Feature Engineering

Yawen Zhang

Using late days: 5 days
Kaggle user name: yawenz

How I used the data?

I used **CrossValidation** for the training data and also submit my prediction result to Kaggle to get the **Prediction Accuracy**. Also, after each iteration, I would inspect the error based on CV to figure out how to improve the performance. I used **three types of feature** in total and incorporating them one by one, to check the performance improvement.
For CrossValidation on training data, **I set 10 passes and 30% of them as validation dataset**.

Feature extraction results

Feature 1: **Bag of words**, change CountVectorizer to **TfidfVectorizer** for normalization

Function: Reveal which words are positive for spoiler prediction and which words are negative.
Example: in "spoiler" sentences, frequent words like "die", "win" and "finale"; in not "spoiler" sentences, frequent words like "school", "party".
Results: CrossValidation_Accuracy = 0.673

Feature 2: The appearance of **some "tropes"**

Function: Figure out which tropes is positive for spoiler detection and which are negative.
Example: For trope like "AnyoneCanDie", all the sentences are labeled as spoiler. For trope like "NewYorkCity", all the sentences are labeled as not spoiler.
Results: CrossValidation_Accuracy = 0.719

Feature 3: The appearance of **"page" word in "sentence"**

Function: Figure out if some page words appearing in the corresponding sentences.
Example: For page like "Nikita", all the sentences are labeled as spoiler because they all have word "Nikita", indicating they are spoiling some information about the main character.
Results: CrossValidation_Accuracy = 0.526

My New Feature: Union of features: **1 + 2 + 3**

Function: My new feature is the union of feature 1, 2 and 3. I combine the three features using "FeatureUnion" and "Pipeline". "FeatureUnion" enables me to add more features to my model and "Pipeline" enables me to process and transform multi-source data in a sequence, for example, I used three fields in the training data: **sentence, page and trope**.
Example: By adding the new model, the sentences with positive words for spoiler can be detection, also, for some trope like "AnyoneCanDie", all of them are labeled as spoiler. Finally, if the page words like "Nikita" appears in a sentence, would be labeled as spoiler.
Results: CrossValidation_Accuracy = 0.742, **Prediction_Accuracy = 0.705** (baseline: 0.641)
Analysis: Training and testing error is similar (difference less than 0.04), no overfitting