN ***back/BB***

Semantic Features

1. **Hypernym**

   The most frequently encoded relation among synsets is the super-subordinate relation (also called hypernymo, hyponymy or is a relation). It links more general synsets like furniture, piece of furniture to increasingly specific ones like bed and bunkbed. All noun hierarchies ultimately go up the root node entity.

2. **Hyponym**

   Hyponymy relation represents transitive relationships for e.g. if an armchair is a kind of chair, and if a chair is a kind of furniture, then an armchair is a kind of furniture.

   WordNet distinguishes among Types (common nouns) and Instances (specific persons, countries and geographic entities). Thus, armchair is a type of chair; Barack Obama is an instance of a president. Instances are always leaf (terminal) nodes in their hierarchies.

3. **Meronym**

   Meronymy, the part-whole relation holds between synsets like chair and back, backrest, seat and leg. Parts are inherited from their super ordinates: if a chair has legs, then an armchair has legs as well. Parts are not inherited upward as they may be characteristic only of specific kinds of things rather than the class as a whole: chairs and kinds of chairs have legs, but not all kinds of furniture have legs.

Results

Base Classifier (Random Classifier)

Sr. No.	Category	Accuracy
1	Politics	33.33 %
2	Business	33.33 %
3	Sports	33.33 %

Enhanced Classifier

Sr. No.	Category	Accuracy
1	Politics	80.10 %
2	Business	99.8 %
3	Sports	80.20 %

Problems Faced

1. While doing Bi-Gram feature, we were dealing with all possible bigrams, if the corpus had *N* unique words, we were dealing with n*n bigrams. It was taking lot of time to train the model i.e. about 10-15 minutes for 1 category. We changed to n-1 feasible bigrams at the time of learning and during testing, if a different bigram occurred, we are adding to the existing bi-gram set. It takes less time than the previous one.

2. While extracting the Headwords using Stanford Parser, it was taking so much time for computing and also the head word we were getting was wrong when the sentences of is of length greater than 150 characters(i.e. 25 words). How It was Resolved: We kept condition that we will get headwords only sentences of length less than 150. Then it was giving results very fast.

Pending Issues

We are not dealing with multi-tag news classification. Example, if an article belongs to Sports as well as Politics. This would affect the accuracy of the either category.

Improvements

1. As mentioned above, one of the improvements would be to make this project a multi-tag news classification.
2. Currently, we are only dealing with three categories. We can add more categories to make the project more useful.
3. We are using only body/text of the article to classify. We can improve the system by making use of article metadata associated such as title, short description and summary etc.

Programming Tools

1. Rita WordNet
2. Stanford POS Tagger
3. model: english-left3words-distsim.tagger
4. Stanford Parser
5. model: Lexical Parser with Probabilistic Context Free Grammar
6. Jaws library
7. JAVA 8