Identifying Salient Named Entity Of a Tweet

Under the guidance of

Vasudev Varma

Mentor:

Priya Radhakrishnan

Team #8:

Shwetha G

Sashank Nistala

Vibhav Srivastav

Joshita Misra

Abstract

- Unlike carefully authored news text and other longer content, tweets pose a number of new challenges
- We propose a solution to the problem of determining what a tweet is about through semantic linking: we add semantics to tweets by automatically identifying concepts that are semantically related to it.
- Empirical analysis of named entity recognition and disambiguation.
- The identified concepts can subsequently be used for, e.g., social media mining, thereby reducing the need for manual inspection and selection.

Phase 1

Identification of Named Entities

- Created around 2000+ tweet datasets.
- Identifying **Named Entity** representing the tweet.
- Evaluation of system for various approaches taken.

Approach

- Collection of tweet dataset raw: 2000+ tweets classified.

- Extraction of Named entities using **Stanford** Ner tool, language used: Python

- Extraction of Named entities using GATE tool ,language used: Java

Approach (Contd..)

- Extraction of Named entities using **custom built ner tool** from POS tagged tweets

  language used: Java

- Noise is removed from original tweets by removing non ASCII characters and some special characters.

Approach (Contd..)

Tagged tweets are processed to identify patterns of Named Entities

_USR (user eg: @username)

_HT (hash tag eg: #felicity)

_NNP (eg: Boehner)

_NNP+ (repeated occurrence of NNP _NNP+ _IN _NNP+ (two sets of NNP with 'of' or 'for' eg: Bank of Thailand)

_NN or _NNS (single occurrence of singular or plural nouns)

Evaluation Tool

- Build evaluation tool to evaluate the results generated from different approaches.
- Evaluation tool calculates the ***Precision*** and ***Recall*** and ***F-Score*** for different approaches.

Observations

- Observations:

Following table shows the precision and recall values obtained for approaches used.

- **Precision** = (correct + 0.5 * partially_correct) / (correct + incorrect + partial)
- **Recall** = (correct + 0.5 * partially_correct) / (correct + missing + partial)

Results :

- Precision and Recall for **Stanford_ner** :
- 42.94 and 9.99 respectively
- Precision and Recall for **Gate_ner**
- 33.55 and 37.98 respectively
- Precision and Recall for **Custom_ner**
- 43.42 and 81.53 respectively

Phase 2

Identifying Salient Named Entities

- Ran Custom_ner on 2000+ tweets.
- Manually identified the SNE of tweets.
- Created NE ranking system to get SNE programmatically.
- Built Evaluation System to compute Precision, Recall and F-Score.

- Compare results and optimize.
- Build a UI Tool to accept tweets from user.
- Identify NE, SNE and display top 3 SNE

Heuristics for SNE

- Get the Titles of the Wiki link containing the named Entities identified.
- Gave weightage to partial and full title match
- Added weightage for n-grams and proper nouns

Evaluation of approach

- Manually classified 2000+ tweets
- Ran the program with the heuristics considered
- Compared the results

Observations:

Precision: 64.83%

Recall : 63.24%

F-Score : 64.031

Optimization approaches

Different approaches were taken to get the results faster

1. Store the scores of NE and re use it when word re-appears.

2. Remove stop words from NE list to enhance accuracy of results

3. Remove duplicate NE from SNE list