Tracking the Diffusion of Named Entities

Matthew Rowe[a], Leon Derczynski[b]

[a] *Lancaster University, Lancaster, LA1 4YW, UK*
[b] *University of Sheffield, Sheffield, S1 4DP, UK*

Abstract

Existing studies of how information diffuses across social networks has thus far concentrated on analysing and recovering the spread of deterministic innovations such as URLs, hashtags, and group membership. However investigating how mentions of real-world entities appear and spread has yet to be explored, largely due to the computationally intractable nature of performing large-scale entity extraction. In this paper we present, to the best of our knowledge, one of the first pieces of work to closely examine the diffusion of named entities on social media, using Reddit as our case study platform. This forms a data-driven methodology to analyse propagation at large scale. We first investigate how named entities can be accurately recognised and extracted from discussion posts, using an automated approach based upon structured prediction and Brown clustering. We then use these extracted entities to study the patterns of entity cascades and how the probability of a user adopting an entity (i.e. mentioning it) is associated with exposures to the entity. We put these pieces together by presenting a parallelised diffusion model that can forecast the probability of entity adoption, finding that the influence of adoption between users can be characterised by their prior interactions – as opposed to whether the users propagated entity-adoptions beforehand. Our findings have important implications for researchers studying influence and language, and for community analysts who wish to understand entity-level influence dynamics.

Keywords: entity diffusion, information diffusion, named entity recognition, social computing

1. Introduction

Understanding who influences whom and under what conditions forms a core component of information diffusion studies. Recovering the so-called *diffusion process* allows researchers and marketers to understand how messages are passed between users, and what contributes to their flow. In turn, this allows for simulation and predictive models to be engineered that forecast the expected spread of information, allowing such *spread potential* to be maximised or minimised.

[1] Submitted to "Information Processing and Management".

To date, studies of information diffusion on social media and in social networks have concentrated on tracking URLs (e.g. through retweets), link-creation between blogs, hashtags being adopted over time, and group membership adoption; and with different diffusion mechanisms under the microscope (e.g. social contagion, homophily, social reinforcement, rumour spread, structural equivalence, etc.). Despite the rise in such studies, and in tandem the proliferation of data over which studies can be performed, as yet, and to the best of our knowledge, no work has tracked the spread of entity mentions over time – a '*named entity*' here being a proper noun representing a person, place, organisation, or something similar. Understanding *how* named entities diffuse through social networks and being able to *predict* their adoption would provide valuable insights into how topics emerge and spread.

The aim of this paper is to understand how named entities diffuse through social media based discourse, using the online community platform Reddit as the focus of our work. However, in order to study named entities and how they diffuse, we must answer the following three research questions: **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.)? **RQ2:** What process governs the spread of entities? And how does such spread occur? **RQ3:** How can we predict the spread of named entities and who will begin talking about them?

We explored the above questions by devising an approach to recognise entities found in community message board posts – using the popular site Reddit [1] as our study platform. Using the recognised named entities we then carried out a study of how such entities were adopted over time, how they spread, and created an approach to (accurately) predict the adoption of named entities by users based upon the computation of influence probabilities (e.g. achieving ROC value of 0.755 in one instance). The contributions that we make in this paper are as follows:

1. A novel method to recognise and extract named entities found with discussion posts based upon structured prediction and Brown clustering, together with an evaluation of this method.
2. A study of how entities spread and are adopted following exposures, using an approach based upon graph isomorphism to track patterns of entity diffusion.
3. A parallelised general threshold diffusion model (using Apache Spark), based on the work of Goyal et al. [2], that incorporates different entity-adoption constructs (entity propagation, influence of interactions, community homophily) when calculating adoption probabilities – this is accompanied by a comparative empirical evaluation of the different constructs when forecasting entity adoption within the diffusion process.

We have structured the paper as follows: in the following section we cover related work within the areas of named entity recognition and information diffusion – paying particular attention in the latter's case to existing works that

are *similar* to entity diffusion. In section 3 we explain the preparation of the Reddit dataset for our experiments – including down-sampling of 100 subreddits – and the adapted named entity recognition (NER) methodology that we employed. Section 4 presents findings from our analysis of entity cascades (i.e. their shapes and forms) and how exposure frequency and entity-adoption probability are associated. This section also describes our implementation of the parallelised general threshold diffusion model and experiments that assess the efficacy of various influence constructs in the entity-diffusion process. Section 5 discusses the findings that we have drawn from this work and plans for future work, and section 6 finishes the paper with our conclusions.

2. Related work

In this work we investigate how entities diffuse over time through the online community platform Reddit. Diffusion of information is a well studied topic, and is of particular interest today given the myriad ways in which Web users consume information and are thus influenced by what they read, and with whom they interact with, online. We first review state of the art approaches for recognising named entities, before then describing existing works that have studied information diffusion.

2.1. Named Entity Recognition

Named entity recognition (NER) has been a long-recognised NLP task, the earliest major instances of which are perhaps the ACE and MUC challenges. The goal is typically to extract mentions of certain types of entities, like organisations, locations or person names. Generally, NER systems can be structured in terms of representation, induction, dependency modelling and integration of real-world knowledge [3, 4].

While initially conducted over newswire [5], older tools tend not to perform so well on modern text types, such as tweets and other short social media text [6]. Simultaneously, the value in non-newswire data has rocketed: for example, social media now provides us with a sample of all human discourse, unmolested by news editors, publishing guidelines and the like, and all in digital format – leading to whole new fields of research opening in computational social science, examining e.g. demographics [7], personality [8] and income [9].

Research on NER for social media content is, accordingly, a hot area, including general approaches [10], topic-specific approaches [11], adapting from known genres [12]; these are driven by and evaluated in multiple recent shared tasks [13, 14]. The task is generally cast as a domain adaptation problem from newswire data, integrating the two kinds of data for training [15] or including a lexical normalisation step [16] to drag the problem back to territory more familiar to existing models and methods. Two major perceived challenges are that NEs mentioned in tweets change over time [17], and that diversity of context makes NER more difficult [6]. Here we address NER without any in-domain data, which leads to a successful and transferable approach.

2.2. Information Diffusion

Studies of information diffusion have largely concentrated on *deterministic* signals of diffusion such as tracking URLs, hashtags, quotes [18], and adoption behaviour (e.g. group signups); however to the best of our knowledge such studies have yet to focus on how entities diffuse. We now focus on key pieces of work that are closely-aligned to the study of entity-diffusion in the context of social networks – should the reader wish to know more about information diffusion models, and in greater detail, then please refer to Guile et al.'s [19] comprehensive survey of such models.

The study of information adoption and sharing was undertaken by Bakshy et al. [20] who conducted a large-scale randomised controlled trial to examine the effects of *information exposure* on information diffusion, using the Facebook platform. The authors were able to assign Facebook users *randomly* with a 1/3 probability to a *feed* group, and the remainder to a *no feed* group and then *hide* information (i.e. status posts) posted within the latter's group. Bokshy et al. found, unsurprisingly, that users who were *exposed* to information (i.e. those in the feed group) from their friends are more likely to share it on – implying that such exposure has an influential effect.

The closest work to the study of *entity diffusion* can be found in studies of hashtag diffusion. For instance, Romero et al. [21] studied the spread of the top-500 hashtags posted in a sample of $>$ 3B tweets collected over a six-month period, finding that users were most likely to *adopt* a hashtag (i.e. mention/cite it in their Tweet) after receiving 4 exposures from their friends. The authors found marked differences in the adoption of hashtags based on their topics, something which – as we will show below – is not present in entity diffusion. More recent work by Yang et al. [22] studied both the role of hashtag content and the role of the hashtag in a community, finding that both factors are associated with hashtag adoption. Our work differs from [22] by studying the adoption of entities based on pairwise interactions between users – i.e. how one user influences another to adopt an entity – as opposed to the content properties of the entity.

The different modalities of diffusion signals encompass the adoption of behaviour by users from previous adopters, for instance the work of Goyal et al. [2] tracked the diffusion of *actions* on Flickr, where actions were defined as users joining a group (i.e. a photography-topical group). A general threshold model was proposed that determines the probability of influence between two arbitrary users based on the relative frequency of action propagations observed before, divided by the absolute number of actions of the user responsible for the propagation. The authors found that computing the average time of influence between two users led to more accurate computation of influence probabilities. In this paper, we use the general threshold framework from [2], but extend it into the entity-mention setting, hence: we track the *mention* of an entity by a user over time and calculate the probability of influence that an *adopter*'s neighbours have had upon him. Furthermore, we also extend this framework to test for two additional constructs: (i) influence of interactions before adoption

(i.e. did the degree to which an individual communicated with a previous entity adopter influence their own adoption?), and (ii) community homophily (i.e. does the similarity between users' interests – based upon similarity in subreddit posting – have an effect on adoption of an entity?).

Prior work on Reddit has examined the site's evolution since launch, seeing it evolve from a bulletin-like page to a large community site with many segragated and unique sub-communities that reinforce a general perception of the overall community [23]. This observation supports the use of Reddit as a study venue for information diffusion, finding that communities are large, well-defined, and cohesive. Later work covers the mapping of popular content [24] and of network structure [25], though not the diffusion of information through those networks.

Fang et al. [26] presented a method to predict adoption probabilities in social networks by controlling for potential confounding, unobservable variables – proposing a modification of expectation-maximisation to induce a Naive bayes predictive model. The authors found that social influence alone is insufficient to recover the diffusion process, and thus external factors – that are latent – must be countered for within any predictive model – this was in the context of predicting the adoption of social ties. The adoption of information within a social network and its propagation was studied by Huang et al. [27] by considering the role of temporal dynamics. The authors found that the probability of diffusion between users (*retweets* on Chinese microblogging platform Sina Weibo) reduces as a function of time from the last interaction between the users, thereby suggesting that *temporal dynamics* have a strong effect in diffusion. We build time *explicitly* into our adaptation of Goyal et al.'s [2] general threshold diffusion model – by comparing static and discrete time versions of adoption probabilities.

3. Datasets preparation and NER

In order to study entity diffusion at a *large-scale* we used the entire dump[2] of Reddit from its inception through to July 2015 – this provided a dataset of 140Gb of data compressed containing $\sim$ 1.7B posts (i.e. original thread starter posts and comments). As we will describe below, we also required corpora from which we could *model* and *train* our named entity recogniser – and also assess its performance – therefore we also used the following three datasets: (i) *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; (ii) *Twitter data (unannotated)*, comprised of a large corpus of English tweets taken from an archive of the garden hose feed, and; (iii) *Twitter data (annotated)*, comprised of datasets annotated with named entities – one from Ritter's 2011 EMNLP paper [10] and a second from W-NUT 2015 shared task [14].

To conduct our study, we needed to convert the full 140Gb of compressed Reddit posts into a set of interlinked and time-ordered conversations and the

[2] `https://archive.org/details/2015_reddit_comments_corpus`

Figure 1: Example Reddit post. Note topic at the top, followed by comments, with conversations descending in a tree-like hierarchical fashion

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 its large volume and 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. Additionally, the datasets are large, which makes it important to choose a good subset of data on which to do prototyping and development. The result that we come to at the end of this stage is a complete dataset for tracking entity diffusion within communities.

The Reddit dataset [28] 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 reply to each others' comments. This leads to a threaded discussion, centred on a particular topic, with a hierarchical comment structure (see Figure 1).

3.1. Subreddit extraction

The Reddit dataset is large, and needed to be pared down for initial analysis. Fortunately, the data is segmented by community, meaning that we could adjust the scope of the study arbitrarily by selecting a certain number of subreddits and taking the whole. This is in contrast to, for example, Twitter, where reducing the sample is performed by reducing the sampling of posts [29], thus leading to broken conversation threads and so on. We chose to examine one hundred entire individual subreddit communities. The subreddits were chosen from a list of top subreddits,[3] which ranks communities based on levels of activity, numbers of subscribers, and rates of growth. The list of chosen communities can be found in the github repository of this work.[4]

3.2. NER for Reddit

In this diffusion analysis, 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 [30], and subsequently word-sized chunks, using the `twokenize` tool with some adaptations [31]. 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"]*

[3] `http://redditlist.com/`

[4] `https://github.com/mattroweshow/NER-Diff-Paper/tree/master/data`

We present named entity recognition here, adapted and applied to Reddit posts and conversations, a text type for which to our knowledge there have been no prior NER efforts. Notably, we experiment with techniques previously demonstrated to be successful on other user-generated content and find them lacking.

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 [6]. One approach to overcome this performance drop when changing text type is to train over a blend of text types. For example, Ritter [10] used both IRC[5] 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 [5]; the twitter data is from Ritter's early experiments and also the W-NUT 2015 shared task [10, 14].

We start using structured predicting in the form of a Conditional Random Fields (CRF) model 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:

- Lower-case form of word;
- Word prefix and suffix (two- and three-character);
- Previous and next word;
- Flags set if the word is all uppercase, titlecase, or digits;
- These flags for the previous and next words;
- The next and prior bigrams.

In addition, we induce Brown clusters [32] and use these as word representations [33]. Brown clustering is a form of hierarchical agglomerative hard clustering, using average mutual information as its objective function. It takes as input a corpus, in the form of a sequence of words, and in its generalised form [34], a single hyperparameter: the size of its active set a. This active set is filled with the a most-frequent classes drawn from all word classes C, with one word per class at initialisation.

The mutual information of two classes, $C_i, C_j \in C$, denoted $MI(C_i, C_j)$, is:

[5]Internet Relay Chat – informal internet conversation text

$$MI(C_i, C_j) = p(\langle C_i, C_j \rangle) \ \log_2 \frac{p(\langle C_i, C_j \rangle)}{p(\langle C_i, * \rangle) \ p(\langle *, C_j \rangle)} \quad (1)$$

The average mutual information of C, denoted $AMI(C)$, is the sum of mutual information of all cluster pairs in C:

$$AMI(C) = \sum_{C_i, C_j \in C} MI(C_i, C_j) \quad (2)$$

Brown clustering proceeds by greedily merging the pair of classes within the active set that causes the least loss to average mutual information, until all classes are merged. A fuller definition is provided in [34]. The result is a sequence of binary merges, describing the set membership of each word type in the corpus as the merges progress. For each single word type, therefore, the path to a destination cluster can be described as a bitstring, which details the sequence of binary merges taken. The zero-length bitstring describes the situation at the top of the hierarchy, where there is one class.

These bitstrings are typically converted to features by *shearing* [34]. 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 introduce in this work a new method of feature extraction from Brown clusters, which we call *provenance-based* feature extraction. 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. For example, given the bitstring `1100101`, the following text features are generated: `1`, `11`, `110`, `1100`, `11001`, `110010`, `1100101`. If the typical bit depths [4] of 4, 6, 10 and 20 were chosen, only the following features would be generated: `1100`, `110010`, `1100101`. As a result of taking all directly-relevant features in the merge list, 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 [35],[6] with custom extractors.

3.3. 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.

In terms of evaluation, we prefer recall over precision. Over the large dataset, spurious entities (i.e. false positives) are likely to be seen rarely. Mis-recognised entity names tend not to be distributed in a few high-frequency clumps, but rather as many different terms, each with a lower frequency. This suggests that

[6] https://github.com/leondz/entity_recognition

there will be great variation in their surface forms, leaving them in the long tail of entities discovered [36]. As our diffusion analysis concerns the more frequent entity lexicalisations, these infrequent spurious mis-recognitions are less likely to have an impact. Conversely, recall expresses how broadly and comprehensively the extraction is performing, and is important to tracking a range of entities.[7] That is to say, the problem addressed is more tolerant to low precision in input data than low recall. We can therefore better evaluate our systems using an adjust F_β score.

$$F_\beta = (1 + \beta^2)\frac{PR}{(\beta^2 P) + R} \tag{3}$$

When $\beta = 1$, precision and recall are balanced in a harmonic mean, e.g. F1-score. That is, false positives and false negatives impact results equally. Given precision P and recall R, typically an F-score is drawn from F_β with $\beta = 1$. To score away from false negatives, i.e. missed entity mentions, we set $\beta = 2$.

Our approach here is to tune an entity recogniser with reference to a dataset that matches the target text type. We draw this development set from Reddit posts, using comments encountered during our work that appear to have missing or spurious annotations. These are then isolated, tokenised, and manually annotated. Our annotation format follows [10] in using the Freebase top-level entity type inventory, but only uses the chunking information, as nothing further than this is needed: only the surface forms of entities. In total, we identified and annotated $3,708$ tokens of Reddit data, including 149 entity chunks. This comprised our development set, which was used to tune a variety of parameters in our approach. Evaluation was performed using the standard `conlleval.pl` tool for entity chunking evaluation, on all systems. This gives results at entity level (as opposed to token level).

In addition, we draw supervised data for multiple datasets in order to approximate the Reddit text type. We take data from Twitter, taking the union of corpora used in previous work that follow the Freebase [37] ten-class entity scheme. The classes given are company, facility, geo-loc, movie, musicartist, person, product, sportsteam, tvshow and other. This scheme gives broader coverage than e.g. the three-class ACE named entity scheme, in a number of ways. It differentiates geo-political entities from smaller granularity locations like buildings; it handles the often-metonymous categories of sports team and music artist (which often present ambiguities between location and organisation, or person and organisation, respectively); and it covers products, as well as introducing a few specific product types that may have unusal names (TV shows and movies). This scheme has proven successful on social media corpora before [10, 14]. For newswire, we use the Reuters RCV1 corpus annotations that were part of the CoNLL-2003 shared task [5]. Classes are removed before training, making this just a chunking task. We evaluated performance when

[7] N.b. It has often been more challenging to achieve high recall in social media texts than high precision [10, 6].

Table 1: Reddit development set NER, varying text type in training data and Brown cluster source. Best per scenario is starred; best overall, bold.

Brown cluster source	**Precision**	**Recall**	**F1**	**F2**
Baseline				
Stanford (3-class builtin)	**87.88**	38.93	53.96	47.81
Newswire training data				
RCV Newswire	63.57	59.73	61.59	60.96
Tweets	62.75	64.43	63.58	63.86
Blended tweets/news	*68.42	61.07	64.54	63.34
Reddit	66.32	**67.11**	**66.71**	**66.84**
Stanford baseline	63.97	58.39	61.05	60.14
Twitter training data				
RCV Newswire	*73.02	30.87	43.39	38.22
Tweets	70.37	38.26	49.57	45.12
Blended tweets/news	65.28	31.54	42.53	38.10
Reddit	76.34	*47.65	*58.68	*54.47
Stanford baseline	65.22	30.20	41.28	36.78
Blended training data, 50:50				
RCV Newswire	66.67	42.96	52.25	48.74
Tweets	66.10	52.35	58.43	56.25
Blended tweets/news	68.69	45.64	54.84	51.39
Reddit	*70.08	*59.73	*64.49	*62.82
Stanford baseline	67.77	55.03	60.74	36.78

trained on just Twitter data, just newswire data, and also a blend of the two. In the base cases, the same amount of data was used. This was capped by the volume of Twitter training data available, 66k tokens; so, the newswire approach was also trained with 66k tokens The blended version used even amounts of both, totalling 132k data, to see if adding the two types of data improved performance. An even split was chosen because, if one is to search for the best split, this is the optimal place to begin a binary search.

The baseline system was the Stanford NER tool [38]. We included two variants: one run of the out-of-the-box stock system, using the `english.all.3class.distsim` binary, and another with a first-order model trained on the same source data as our system. The properties files for these runs are included in the github repository.[8]

Firstly, we tuned our word representations. Specifically, we needed to estimate the number of Brown clusters C to use in feature extraction. We also then used this to determine what blend of social, Reddit or news data gave best results in unsupervised feature generation and extraction. To tune the value for C, we examined a similar scenario with similar dataset sizes, and estimated an optimum. We noted that in prior work [39], entity recognition performance with decomposed class prefixes – similar to the provenance-based features we propose here – peak at around $C = 2500$ for corpora of 16k tokens, $C = 5000$ for corpora of 32k tokens, and at higher values for larger datasets. As General Brown clustering is dependent on the number of types and the size of the active set a, and results are unreliable with $a > C$, we set $C = a = 2560$. This offers a decent trade-off between computational cost of building clusters, and the quality of the clusterings used. We then experimented with combinations of newswire, Twitter and Reddit data. Brown clusters are extracted using the generalised-brown package [40]. Results are given in Table 1.

Note how the scores are consistently best in each category when inducing Brown clusters from Reddit data. Attempts to approximate this using newswire, twitter, or a blend of those two did not score as well. This is remarkable, considering that we used on 64M tokens of Reddit data for cluster induction, compared to around 130M total for the other text type blends. Even half the amount of in-text-type data provides notably improved unsupervised representations than an approximation. This illustrates the importance of taking the target text type into account during unsupervised feature extraction. In fact, this is good news: we can improve NE performance in a new genre without doing any human annotation of text in the genre, as long as a sample of the text is available. The Twitter Brown clusters consistently give second-best F2 scores, leading newswire.

We also examined the learning rate of our entity chunking system, scaling up the RCV newswire data. We experimented with pure newswire and also newswire plus tweet training data, and with pure newswire vs. blended clusters. Results are given in Figure 2.

[8] `https://github.com/mattroweshow/NER-Diff-Paper`

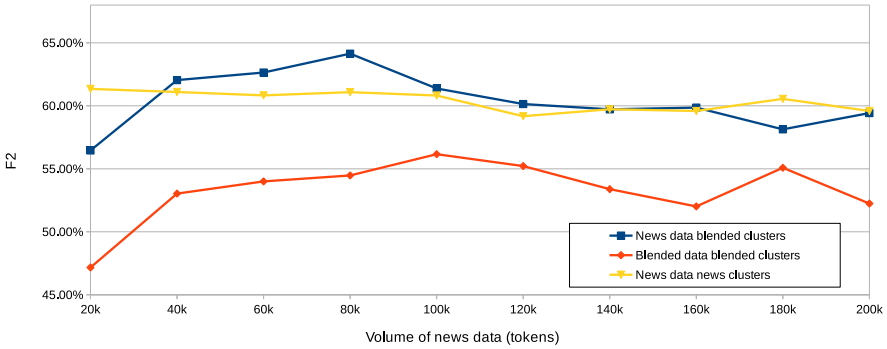


Figure 2: Scaling news training data under three conditions: news clusters; blended news and twitter clusters; blended news and twitter clusters, with 66k extra twitter training data. Note increasing news training data (x-axis) eventually lead to decreased performance in every case.

Unlike prior experiences with NLP on user-generated content, blending training text types in supervised learning did not lead to improved performance. They suggest that adding too much newswire reduces performance, and that sticking to just newswire clusters also reduces performance.

Based on this data, we hypothesised that insufficient regularisation had lead to overfitting. To test this, noting the downward turn in newswire-trained blended cluster performance after 80k tokens (Figure 2), we re-ran the experiments with 100k newswire data using a c_2 regularisation penalty of 10^{-1} compared to the default 10^{-3}. The prior performance, F2 61.38%, rose to 62.07%; so, while effective, still a marginal increase. Therefore, we continued using newswire training data with Reddit clusters, the highest performing option (Table 1).

A potential route for improvement is to manually label a quantity of Reddit data for named entities, but given its prohibitive cost, we find we have a strong enough solution from near-genre supervised training data and in-genre unsupervised feature representations (Brown clusters). Hence, we used training data composed from newswire, with Reddit Brown clusters, for annotating the named entities across the 100 sampled subreddits.[9]

3.4. NER analysis

It is interesting to note that adding the other UGC chunking data (from Twitter) is often not helpful. Also, performance generally drops when adding extra newswire data. This drop reflects the findings of earlier work [10], where a little Twitter training data improved social media NE chunking performance more than additional newswire training data.

[9]N.b. When predicting entity diffusion we used the top-500 entities which were manually verified for accuracy, and were found to be correct, qualitatively bearing out our earlier assumption of spurious entity surface forms not be clustered with high individual frequencies.

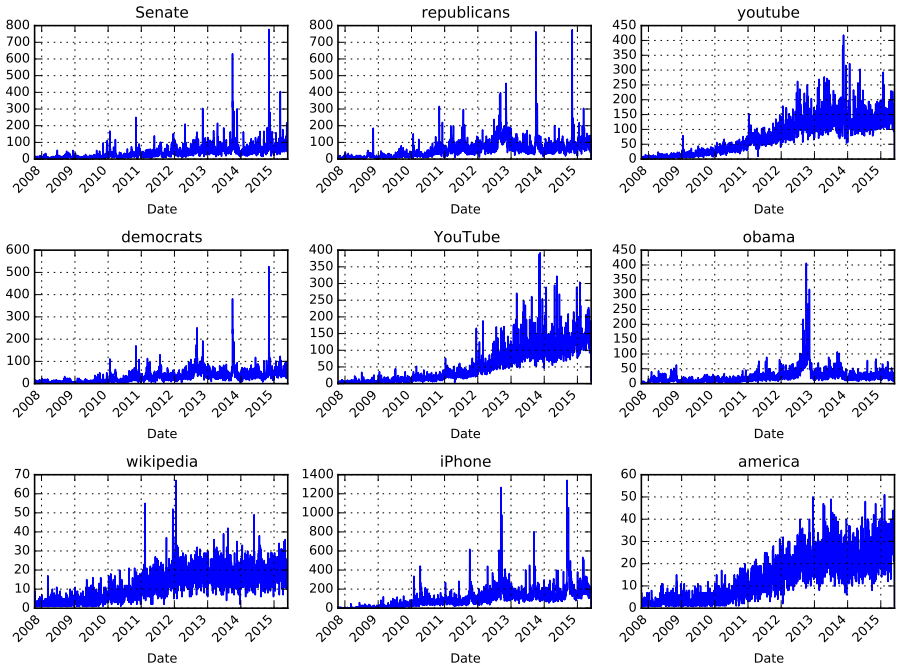


Figure 3: Top nine entities and their mention intensity over time

Based on these NER results, we ran entity chunking over the subreddits selected. Examining entities by mention count, we were then able to build a longitudinal analysis of popular entities. Figure 3 shows the mention intensity of the top nine entity surface forms over time. Note the field effect of Reddit growth pushes entity mention frequency up universally over time, and that spikes corresponding to major events. For example, Obama peaked in late 2012, around the national election in the US; and iPhone mentions peak around iPhone releases.

Note that while some figures seem low when compared to typical newswire level performance, the toolkit used is high-ranking, state-of-the-art research software, coming third in the 2015 W-NUT challenge for entity chunking over tweets. The task is simply difficult; Twitter NER recall has always been low. In addition, it is a generally consistent finding that generalising NER systems beyond newswire is not yet well understood; systems that perform very well on this text type (e.g. F1 of 0.89 from Stanford NER [6]) can often score very poorly on social media content (F1 of 0.41). This may be due to overfitting of tools to newswire over time, due to community challenges, dataset in just one type, extra custom rules adapting to formal news text, or other things – but this is beyond the scope of this paper. We do note that our approach uses largely unsupervised feature extraction and performs better than on other social media corpora, also beating the Stanford NER system, in this first attempt at named entity chunking for Reddit.

4. Entity Diffusion

In this section we now move on to examining how the recognised named entities emerge and *diffuse* through the analysed subreddits. As per prior work, one of the first things that we can inspect is the *shape* of entity mention cascades: that is, the patterns of diffusion that such entities exhibit when cited in conversation chains. We begin by explaining how such patterns are derived, before then moving on to showing what patterns emerge.

4.1. Entity Mention Cascades

Prior work by Leskovec et al. [41] examined the shapes of hyperlink cascades through the blogosphere to identify patterns of link diffusion. We follow a similar process here, however we instead inspect the emergence of entities in conversation chains in Reddit. We first make the following explicit.

Definition 1. *(Entity Cascade) A cascade of* $< p_i, pj > \in C_e$ *of an entity* $e \in E$ *occurs when two or more posts citing the entity are chained together in a reply graph. Hence:* $C_e = \{< p_i, p_j >: p_i \rightarrow p_j \in R, cites(p_1) = cites(p_j) = e\}$.

Our goal is to derive all cascades for each entity in our analysis, and then examine how the shapes and sizes of these cascades differ. To gather each entity's cascades, we retrieved all (of the 100) subreddit posts that contained a given entity e. Then, for each post ($p \in P_e$) we recovered the reply-chain

that that entity appeared within – this was performed by going *up* the reply chain from p to its parent post (i.e. the post that p was replying to) and *down* the reply chain by getting the posts that replied to p. When iterating through the posts, if we came across a post that replied to another post in an existing chain then that post was added to the chain. We only maintained posts within the chain that cited the entity in question: this produced entity cascades where each consecutive post in the chain mentions the entity – we refer to this as *strict cascade derivation*, as we do not consider posts higher-up or lower-down the reply chain that cite the entity yet are connected by a non-entity citing post.[10]

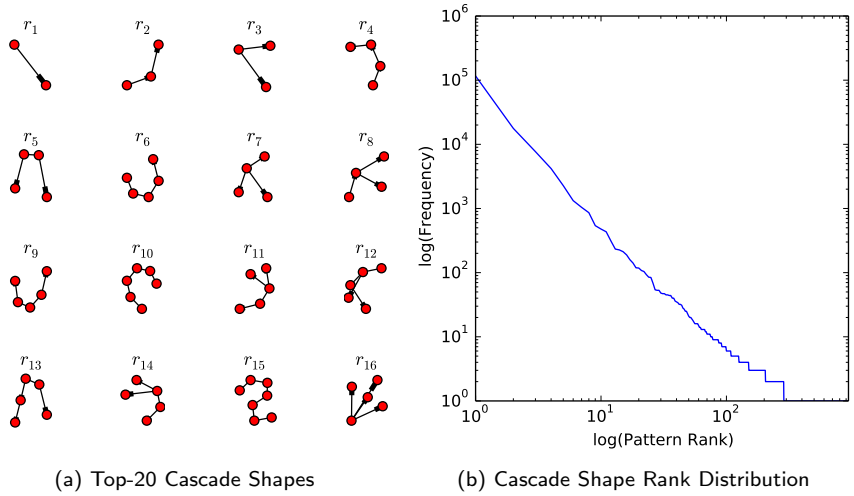
This process produces, in essence, a collection of cascade graphs for each entity, each of which may have isomorphic shapes yet contain different node labels (i.e. different post ids). We reduced each entity's cascade graph collection down to a frequency distribution of the *canonical form* of each graph using Cordella at al.'s [42] graph isomorphism approach. A further reduction was run to compile a frequency distribution of the cascade shapes across all entities. Fig. 4 show both the top-20 entity cascade shapes on the left (Fig. 4a) and the ranking of the patterns' frequencies on a log-log scale (Fig. 4b). Upon inspection, one thing becomes immediately apparent: entity cascades are shallow and short at the top-3 ranks, however after this position we start to see chains of discussions as being popular which are deeper and narrower. This result contrasts somewhat to prior work [41] where cascades of hyperlinks between blogs were shallower in depth yet wider – in terms of the breadth of diffusion at the first level from the seed. The ranking of the patterns follows a general power-law distribution where a small section of patterns (i.e. the top-20) are seen most often – this is somewhat expected as it would be very rare for an entity to be cited in a long thread with many branching reply-chains.

4.2. Entity Adoption Post-$(k-1)^{th}$ Exposure

Inspection of the shape of entity cascades through Reddit discussion threads reveals some interesting traits, suggesting that an entity *spreads* through narrow diffusion paths – i.e. with little branching occurring. One natural question that emerges from this is to question the extent to which exposures to an entity play a role in actually adopting (i.e. citing) the entity in question. To investigate the relationship between exposures and adoptions, we took the top-500 entities from our whole annotated dataset and calculated the probability of a user adopting an entity after being *exposed* to the entity k times, defining an exposure as follows:

Definition 2. *(Exposure) A user u is exposed to an entity e at time t if a given post $p \in P^{\Gamma(u)}$ authored by a neighbour of u (i.e. $v \in \Gamma(u)$) contains the entity e, where neighbours interacted with u prior to t.*

[10] Chain-derivation Python code can be found here: `https://github.com/mattroweshow/NER-Diff-Paper/tree/master/diffusion-scripts`



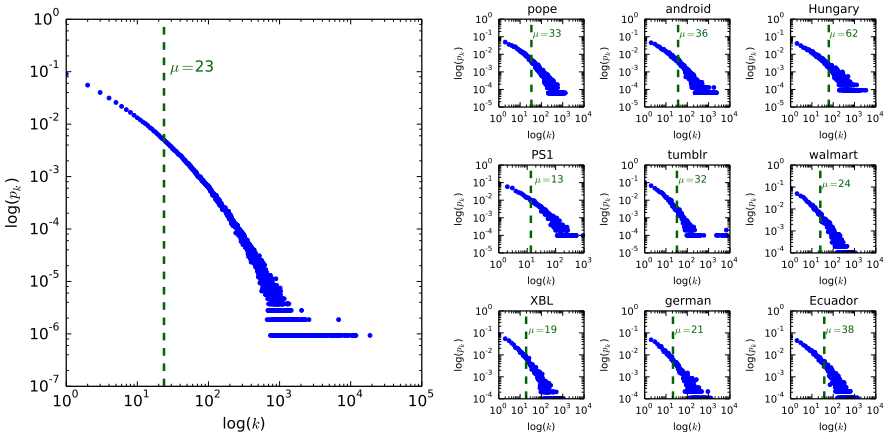
(a) Top-20 Cascade Shapes

(b) Cascade Shape Rank Distribution

Figure 4: The top-20 cascade shapes are generally deep and narrow with little branching (Fig. 4a, while the cascade shape rank follows a power-law distribution (Fig. 4b.

Based on this definition we iterated through all posts chronologically that cited a given entity, and if the post was the first time that a user cited the entity (i.e. he/she was not *activated*) then we counted how many *exposures* the user had received prior to the time of the post – logging this as k. Fig. 5a presents the *overall* plot of the probability (i.e. relative frequency) of users adopting an entity after k exposures to the entity. Immediately, one can note that the mode of this distribution is at 0 and that the mean is $k = 23$: this implies that users are most likely to actually cite an entity without having been exposed to it, in fact $P(adoption) \rightarrow 0, k \rightarrow \infty$. We are somewhat guarded in *generalising* from this result, as our experimental setup here – given the scale of the data we are playing with and the tractability of annotating the entirety of Reddit – does result in only a fraction of Reddit being annotated with entities. Hence, it is possible that entities emerge from other subreddits, yet we are unable to capture this at present – our future work makes suggestions as to how this effect can be validated. Furthermore, this finding contrasts somewhat to existing patterns of *hashtag* adoption [21] where there is a clear mode at around $k = 4$ exposures, after which the probability of adoption curtails. This difference is likely due to two factors: firstly, the differences of the platforms – as Twitter acts as a public broadcast spectrum where information is presented in feeds and is then passed on, while Reddit is more interaction and discussion-driven, and; secondly, the manner in which users are *exposed* to information – on Twitter this is via subscriptions to other users and observing trends in the trending topics area, while Reddit requires users to read through threaded discussions and *notice* entities within.

The second plot below (Fig. 5b) shows a sample of 9 entities' adoption-exposure distributions, all of which have similar shapes (with a mode at 0) and a heavy-tail. Variance exists, however, in the means that these distributions have, for instance the entity *PS1*[11] has a much lower mean than the entity *Hungary* suggesting that users require less stimulation to discuss the former than the latter. The nature of how and why the distributions differ is something that requires further investigation – for instance, clustering the entities per-topic and then examining the *pooled* exposure-adoption graphs would allow one to understand topic-level patterns of spread (in a similar vein to [21]).



(a) Exposure-Adoption Global Distribution (b) Sampled Entity Exposure-Adoption Distributions

Figure 5: The probability of a user adopting an entity as a function of k prior exposures to the entity has a heavy-tailed distribution (Fig. 5a) that is consistent across all entities, including a sample of 9 random entities (Fig. 5b).

4.3. Global Threshold Diffusion Model

We now move on to forecasting the diffusion of entities across Reddit. For this, we used a modified implementation of Goyal et al.'s general threshold model [2] to parallelise the computation of the model. The core principle of the model is that one can calculate the probability of a user (u) adopting an entity (e) based on how their neighbours ($v \in \Gamma(u)$) have influenced them previously. Hence, the probability of u adopting an entity is calculated as follows:

$$p_u\big(\Gamma(u)\big) = 1 - - \prod_{v \in \Gamma(u)} \big(1 - - p_{v,u}\big) \tag{4}$$

[11] Denoting the original Playstation video-games console.

In Goyal et al.'s prior framework, the probability of influence ($p_{v,u}$) of v on u is based upon the maximum likelihood estimate of a single Bernoulli trial. An entity propagation occurs between v to u when the latter cites e after being exposed to it by the former (as per Definition 2), hence a count of how many entities propagate between v and u can be recorded in E_{v2u}. From this, the influence probability between v and u based on such *propagation* can be calculated as follows, where E_v is how many times v has cited an entity:

$$p^E_{v,u} = \frac{E_{v2u}}{E_v} \tag{5}$$

The authors explain how there are two variants of this calculation: (i) a static Bernoulli random trial where Equation 5 is calculated from the training set (ignoring time), and: (ii) a discrete time Bernoulli random trial where counts are only placed within E_{v2u} and E_v if the citation of an entity is within a discrete time interval, that is: if the time that u adopts an entity e is given by time t_u then E_{v2u} and E_v are composed from the entity posts of v which each have time $t_v \in [t_u - -\tau_{v,u}, t_u)$, where $\tau_{v,u}$ is derived as follows (only considering $v, u \in U$ (set of all users) if u has contacted v prior to t_u:

$$\tau_{v,u} = \frac{\sum_{e \in E} (t_u(e) - -t_v(e)}{E_{v2u}} \tag{6}$$

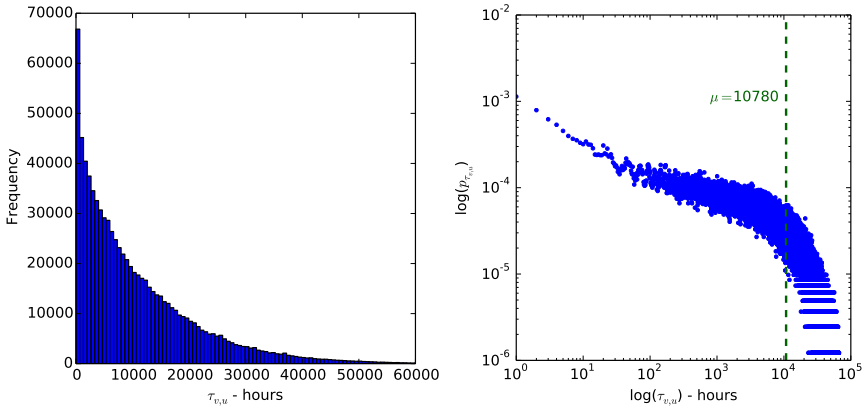
Fig. 6a shows the binned distribution of the $\tau_{v,u}$ values: one can note how the distribution has a right skew with the mode of the distribution (roughly) being at one hour, this then gradually curtails off with fewer people having larger *influence windows*. The log-log plot of the relative frequency distribution (Fig. 6b) shows the *heavy-tail* property of the distribution, and that the mean window width is $10,780$ hours (≈ 449 days ≈ 1.2 years), thus suggesting a degree of *stickiness* in the Reddit communities where people remain for long periods.

4.3.1. Additional Influence Dynamics – Entity-Adoption Constructs

The neat formulation of the general threshold model, and the monotonic-submodular nature of the probability of adoption function ($p_u(\Gamma(u))$, means that we can vary the mechanism by which we derive the *influence probability* ($p_{v,u}$) between two users v and u to test for different influence effects – we refer to these as *entity-adoption constructs*. Our contribution here is to test for the influence of *prior interactions* and *community homophily* using the general threshold model. To compute the influence probability based on interactions, we derive $p^I_{v,u}$ as follows:

$$p^I_{v,u} = \frac{|\{p_u : p_v \in P_v, p_u \in P_u, p_u \rightarrow p_v\}|}{|\{p_u : p_u \in P_u, p_u \rightarrow .\}|} \tag{7}$$

Where P_u and P_v denote the set of posts by users u and v respectively, and $p_u \rightarrow p_v$ indicates that post p_u replied to post p_v. The influence probability based upon community homophily ($p^C_{v,u}$) is derived as follows:



(a) Binned Frequency Distribution of $\tau_{v,u}$ values(b) Log-log Plot of the Binned $\tau_{v,u}$ distribution

Figure 6: The influence window ($\tau_{v,u}$) between two arbitrary users characterises the average time for an entity to propagate from v to u. In hours, this value has a *right-skew* (Fig. 6a), while the log-log plot (Fig. 6b) of the relative frequency distribution demonstrates the heavy-tail nature of the distribution with a mean of $10,780$ hours (≈ 449 days ≈ 1.2 years).

$$p^C_{v,u} = \frac{|C_u \cap C_v|}{|C_u \cup C_v|} \tag{8}$$

Where C_u and C_v are the sets of subreddits that u and v has posted in respectively. We calculate the same two variants as the entity-propagation influence probability as above (static Bernoulli, and discrete-time Bernoulli), for the discrete-time case we only consider interactions between u and v that fall within $[t_u - -\tau_{v,u}, t_u)$ (interactions-based) and posts within subreddits by u and v that were made within $[t_u - -\tau_{v,u}, t_u)$ (community-homophily).

In order to eliminate bias in our below experiments – when we attempt to forecast entity adoption for users – we divided the top-500 entities into an 80%:20% split for training and testing respectively. Then, for the above influence probabilities (entity-propagations, interactions-based, community-homophily) we used different strategies for their calculation. For the entity-propagation influence probability ($p^E_u(\Gamma(u))$) we used the training segment to calculate the values of E_{v2u} and E_v, and also $\tau_{v,u}$ – for all pairs of interacting users within the training segment – this follows the experimental setup of [2]. One thing that is somewhat limited about this approach, is that we are observing future effects when calculating E_{v2u} and E_v that we take forward into our experiments, as we observe how influence has occurred between users prior to an adoption happening. To some extent, this is somewhat unavoidable in the context of the dataset splitting as $\tau_{v,u}$ must be calculated somehow – an alternative for future work here is to use a fixed time-split and use the first 80% of entity-posts for

training and the rest for testing.

4.4. Experiments

We now move on to forecasting the adoption of named entities by users as they spread through Reddit. To this end, we used an experimental setup that induces the joint probability function (Eq. 4) on a per-entity basis within the test set: each user's probability of adoption was computed as product of their neighbours' influence. Our goal therefore was to examine which of the above entity-adoption constructs were best suited to predicting adoptions.

4.4.1. Experimental Setup

Using the 100 randomly sampled subreddits and running the above Named Entity Recogniser over these subreddits' posts, resulted in a total of > 300 million posts in our dataset (using only those from the 100 subreddits) written by $4,139,814$ users – the entity recogniser also extracted $8,797,271$ unique entities. For our experiments we tested 6 different models that resulted from permutations of the 2 probability settings (i.e. static Bernoulli or discrete-time Bernoulli (i.e. $t_u \in [t_u - -\tau_{v,u}, t_u)$) and the 3 entity-adoption constructs (entity-propagation, interactions-based, and community-homophily).

Deriving Adoption Probabilities. In order to test which model was best (from above) we took the entities within the test set, and ran the following process: we chronologically ordered each entities' posts and then iterated through the posts set one-by-one. For each post's author (v) we then checked if they had been *activated* before – i.e. had they cited the entity? – if not, then this would be first time they had cited e. If this was the case then we retrieved the prior interactions that the user had had and calculated (for each prior neighbour – $u \in \Gamma(v, t_u)$) the probability of influence between v and u using the above influence probability variants (e.g. interactions-based with static Bernoulli setting). We then updated the probability of adoption of u. By iterating through the set of time-ordered posts we maintained adoption-outcome tuples of the form $< u, p_u, r_u >$ where $r_u \in \{0, 1\}$ denoting whether the user ultimately adopted the entity e or not. Our evaluation of the models used these tuples to calculate the area under the Receiver Operator Characteristic (ROC) curve, aiming to achieve a value of 1 (for perfect prediction).

Parallelising Processing. As we are working with *big data* here (i.e. > 300 million posts), we made two efforts to parallelise the induction of adoption probabilities over the test set entities. First, all of the data used for the experiments (timestamped interactions between users, entity posts, post details, E_{v2u} values, τ_{v2u} values) was uploaded into HBase[12] tables ensuring that we could *quickly* access the data using time-specific queries. Second, we used Apache Spark[13] to

[12] `https://hbase.apache.org/`
[13] `http://spark.apache.org/`

parallelise the per-entity diffusion processes. This was performed by loading the names of the test entities into HDFS and then forcing Spark to partition the entity list into at least 30 partitions. Each partition was then iterated over and the above test process run: (i) retrieving time-ordered entity posts from HBase, (ii) iterating over the post set, (iii) retrieving per-user interactions prior to the time of a given post, and (iv) calculating the pairwise influence probabilities. The final calculated probability of adoption for each user (u) together with the label of whether they adopted the entity or not were recorded in a separate HBase table of results.

Unfortunately, due to the use of a sample of 100 of the top-500 entities in our experiments, iteration over the time-ordered post set required an expensive sequential scan – which cannot be avoided. That said, we were able to add a second level of parallelism however, given the sub modular and monotonic nature of the joint probability as follows. Calculation of the probability of e being adopted by u is derived from Eq. 4, and is calculated from the prior neighbours of u before adoption. Now, as this function is sub-modular and monotonic, we could *update* the probability of adoption given a new neighbour's (v) influence probability as follows:

$$p_s(\Gamma(u) \cup p_{v,u}) = p_s(\Gamma(u)) + (1 - -p_s(\Gamma(u))) * p_{v,u} \tag{9}$$

Our additional layer of parallelism was achieved by *multithreading* the calculation of the influence pdrobabilities between v and each of his neighbours $u \in \Gamma(v)$, thus we calculated these pairwise influence probabilities in parallel and then updated $p_u(\Gamma(u)) : \forall u \in \Gamma(v)$.[14] The Java implementation of this code can be found in the github repository,[15] including the functions for building the HBase tables, deriving the entity-propagation counts (E_{v2u}) and the test algorithm.

4.4.2. Results

Table 2 presents the results from our experiments of the various entity-adoption constructs and probability settings including the micro-averaged *ROC* values for each model and the macro-averaged *ROC* values – together with their standard deviation – for each model. The former values are computed by *pooling* together all result tuples (i.e. $< u, p_u, r_u >$) from all the test entities, and working out the *ROC* value from that pool; while the latter values are computed by working out the entity-specific *ROC* values and deriving the average (and standard deviation) from those. Overall the results indicate that the static Bernoulli probability achieves the best performance, and that the best performing models are the Interactions-based model (from Micro-*ROC*) and the Community-homophily model (from Macro-*ROC*). We note the following salient points:

[14] N.b. the maintenance of the interactions between users records both the source and target of the interaction, hence we can retrieve directed interactions both ways – i.e. $v \rightarrow u \wedge v \leftarrow u$.

[15] `https://github.com/mattroweshow/NER-Diff-Paper`

Table 2: Area under the Receiver Operator Characteristic Curve (ROC) values for the different probability settings and influence probability settings within the general threshold model.

Entity-adoption Construct	Probability Setting			
	Static		Discrete-Time	
	Micro-*ROC*	Macro-*ROC*	Micro-*ROC*	Macro-*ROC*
Entity-propagations (p_u^E)	0.730	0.713($\pm$0.095)	0.730	0.714($\pm$0.096)
Interactions-based (p_u^I)	0.755	0.710($\pm$0.095)	0.666	0.644($\pm$0.091)
Community-homophily (p_u^C)	0.715	0.740($\pm$0.147)	0.643	0.631($\pm$0.085)

- The window of influence (characterised by $\tau_{v,u}$) is too narrow, as emphasised by Fig. 6a. The use of static probabilities, although capturing influence over a large time-period, actually contribute information towards understanding influence between users based on interactions and similarity of communities posted within, as a result the window omits interactions prior to this. This effect is represented in the difference in performance (of both Micro-*ROC* and Macro-*ROC* values) for the interactions-based and community-homophily models.

- Interactions have the greatest effect on adoption, not prior entity adoptions. The static Bernoulli model indicates that *in general* adoption of an entity is influenced by the intensity of interactions between two individuals, and not necessarily just whether propagation has actually occurred. This finding reflects the *communal* nature of Reddit where users constantly follow-up to posts with comments, which then evolve into a threaded discussion. It is likely that in this context that interactions around specific topics (within designated subreddits) occur frequently between clusters of users, thereby leading towards discussions around certain entities later.

- Adoption from community homophily varies between entities. The (relatively) large standard deviation for the community homophily model with static Bernoulli setting in Table 2 indicates how varied community-homophily can be. One could hypothesise here that entities which are specific to a given community and/or are emergent within a community would require a user to be *similar* to his peers in order to adopt it from them; whereas general entities are more likely to be ignored.

5. Discussion and Future Work

In this paper we have presented one of the first pieces of work to examine how entities spread through social networks. As a result of this novelty, our work has prompted a variety of avenues for future work. Therefore in this section we reflect on the approach we adopted and any potential issues that may arise from this, before then outlining our future work plans. One of the

core findings that we presented in this work is that the mode of the exposure-adoption function (i.e. probability of adoption as a function of k exposures to an entity) resides at $k = 0$. As we had to restrict the annotation of Reddit to a sample of 100 subreddits, it is possible that users were *exposed* to entities beforehand but within communities that we did not annotate. Therefore to validate our finding, our future work will take a sample of $1,000$ entities and form regular expressions that match those entities within the whole of the Reddit dataset, the exposure-adoption graphs will then be derived once again from this information.

Our second proposal for future work is to extend the univariate deterministic case that we have explored thus far – i.e. calculating the probability of u adopting e – to a multivariate case – i.e. calculating the probabilities of u adopting members from entity set E, where members of E are *colinear*. This would allow for the investigation of *vulnerability* windows to be explored which would characterise how *susceptible* a given user is to adopting an entity (or any colinear contagion) based on his recent adoptions. The third future work effort will be to extend the calculation of the adoption probabilities to the continuous time case – as in [27] – by computing the probability of one user *influencing* another user based on the latency to their latest interaction. This would allow for entity adoption to be explored from the perspective of pairwise interactions, as opposed to entity-propagations – thereby potentially alleviating the confounding effect of the influence window that we found in the discrete-time setting.

6. Conclusions

Understanding how entities spread through social networks provides researchers and marketers with valuable insights to recover and forecast the diffusion process. Our study of entity diffusion began by presenting an accurate means to obtain named entities from within discussion posts, before moving on to examining what *patterns* of entity diffusion occur – and how frequently – and how exposures to entities are associated with the probability of a user adopting an entity. Following these findings we presented a general threshold diffusion model that allows for different entity-adoption constructs to be tested within the diffusion process: our results from applying this model indicated that the interactions between individuals provide the most accurate means of calculating influence probabilities and thus forecasting entity adoption.

In the introduction of this paper we set forth three research questions. We now revisit these questions and highlight the evidence presented in our paper and how this has contributed towards answering the questions:

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.)? We have presented a method to detect named entities within Reddit posts that uses structured prediction and Brown clustering. Furthermore, we presented an empirical evaluation of our method when trained using a blend of named entity annotated corpora to transfer existing annotations from disparate corpora (covering different language

styles) as training data. We found that having a volume of in-genre unsupervised data could be used to much greater effect than approximations from blended corpora.

RQ2: What process governs the spread of entities? And how does such spread occur? We derived key insights into what diffusion patterns are found when entities spread through threaded discussions, finding that, unlike the spread of hyperlinks in the blogosphere [41], entities exhibit relatively *deep* and *narrow* diffusion traces. We also investigated the association between the number of exposures that users receive of an entity and the probability of a user adopting said entity thereafter, discovering that adoption probability decays as exposure count increases.

RQ3: How can we predict the spread of named entities and who will begin talking about them? Putting all the pieces together, we implemented a modified version of a general threshold model which incorporated entity-adoption constructs to test different mechanisms for computing user-to-user influence probabilities and can be learnt in parallel. Our empirical evaluation of this framework found that interactions had the greatest *overall* effect, while there was variance between entities in terms of the impact of *community-homophily* on users adopting an entity.

Acknowledgments

The authors acknowledge European Commission support from 7th Framework Program through grant No. 611223, PHEME, from Horizon 2020 through grant No. 687847, Comrades, and from NERC through grant ref. NE/L010100/1 Analysis of historic drought and water scarcity in the UK: a systems-based study of drivers, impacts.

References

[1] M. Duggan and A. Smith, "6% of online adults are Reddit users," *Pew Internet & American Life Project*, vol. 3, 2013.

[2] A. Goyal, F. Bonchi, and L. V. Lakshmanan, "Learning influence probabilities in social networks," in *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 2010, pp. 241–250.

[3] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Journal of Linguisticae Investigationes*, vol. 30, no. 1, pp. 1–20, 2007.

[4] L. Ratinov and D. Roth, "Design challenges and misconceptions in named entity recognition," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, 2009, pp. 147–155.

[5] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition," in *Proceedings of the seventh Conference on Natural Language Learning*. ACL, 2003, pp. 142–147.

[6] L. Derczynski, D. Maynard, G. Rizzo, M. van Erp, G. Gorrell, R. Troncy, J. Petrak, and K. Bontcheva, "Analysis of named entity recognition and linking for tweets," *Information Processing & Management*, vol. 51, no. 2, pp. 32–49, 2015.

[7] D. Hovy, A. Johannsen, and A. Søgaard, "User review sites as a resource for large-scale sociolinguistic studies," in *Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 452–461.

[8] B. Plank and D. Hovy, "Personality traits on Twitter–or–How to get 1,500 personality tests in a week," in *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2015, pp. 92–98.

[9] D. Preoţiuc-Pietro, S. Volkova, V. Lampos, Y. Bachrach, and N. Aletras, "Studying user income through language, behaviour and affect in social media," *PloS one*, vol. 10, no. 9, p. e0138717, 2015.

[10] A. Ritter, S. Clark, O. Etzioni *et al.*, "Named entity recognition in tweets: an experimental study," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. ACL, 2011, pp. 1524–1534.

[11] X. Liu, S. Zhang, F. Wei, and M. Zhou, "Recognizing Named Entities in Tweets," in *Proceedings of the HLT-ACL*, Y. Matsumoto and R. Mihalcea, Eds. Portland, Oregon, USA: Association for Computational Linguistics, 2011, pp. 359–367.

[12] B. Plank, D. Hovy, R. McDonald, and A. Søgaard, "Adapting taggers to Twitter with not-so-distant supervision," in *Proceedings of COLING: Technical Papers*, 2014, pp. 1783–1792.

[13] M. Rowe, M. Stankovic, and A.-S. Dadzie, "#Microposts2015 – 5th Workshop on 'Making Sense of Microposts': Big things come in small packages," in *Proceedings of the 24th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 2015, pp. 1551–1552.

[14] T. Baldwin, Y.-B. Kim, M. C. de Marneffe, A. Ritter, B. Han, and W. Xu, "Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition," *Proc. W-NUT (ACL-IJCNLP)*, pp. 126–135, 2015.

[15] C. Cherry and H. Guo, "The unreasonable effectiveness of word representations for Twitter named entity recognition," in *Proceedings of NAACL*, 2015, pp. 735–745.

[16] B. Han and T. Baldwin, "Lexical normalisation of short text messages: Makn sens a #twitter," in *Proceedings of the ACL-HLT*, Y. Matsumoto and R. Mihalcea, Eds. Portland, Oregon, USA: Association for Computational Linguistics, June 2011, pp. 368–378.

[17] H. Fromreide, D. Hovy, and A. Søgaard, "Crowdsourcing and annotating NER for Twitter #drift," in *Proceedings of LREC*. European Language Resources Association, 2014, pp. 2544–2547.

[18] C. Suen, S. Huang, C. Eksombatchai, R. Sosic, and J. Leskovec, "Nifty: A system for large scale information flow tracking and clustering," in *Proceedings of the 22nd international conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2013, pp. 1237–1248.

[19] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information diffusion in online social networks: A survey," *ACM SIGMOD Record*, vol. 42, no. 2, pp. 17–28, 2013.

[20] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information diffusion," in *Proceedings of the 21st international conference on World Wide Web*. ACM, 2012, pp. 519–528.

[21] D. M. Romero, B. Meeder, and J. Kleinberg, "Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter," in *Proceedings of the 20th international conference on World wide web*. ACM, 2011, pp. 695–704.

[22] L. Yang, T. Sun, M. Zhang, and Q. Mei, "We know what@ you# tag: does the dual role affect hashtag adoption?" in *Proceedings of the 21st international conference on World Wide Web*. ACM, 2012, pp. 261–270.

[23] P. Singer, F. Flöck, C. Meinhart, E. Zeitfogel, and M. Strohmaier, "Evolution of Reddit: from the front page of the internet to a self-referential community?" in *Proceedings of the companion volume of the 23rd International World Wide Web Conference, Web Science track*, 2014, pp. 517–522.

[24] T. Weninger, T. J. Johnston, and M. Glenski, "Random voting effects in social-digital spaces: A case study of Reddit post submissions," in *Proceedings of the 26th ACM Conference on Hypertext & Social Media*. ACM, 2015, pp. 293–297.

[25] R. S. Olson and Z. P. Neal, "Navigating the massive world of Reddit: Using backbone networks to map user interests in social media," *PeerJ Computer Science*, vol. 1, p. e4, 2015.

[26] X. Fang, P. J.-H. Hu, Z. Li, and W. Tsai, "Predicting adoption probabilities in social networks," *Information Systems Research*, vol. 24, no. 1, pp. 128–145, 2013.

[27] J. Huang, C. Li, W.-Q. Wang, H.-W. Shen, G. Li, and X.-Q. Cheng, "Temporal scaling in information propagation," *Scientific reports*, vol. 4, 2014.

[28] J. Baumgartner, "Complete public Reddit comments corpus," https://archive.org/details/2015_reddit_comments_corpus, 2015.

[29] D. Kergl, R. Roedler, and S. Seeber, "On the endogenesis of twitter's spritzer and gardenhose sample streams," in *Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on*. IEEE, 2014, pp. 357–364.

[30] T. Kiss and J. Strunk, "Unsupervised multilingual sentence boundary detection," *Computational Linguistics*, vol. 32, no. 4, pp. 485–525, 2006.

[31] B. O'Connor, M. Krieger, and D. Ahn, "TweetMotif: Exploratory Search and Topic Summarization for Twitter," in *Proc. ICWSM*. AAAI, 2010, pp. 384–385.

[32] P. F. Brown, P. V. Desouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai, "Class-based n-gram models of natural language," *Computational linguistics*, vol. 18, no. 4, pp. 467–479, 1992.

[33] J. Turian, L. Ratinov, Y. Bengio, and D. Roth, "A preliminary evaluation of word representations for named-entity recognition," in *NIPS Workshop on Grammar Induction, Representation of Language and Language Learning*, 2009, pp. 1–8.

[34] L. Derczynski and S. Chester, "Generalised Brown Clustering and Roll-Up Feature Generation," in *Proceedings of the 30th conference of the Association for Advancement of Artificial Intelligence*, 2016.

[35] L. Derczynski, I. Augenstein, and K. Bontcheva, "USFD: Twitter NER with Drift Compensation and Linked Data," *Proc. W-NUT (ACL-IJCNLP)*, pp. 48–53, 2015.

[36] M. Leginus, L. Derczynski, and P. Dolog, "Enhanced information access to social streams through word clouds with entity grouping," in *Proceedings of the conference on Web Information Systems and Technologies (WEBIST)*, 2015.

[37] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. ACM, 2008, pp. 1247–1250.

[38] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by gibbs sampling," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2005, pp. 363–370.

[39] L. Derczynski, S. Chester, and K. S. Bøgh, "Tune Your Brown Clustering, Please," in *Proceedings of the conference on Recent Advances in Natural Language Processing (RANLP)*, 2015, pp. 110–117.

[40] S. Chester and L. Derczynski, "generalised-brown: Source code for AAAI 2016 paper," Nov. 2015. [Online]. Available: http://dx.doi.org/10.5281/zenodo.33758

[41] J. Leskovec, M. McGlohon, C. Faloutsos, N. S. Glance, and M. Hurst, "Patterns of cascading behavior in large blog graphs." in *SDM*, vol. 7. SIAM, 2007, pp. 551–556.

[42] L. P. Cordella, P. Foggia, C. Sansone, and M. Vento, "An improved algorithm for matching large graphs," in *3rd IAPR-TC15 workshop on graph-based representations in pattern recognition*. Citeseer, 2001, pp. 149–159.

Vitae

Matthew Rowe Matthew Rowe is a Lecturer in Social Computing at Lancaster University. His work focuses on computational models of user and community behaviour on social media, with applications spanning information diffusion, recommender systems, and churn prediction. Dr. Rowe's work has been used in enterprise communities and by web-based systems to predict user behaviour, customer retention, and customer dissatisfaction. He currently runs a small social computing team comprised of three researchers within the School of Computing and Communications at Lancaster University.

Leon Derczynski Leon Derczynski is a Research Associate in the Natural Language Processing group at the University of Sheffield. His work focuses on social media information extraction and spatio-temporal information extraction. Tools that Dr. Derczynski has worked on are in daily use at many large enterprises, including Intel and CSIRO, and he frequently serves in an advisory role on social media, natural language processing and information retrieval.