Tracking the Diffusion of Named Entities

TBD

Abstract

To do

Index Terms

To do

I. Introduction

The aim of this paper is to understand how named entities *emerge* and *spread* through social media based discourse. We are interested in exploring the following research questions:

1) **RQ1:** How can we accurately detect named entities in social media based discourse, given its myriad formats, often informal vernacular, and inherent noise (e.g. misspellings, abbreviations, etc.)?
2) **RQ2:** Under what conditions do entity mentions diffuse through discourse? And when are people *most likely* to be influenced into then discussing entities?
3) **RQ3:** How can we predict the discussion of certain named entities and who will begin talking about them?

II. Datasets

For this research we will use the following datasets:

1) Reddit data – download and access all of the data from the full dump.[1]
2) CoNLL 2003 data – a corpus of newswire texts, annotated for named entity chunks and types. This describes where entity mentions are in the text, including locations, organisations, and person mentions.
3) Twitter data; unannotated – we have a large corpus of English tweets that we can use here.
4) Twitter data; annotated – there are two datasets annotated with named entities. These are from Ritter's 2011 EMNLP paper, and the W-NUT 2015 shared task.

III. Research Stages

A. Stage 0: Data Preparation and NER

-To do:

–Annotate corpora with detected entities using basic typing of: person, location, organisation

–Run NER software over dataset and validate accuracy of this (using basic measures)

–Run NER over entire dataset to extract entities

[1] https://archive.org/details/2015_reddit_comments_corpus

B. Stage 1: Exploratory Analysis

-To do:

–Plot relative frequency distribution as a function of time for named entities, and characterise the *shape* of the entities

–Apply lifecycle model to profile users' NER citations over time and investigate how users' profiles are influenced by global, community, and prior behaviour dynamics

C. Stage 2: Diffusion Analysis

-To do:

–Model the spread of named entities through user profiles (could use multivariate diffusion models here)

D. Stage 3: Forecasting

-To do:

–Implement models to forecast if a user will mention an entity and who that will be (hard!)

IV. Related work

twitter info propagation

reddit compared to other OSNs: "Lifespan and propagation of information in On-line Social Networks: A case study based on Reddit" http://www.sciencedirect.com/science/article/pii/S1084804515001307

network structure of reddit: "Navigating the massive world of reddit: using backbone networks to map user interests in social media" https://peerj.com/articles/cs-4/

twitter nlp

V. Data preparation and NER

To conduct our study, we need to convert 140GB of compressed Reddit posts into a set of interlinked and time-ordered conversations and the entities mentioned in each of them. This provides a number of sub-challenges: sampling of the Reddit data, creating a linked series of conversations, and picking out entity mentions in this text type. Reddit data is largely unexplored in the NLP community, despite the large volume of it and the especially rich metadata. This poses additional challenges: certainly, given the lack of work on Reddit text, there are no annotated datasets available yet, so supervised in-domain work is not directly possible yet. Additionally, the datasets are large, which makes it important to choose a good subset of data on which to do prototyping and development, in order to keep research cycles short. The result that we come to at the end of this stage is a rich dataset for tracking entity and concept diffusion within and across communities.

The Reddit dataset [**?**] is comprised of a sequence of comments, with one JSON record for each one. These are ordered temporally. Reddit itself is roughly similar to a forum, where top-level divisions are made by topic. Within each topic, or *subreddit*, there are posts, which begin with either a short piece of text or a link to an external resource – typically an image, video, or interesting article. Users then may publish comments for each post, and

Fig. 1. Example discussion around a Reddit post.

reply to each others' comments. This leads to a threaded discussion, centred on a particular topic, with a hierarchical comment structure (see Figure V).

A. *Subreddit extraction*

- subreddit extraction

top lists

describe top lists site

pick 100

extract these over whole sample

figures for raw dataset (# per year; total volume over time; volume-rank graph of subreddits; possible ref to appx)

B. *NER for Reddit*

– where is NER first mentioned and defined?

We model micro-topics in conversation as entity mentions. This allows tracking of topics at maximally fine granularity, looking at each user's interests at a low level, as opposed to monitoring broader topics such as "consumer electronics", "politics" and so on. In fact, these broader topics are already explicitly annotated by means of the subreddit topics.

Entity mentions are extracted through named entity recognition. Generally, this task aims to detect the boundaries of certain kinds of entities within a certain piece of text. In this instance, we tokenise text, splitting it into

sentences using the Punkt tokeniser [**?**], and subsequently word-sized chunks, using the `twokenize` tool with some adaptations [**?**]. This tool performs Penn Treebank-style tokenisation, a common standard, with some specific adaptations to enable it to handle the noise present in user-generated text. After this, we take a structured prediction approach to deciding which tokens in each sentence are part of an entity, and possibly the type of the entity. Finally, we concatenate entity tokens, and use these to build a list of entity mentions in any given input text. For example, given the input comment from the source JSON:

> "body": "There are still some really good fighters on this card. Conor McGregor is on the card and so is Gunnar Nelson."

The following output entities should be collected:

> "entity_texts": ["Conor McGregor", "Gunnar Nelson"]

Typically, many NER systems take a supervised approach; that is, they use data labelled by humans as training data, from which features are extracted to form training instances for a machine learning algorithm. However, NLP systems can be hard to transfer between text types; for example, NER systems for newswire might reach about 89% F1 on news articles, but only around 40% on tweets (a form of user-generated content), as found in [**?**]. One approach to overcoming this performance drop when changing text type is to train over a blend of text types. For example, Ritter [**?**] used both IRC[2] and newswire data when developing a part-of-speech tagger for tweets, as well as an unsupervised language model from the target text type. This led to strong performance improvements. We follow a similar approach, using a blend of NE-annotated corpora from both newswire and tweets. The newswire data is drawn from the CoNLL-2003 evaluation task set [**?**]; the twitter data is from Ritter's early experiments and also the W-NUT 2015 shared task [**?**], [**?**].

We start using structured predicting in the form of a CRF to label whole sentences at a time. For features, we use a fairly classical set, and add some unsupervised word representations to this. Our base features are:

•

In addition, we induce Brown clusters [**?**] and use these as word representations [**?**]. Brown clusters are [] These are typically converted to features by shearing [**?**]. This involves only examining the first n bits of a bitstring. However, shearing does not maximise the information preserved in the representation – sub-clusterings at many levels are lost. We therefore experiment with a new method of feature extraction form Brown clusters. We take the cluster identifier at every level, tracing the provenance of a terminal word cluster all the way to the root cluster (which contains all word types). This preserves the entire set membership of any given term, throughout the induced hierarchical clustering. Formally, [] As a result, the lossy nature of shearing-based feature extraction from Brown clusters is avoided.

Feature extraction, training, classification and JSON annotation are all performed using an entity recognition toolkit (https://github.com/leondz/entity_recognition [**?**]), with custom extractors.

[2]Internet Relay Chat – informal internet conversation text

C. Tuning entity recognition

Entity recognition needs to be tuned to fit Reddit data well. There are a number of parameters in our training data balance, feature extraction, and objective function that all reflect the nature of the data and the task at hand. We present our method for estimating of these factors, and intrinsic NER evaluation

- prefer precision; beta = 2 (as per ff-rpg paper)

- describe reddit annotation/tuning annotation approach (skim, find things it got wrong, annotate them, add to set)

- extract reddit tuning set

- estimate Brown c, a based on impact paper findings [**?**]

- compare blended brown cluster src vs. reddit-only

- did this using the generalised-brown package [**?**].

- tune twitter / newswire balance

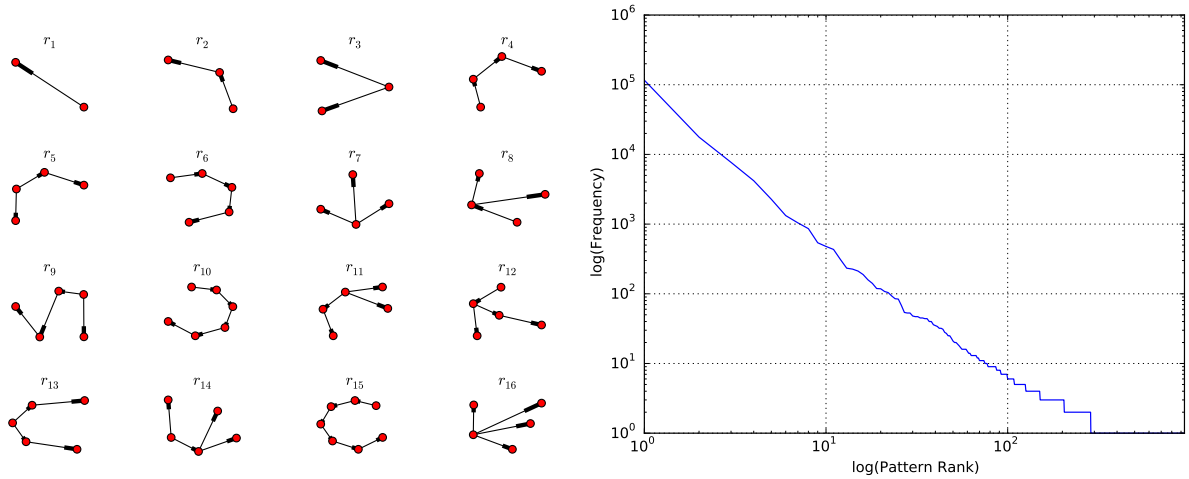
- one, three, four, ten entity classes?

- note that while some figures low, ER toolkit is high-ranking sota software, and the task is simply difficult; generalising NER beyond newswire is not yet well understood.

VI. Entity Diffusion

A. Entity Mention Cascades

-Describe the entity cascade graphs and what these entail -Explain how these were derived using graph isomorphism and then the rank induced



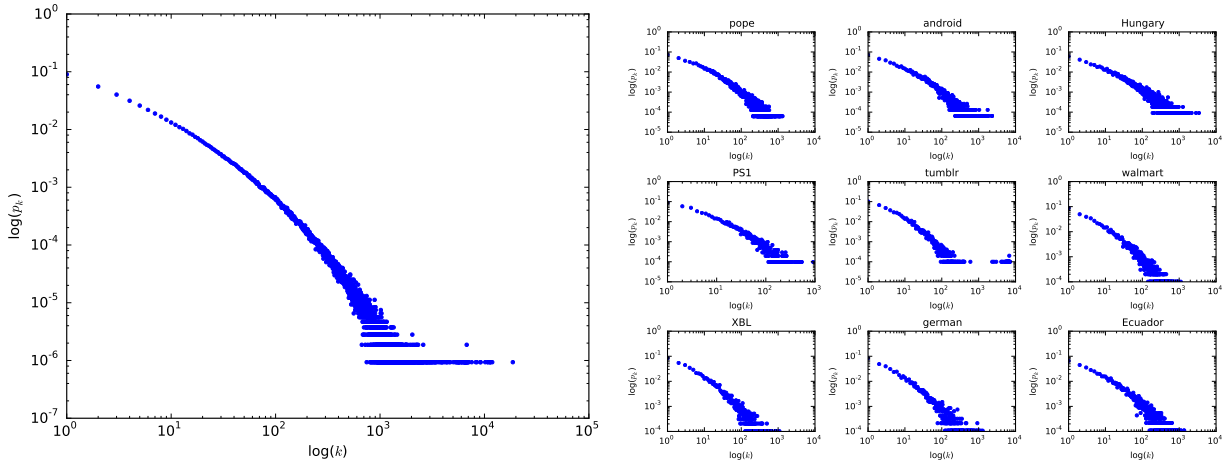
(a) Cascade Shapes

(b) Shape Rank Distribution

Fig. 2. To do

B. Entity Adoption Post-$k-1^{th}$ Exposure

-Potential for adoption at the point of one exposure -Per entity-exposure dynamics - add plots of random selection of entities



(a) Exposure-Adoption Global Distribution

(b) Random Entity Exposure-Adoption Distributions

Fig. 3. To do

C. Global Threshold Diffusion Model

-Explain the implementation of WSDM2010 work and adaptation for non-sequential dataset scans - necessary due to memory load being infeasible

1) Distribution of $\tau_{v,u}$: -Present the distribution of $\tau_{v,u}$ and what this implies for the model

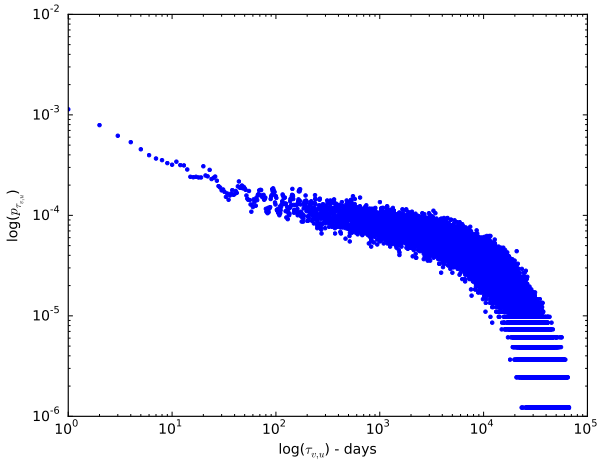


Fig. 4. To do

D. Experiments

-Explain the evaluation process and the computation of the ROC values - per entity -Plot variance in the ROC values achieved