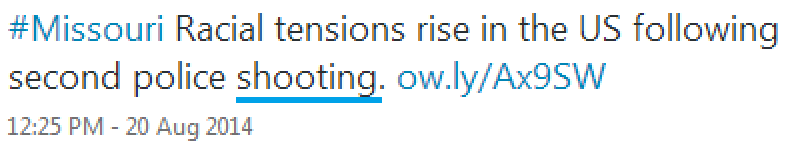
Centre for
Data Analytics

A Keyword Sense Disambiguation Based Approach for Noise Filtering in Twitter

Sanjaya Wijeratne and Bahareh R. Heravi

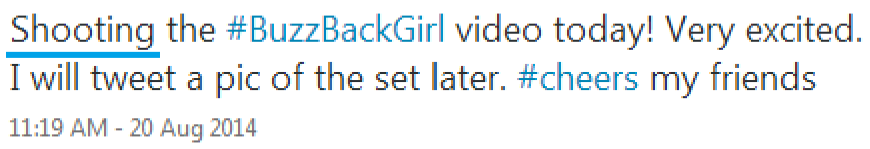
Motivation

With growing popularity of streaming social media platforms such as Twitter for news reporting, locating timely and newsworthy information from them has become an essential step in Digital Journalism. Journalists use keywords-based tweet filtering to locate tweets created by eyewitnesses in order to create news stories. Keywords-based tweet filtering also brings a lot of irrelevant tweets as well. The motivation of this work is to help journalists to find newsworthy content that interest them from Twitter by filtering out noisy tweets collected by keywords-based tweet filtering.



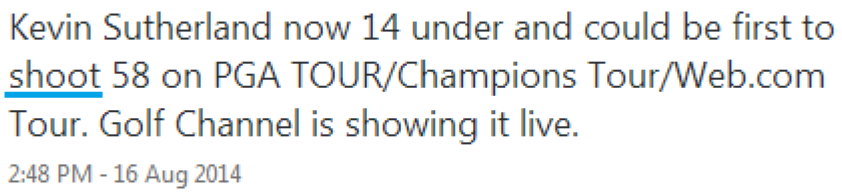
#Missouri Racial tensions rise in the US following second police shooting. ow.ly/Ax9SW

12:25 PM - 20 Aug 2014



Shooting the #BuzzBackGirl video today! Very excited. I will tweet a pic of the set later. #cheers my friends

11:19 AM - 20 Aug 2014



Kevin Sutherland now 14 under and could be first to shoot 58 on PGA TOUR/Champions Tour/Web.com Tour. Golf Channel is showing it live.

2:48 PM - 16 Aug 2014

Problem Statement

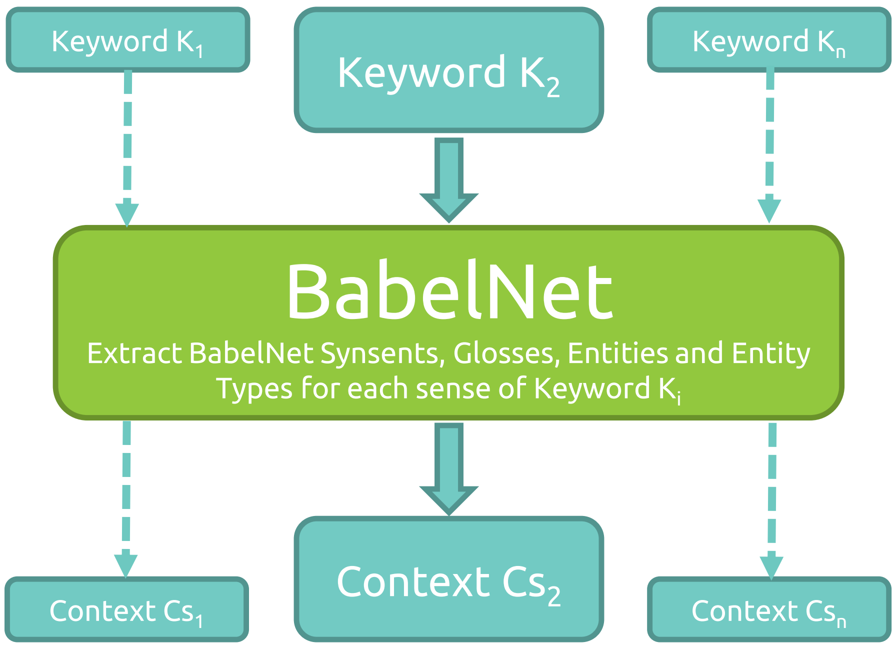
Let K be the set of all keywords used to collect T, which is the set of all tweets collected. Let S be the set of all senses for all keywords in K. Let $S^+ \in S$ be the set of all senses that could collect interesting tweets to the user for all $k_i \in K$. Let P, a subset of K be the set of all tweet collecting keywords present in tweet $t \in T$. Given a tweet t $\in$ T, K and S^+, can we determine whether t is an interesting tweet to the user?

Proposed Solution

For each tweet collecting keyword $k_i \in K$, we extract it's senses from BabelNet to generate S. S_i is the set of all senses of $k_i \in K$. For each $k_i \in K$, the user will select a set of senses S_i^+ which is a subset of S_i, that made the user to pick k_i as a keyword, which helps us to understand what senses of k_i would bring interesting tweets to the user. All senses of a keyword k_i that are not selected by the user S_i^- which is a subset of S_i are considered as senses that could bring noise for k_i. For each sense $s_i \in S_i$ of k_i, we generate a list of associated words (stopword removed and stemmed) using BabelNet synsets, glosses, entities and their types, which act as the context Cs_i for each sense $s_i \in S_i$ of k_i. Given a tweet t, we identify entities, their types, and remove stopwords and stem the remaining words to generate the context of the tweet C_t. For each keyword $p_i \in P$ in t, we disambiguate and assign the best sense to p_i using Simplified LESK algorithm by calculating the overlap of each keyword's sense's context Cs_i with C_t. If the sense assigned to keyword p_i is from S_i^-, t will be classified as a noisy tweet for p_i and will be filtered.

This project has been funded by SFI grant RSF 12/72.

Generating Context for Keywords' Senses



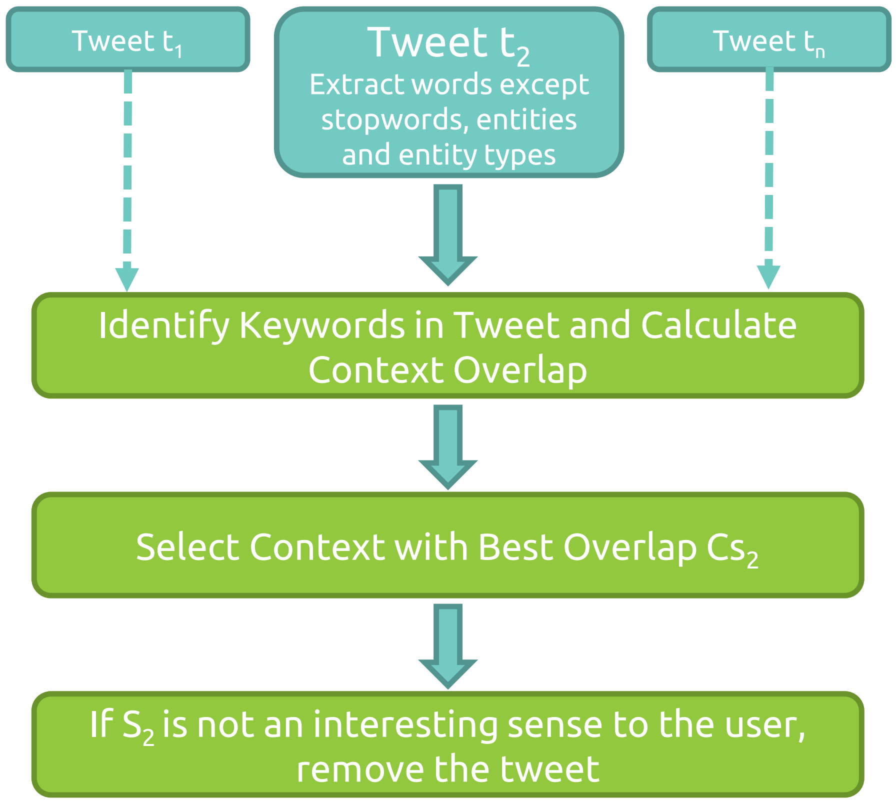
Keyword K_1 | Keyword K_2 | Keyword K_n

BabelNet

Extract BabelNet Synsents, Glosses, Entities and Entity Types for each sense of Keyword K_i

Context Cs_1 | Context Cs_2 | Context Cs_n

Identifying and Removing Noisy Tweets



Tweet t_1 | Tweet t_2 | Tweet t_n

Extract words except stopwords, entities and entity types

Identify Keywords in Tweet and Calculate Context Overlap

Select Context with Best Overlap Cs_2

If S_2 is not an interesting sense to the user, remove the tweet

Discussion on Evaluation

We plan to evaluate our approach using randomly selected tweet samples on each keyword that we used to collect tweets. We will manually remove any duplicate tweets in them. Accuracy will be measured on how precisely our approach identifies noisy tweets. In our initial evaluation for keyword "shoot" with a sample set of 100 tweets (66 noisy), we achieved 89% accuracy in removing noisy tweets.