**Topic: Content Driven Frame Work for Geo location**

Note : This paper is not included in the list of conferences provided by sultan.However we can use some of its conetent that might help.

***Highlights of the paper:***

1. The paper overcomes the sparsity of geo enabled location based sensing system by

* its reliance purely on publicly available content;
* a classification component for automatically identifying words in posts with a strong local geo-scope
* a lattice-based neighbourhood smoothing model for refining a user’s location estimate

1. To overcome this location sparsity problem, the paper propose that a reasonable framework to predict a microblog user’s location should contain the following features:

* the proposed framework should be generalizable across social media sites and future human-powered sensing systems;
* the framework should be robust in the presence of noise and the sparsity of spatial cues in a microblog user’s posts;
* the framework should provide accurate and reliable location estimation; and
* the prediction framework should be based purely on the publicly available data from the user, with no need for proprietary data from system operators (e.g., backend database) or privacy-sensitive data from users (e.g., IP or user/pass).

1. Content Analysis:

* In terms with gazetteer - based on geo-related terms in a specialized external knowledge base extracted addresses, postal code, and other information listed in a geographical gazetteer from Web content to identify the associated geographical scope of Web pages and blogs.
* With probabilistic language models – based on tags that photos are labeled by flicker. Based on these models and Bayesian inference they show estimate how to estimate the location of photo .
* Inference via social relations(Privacy Inference) – how private information may inferred through analysis of user’s social relations ? Assuming that users have common attributes. Given location trails from two users, both propose metrics to measure similarity between two users given their location history, and predict friendship according to the similarities

***Location sparsity on Twitter :***

This papers uses two crawling strategies for Twitter:

1. Crawling public timeline API
2. Crawling by breadth – first search through social edges to crawl user’s friends and followers.

***Problem statement***

Given the lack of granular location information for Twitter users, our goal is to estimate the location of a user based purely on the content of her tweets. **Having are reasonable estimate of a user’s location** can enable content personalization (e.g targeting advertisements based on the user’s geographical scope, pushing relatednews stories, etc.), **targeted public health Web mining** (e.g., a Google Flu Trends-like system that analyzes tweets for regional health monitoring), and **local emergency detection** (e.g., detecting emergencies by monitoring tweets about earthquakes, fires, etc.). By focusing on the content of a user’s Twitter stream, such an approach can avoid the need for private user information, IP address, or other sensitive data. With these goals in mind, we focus on city-level location

***Location Estimation Problem.***

Given a set of tweets Stweets(u) posted by a Twitter user u, estimate a user’s likelihood score of being located in city i: slikelihood(i|Stweets(u)), such that the city with maximum likelihood score lest(u) is the user’s actual location lact(u).

To evaluate the quality of a location estimator, we compare the estimated location of a user versus the actual city location (which we know based on the city corresponding to her latitude/longitude coordinates). The first metric we consider is the error distance which quantifies the distance in miles between the actual location of the user lact(u) and the estimated location lest(u). **The Error Distance** for user u is defined as

*ErrDist*(*u*) = *d*(*lact*(*u*), *lest*(*u*)).

To evaluate the overall performance of a content-based user location estimator, we further define the ***average error distance***across all test users *U*.