Profession-Based Person Search in Microblogs: Using Seed Sets to Find Journalists

Mossaab Bagdouri
Department of Computer Science
University of Maryland
College Park, MD, USA
mossaab@umd.edu

Douglas W. Oard
iSchool and UMIACS
University of Maryland
College Park, MD, USA
oard@umd.edu

ABSTRACT

We introduce the problem of searching for professionals in microblogging platforms. We describe a study of how a group of professional journalists with some common characteristics (e.g., works in a specific language, belongs to certain region, or specializes in a particular media) can be found. Starting from seed sets of different sizes, social network features and profile content features are used to find additional journalists. The results show that combining the social network features of the reciprocated mentions and a bidirectional friend/follower graph provides a signal stronger than either of them taken independently, that both social network and profile content features are useful, and that profile content features are able to find larger numbers of less prominent journalists. We apply our methods to find the Twitter accounts of British and Arab journalists.

Categories and Subject Descriptors:
H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

Keywords: Microblogs; Person Search; Journalists

1. INTRODUCTION

The variety in social media services induces a diversity of applications, ranging from personal communication and entertainment to professional networking and collaboration. Some of these services, such as LinkedIn, are, by design, more business-oriented than others. They are, perhaps, the most obvious places to search for some particular experts. Certain professionals, however, need regular communication with their public, and might opt for more engagement in other popular platforms such as Facebook or Twitter. We introduce the problem of finding journalists on Twitter, one type of profession-based person search.

One use of microblogging is to disseminate breaking news [17, 28]. The wide adoption of these social technologies has also enabled some news gathering, filtering and dissemination activity to be led by nonprofessionals who have come known as "citizen journalists" [5]. The mainstream media has responded by encouraging their audience to interact through some predefined hashtags, by maintaining a Twitter account to keep their followers up to date, by creating new positions such as social media editors [33], and by leading crowdsourcing efforts [10]. The impact of these changes on the professional activities of journalists in the United States has been the subject of some recent qualitative studies such as Parmelee et al. [26], and our ultimate interest is in conducting studies of this sort that focus on the Arab world, which is in the midst of some dramatic changes.

One prerequisite for such studies is the ability to identify a sufficiently large and representative set of members of the population to be studied. While such lists can sometimes be constructed from the membership of professional associations, that approach is more useful in some places than others. In the case of journalists who cover the Arab world, a group in which we are interested, only relatively small and incomplete lists can be found. Because we are interested in how these journalists use Twitter, it is natural to look to Twitter itself as a way to finding.

Our focus is therefore to design, build and evaluate an automated system that can find a large set of authentic Twitter accounts belonging to a group of journalists with specific characteristics. After surveying the state of the art of related work in Section 2, we present in Section 3 the input and the search space of our systems. In Section 4 we then describe two families of methods for finding fairly homogeneous groups of journalists, one based on social network features, and another based on profile content features. We first assess the potential of our approach using an existing fairly comprehensive list of British radio journalists (Section 5) before tackling the more challenging problem of finding Arab journalists, for which we must create manual annotations as a basis for evaluation (Sections 6 and 7). We conclude with a few remarks about future work in Section 8.

2. RELATED WORK

Our task is an instance of person search, which can be seen as a special case of the broader problem of prediction of a demographic characteristic of social media users. Our methods are an instance of partially supervised learning. In this section we review related work on those topics.

2.1 Person Search

Much of the work on person search has focused on finding people with expertise on some topic, a variant of the person search problem referred to as expert search. For example,

CIKM'15, October 19–23, 2015, Melbourne, VIC, Australia.

ACM 978-1-4503-3794-6/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2806416.2806466 .

Han et al. built an interactive system that searched for academic experts, finding that modeling the degree of the social connection offered useful evidence, even when relevance and authority had already been modeled [16]. Their experiments were limited to searching for few persons and their collection (a crawl of ACM papers) was not subject to the noise that characterizes online social media, but we leverage this insight that social connection features can be informative.

Focusing more specifically on Twitter, Ghosh et al. designed a method relying on the feature of Twitter lists that allow users to organize some profiles of interest into lists of common themes or topics [16, 14]. They mined the titles and descriptions of these lists to infer expertise on several topics among many millions of Twitter users. In follow-up work, they used the same method to find groups of people that include experts and seekers of information about some topic [4]. They note, however, that reliance on user-generated lists from non-authoritative sources may be vulnerable to manipulation by spammers [16]. For this reason, we begin with what we judge to be authoritative lists, thus limiting our approach to the use of at most a small number of lists.

Topic-based person search naturally tends to find both experts and interested parties, whereas for our task we seek to find journalists rather than others. For example, a political topic might find both politicians and political journalists, and a journalism topic might cluster together journalists with news organizations and with researchers who study journalism. Cheng et al. [7] sought to mitigate a similar problem by also using location features in order to balance between topical and local authority. Such features are promising, but only a small portion of the Twitter stream is reliably geocoded, and our work on location inference in Arabic is not yet at a stage where we can make broader use of such features. We therefore look instead to the literature on predicting user demographics for further inspiration, since we can view profession as a demographic characteristic.

2.2 Predicting User Demographics

A variety of recent work has tackled the task of predicting user demographics in microblogging platforms, of which profession classification is a special case. Rao et al. [30] introduced the problem of user classification in Twitter. They defined four tasks to classify hundreds of users from four balanced datasets based on their gender [12], age [25], regional origin [18] and political orientation [8], using different social and linguistic features for each task. Bergsma et al. [2] clustered names and locations using connections created by user mentions, finding improvements in the prediction of geolocation [15], language [3], gender, ethnicity and race of users [23]. Pennacchiotti et al. [27] demonstrated that improvements can be gained by incorporating additional features such as profile content, and statistics about followers, friends, tweeting rate, hashtags and URLs. Our work is, to the best of our knowledge, the first to consider detecting members of any specific profession among microbloggers as its focal prediction task. We make use of the insight of Pennacchiotti et al. that profile text can productively be used together with friend and follower features, and Bergsma et al.'s insight that mentions also yield a useful feature set.

2.3 Partially Supervised Learning

Finding documents similar to a given set of known positive documents is a text classification problem known as *PU classification* (Positive/Unlabeled) or *partially supervised learning* [20]. Unlike traditional text classification tasks in which two sets, ideally of comparable sizes, are given to a learner as positive and negative labels, partially supervised learning starts with a small set of positive instances and a larger unlabeled set. The challenge becomes then to find a subset of "good" (i.e., useful) negative examples within the unlabeled documents to be fed to the learner.

A wide range of approaches have been suggested for this problem. Qiu et al. [29] focused on finding the initial set of positive documents, starting from a keyword and then mining labels from Wikipedia hyperlinks. We avoid this challenge by starting with a manually generated seed list. Fung et al. [13] built two unigram models for a positive and an unlabeled set that they then used to generate a ranked list of *core vocabulary* associated with the positive documents, before extracting a set of *reliable negative examples*. Sadamitsu et al. [31] proposed a method to expand a set of entities using topic models. Mordelet and Vert [24] suggested building several binary classifiers with the same known positive documents and different random samples from the unlabeled set as artificial negative instances. Bagging was then used to aggregate over the trained models. Our approach to using profile content, described in Section 4.2 most closely resembles the focus of Fung et al. on leveraging core vocabulary, although our approach to normalization differs somewhat, and we need only positive examples.

A technique that is close in spirit to the way we perform PU classification is Pseudo-Relevance Feedback (PRF), in which the basic approach is to enrich a one-sided query representation and then to rank the content to be searched without any reliance on negative examples [9]. The usual approach is to find some highly ranked documents using an initial query, and then to extract some terms from those documents and add them to the original query. Miyanishi et al. [22] proposed a two-stage PRF method for Twitter that consists of selecting a single tweet from the original ranked list, followed by a temporal query expansion relevance model. Wang et al. [21] used a related approach, expanding a query targeting short texts by issuing the query to a commercial search engine and exploiting the returned set to gather more terms. A rather different approach, similar in spirit, is based on the observation that URLs are sometimes the core information in the text of a tweet. Rather than expanding based on topical similarity, extends traditional PRF with expansion based on embedded hyperlinks [11]. While these techniques are widely used when improving search is the goal, we are not aware of any Twitter research using PRF in which searching for specific entity types (as with our interest in journalists) has been the goal.

3. SEED AND CANDIDATE JOURNALISTS

While some professionals such as physicians, engineers and lawyers acquire their titles after graduating from an appropriate school, some journalists become so by practicing journalism, rather than by taking classes. This can make it difficult to even assess whether somebody is a journalist. Wikipedia[1] defines a journalist as "a person who collects, writes or distributes news or other current information." In practice, it would be difficult to identify journalists based only on this definition. For example, while an act as simple

[1] `http://en.wikipedia.org/wiki/Journalist`

Figure 1: The self-identified journalist @youssefkuw shares an article of @fdalqabandi who lakes a profile description.

as retweeting a story is a way of distributing it, it is hard to believe that merely retweeting news stories would make one a journalist. On the other hand, some non-professionals play such an active role in collecting news that they consider themselves to be "citizen journalists." Instead of relying solely on any normative a priori definition, we therefore decided to adopt a descriptive approach, starting with a set of people who are authoritatively asserted to be journalists. This approach yields two benefits. First, we begin to develop and refine annotation guidelines that span a diverse range of members of the profession. Second, we can adopt this set as a "seed" from which we can find additional journalists.

Our "seed journalists" are some set of journalists who share some common characteristics that define the population we ultimately wish (in future work) to study (e.g., they work in a specific language, are from certain region, or specialize in a particular media). When such a seed set is not readily available, we can build one at some annotation cost, as we show in Section 6. Our goal in this paper is then to apply an automated method for finding additional journalists with similar characteristics.

To do this, we first need to define some search space. We do this by assuming that many journalists with similar characteristics will share some types of social connections. We thus consider two social graphs that are built from our seed journalists. The first graph relies on the network of followers (i.e., accounts that follow a seed journalist) and friends (i.e., accounts that the journalist follows). We observe that some journalists are very well known, with millions of followers. The density of journalists within such a large set of followers would be expected to be rather low, however. Similarly, journalists might be expected to follow many types of accounts that are not necessarily journalists (e.g. @BarakObama). With a goal of high precision in mind, we therefore restrict our friend/follower graph to bidirectional relationships. We collect this graph by querying Twitter API for both friends and followers of each account in the seed set, then intersecting these friends and followers.

The second graph we consider is constructed from mentions. A user can draw the attention of another user by including her screen name preceded by the '@' sign. We therefore also build a mention graph by connecting edges to accounts having reciprocated mentions with any of our seed journalists. Gathering this set is not as trivial as with the friend/follower graph because the Twitter API lacks a feature that allows retrieving the list of users who have ever mentioned a given account. We therefore begin by using the API to crawl all of the available tweets from the "timeline" (i.e., the sequential listing of tweets) of each of the seed journalists. The API returns up to 3,200 of the most recent Tweets.[2] We then similarly crawl all of the available tweets of accounts that have been mentioned in any tweet of any seed journalist. We search this collection to find any mentions of any seed journalist. The 3,200-tweet API limit means that we may miss some mentions by prolific accounts.

4. METHODS

We present three approaches for finding journalists based on one or both social graphs, and one based on profile text.

4.1 Social Graph

We use identical methods to rank candidates using the friend/follower and the mention graphs, and then we combine the two resulting rankings to produce an integrated ranking based on both sources of evidence.

4.1.1 *Follows and Mentions*

Kang and Lerman [19] demonstrated the effect of homophily in Twitter, which is the tendency of individuals to connect to similar others. We illustrate this observation for the mentions graph in Figure 1. The a-priori known journalist *@youssefkuw* mentions his colleague *@fdalqabandi*, whose profile does not indiate any occupation, while sharing his article. The latter mentions him back in a thank-you note. Reciprocal connections can be noisy, as some users tend to follow/mention back each incoming follow/mention. We can reduce this noise by enlarging the number of distinct journalists a candidate user is connected to. We therefore want to rank candidate accounts by their connectivity to the seed journalists.

Formally, let G be a set of vertices partitioned into S and V, corresponding to the sets of seed and candidate vertices respectively. E is the adjacency matrix of V and G. That is, for two vertices v and g in V and G respectively, $e(v,g) \in E$ has a value of 1 if a connection exists between v and g, and 0 otherwise. For a given vertex v in V, and the set of seed vertices S, we define a connectivity measure as a function of all the weights of the edges connecting v to S. In this paper we use two connectivity measures:

- Raw count, which is an integer between 0 and $|S|$: $c(v,S) = \sum_{s \in S} e(v,s)$
- Relative density, a rational number between 0 and 1: $d(v,S) = (\sum_{s \in S} e(v,s))/(\sum_{g \in G} e(v,g))$

We sort the candidate set in a decreasing order by raw count, breaking ties by sorting by decreasing relative density.

We apply this to the follows and mentions graphs independently. We denote by S-Follows and S-Mentions the associated approaches, respectively. We leave the study of other graphs such as hashtags, retweets and replies to future work.

4.1.2 *Intersection-Based Reranking*

Combining two ranked lists can yield a new ranking that is sometimes better than either. Algorithm 1 describes a

[2] `http://dev.twitter.com/rest/reference/get/statuses/user_timeline`

Algorithm 1 $Combine(L_0, L_1)$

input: L_0, L_1
output: L // Combined list
1: $L \leftarrow \{\}$
2: $l_0, l_1 \leftarrow \{\}$ // Elements seen in L_0 and L_1
3: **for** iteration i in 1 .. $Max(|L_0|, |L_1|)$ **do**
4: **for** j in $0, 1$ **do**
5: $e_j \leftarrow L_j.get(i)$
6: **if** $e_j \in l_{1-j}$ **then** $L.add(e_j)$
7: **else** $l_j.add(e_j)$
8: **if** $e_0, e_1 \in L$ **and** $L_1.index(e_0) < L_0.index(e_1)$ **then**
9: $L.swap(e_0, e_1)$

method that combines two lists ranked by the approaches S-Follows and S-Mentions introduced in the previous section. At any iteration i, we look at the i-th elements of the input lists L_0 and L_1. If either of these two elements has already been seen in another list, then we add it to the combined list L. Otherwise we hold it in a temporary list l_i for future lookup. If both of these two elements are to be added to L, then we sort them according to which of them was seen first. This algorithm is illustrated in Table 1.

4.2 PU Classification

We expect a language model built on top of text describing a set of homogeneous journalists to contain a signal that differentiates that set from the language model corresponding to a set of accounts in which these journalists constitute a minority. Based on this assumption, we can rank any given account based on its similarity to the language model of the set of known journalists.

Drawing from the work on text classification without negative examples [13], we denote by $DF(w_G)$ the document frequency of a word w_G from the set of Twitter accounts G, where a document is a text associated with a Twitter account. $DF(w_G)$ is, thus, the count of accounts in which w_G appears in the corresponding document. We then scale $DF(w_G)$ to a value between 0 and 1 as:

$$df(w_G) = \frac{DF(w_G) - \min_{w'_G \in G} DF(w'_G)}{\max_{w'_G \in G} DF(w'_G) - \min_{w'_G \in G} DF(w'_G)}$$

For a word w that appears in both the seed set S and the candidates set V, we denote by H the value $H(w) = df(w_S) - df(w_V)$. This value gives higher credit to words that are more frequent in the seed set than in the candidates set. In other words H defines a ranking of words by their relatedness to the documents associated with the journalist accounts. To avoid noise that could be generated by the words ranked at the bottom of this list, we define a threshold θ above which we truncate this list H into a sublist H':

$$\theta = |W_S|^{-1} \sum_{w \in S} H(w),$$

where $|W_S|$ is the count of unique words in the documents associated with the seed set S. Finally, each element h_w in H' has a rank r_w that we use to define the unnormalized positive referencing power p_w as:

$$p_w = exp(-r/|H'|)$$

For a given Twitter account g in G with a corresponding document d_g, we want to sum over all $p_{w \in d_g}$ to compute the similarity score. This sum, however, needs to be normalized with respect to the document length l_{d_g}, which is the count of unique words in d_g. If we normalize naively by dividing the sum over l_{d_g}, we risk giving a high rank to documents with a single noisy word (i.e., a word that undesirably has a high p_w value). Instead, we dampen the document length logarithmically. Finally, we score the document d_g based on the text features as:

$$t(d_g) = (\sum_{w \in d_g} p_w)/log(1 + l_{d_g})$$

Several document types can be associated with a Twitter account, such as the set of all hashtags and the concatenation of all or the most recent tweets. We limit our experiments in this paper to the description field appearing in the user profile. We denote by T-Desc this method that is based on the text features of the description field. We limit the set of journalists to be ranked by T-Desc to the candidates that are found by at least one of the social graphs with the same seed set, and that are not missing a profile description. In addition to the computational convenience, this restriction increases the precision by limiting the false positives. As an example, some of the Arab journalists that we focus on in Section 7 use some French and English words (e.g., producer) in their description in lieu of their Arabic counterparts (e.g., منتج). As a consequence, some of these foreign words would get a high rank in the core vocabulary. Had we not made this restriction, a higher number of English and French speaking journalists would thus likely be retrieved.

Table 1: Example of combining two ranked lists. At step i, there is no intersection between the two lists. At step ii, element F appears in List 1. We add it to the top of the combined list because we have already seen it in List 0. The next intersection involves element B at step iv. At step v, both element D and G are added to the combined list, but we start with G because it was seen first at step ii. List 1 is exhausted, but List 0 still has element H which has been seen at step iv. We append it to the combined list.

Step	List 0	List 1	Combined
i	F	B	-
ii	G	F	F
iii	A	D	-
iv	B	H	B
v	D	G	G, D
vi	H	-	H

5. FORMATIVE EVALUATION RESULTS

We begin our experiments by exploring the performance of our methods with a set of British radio journalists that is large enough to be divided into training and test sets. Our goal in this first set of experiments is to find out to what extent our algorithms can exploit a seed subset of these journalists to find others from the full set.

The website `Media.info` collects a list of contacts of media organizations and journalists from the United Kingdom (UK), Ireland (IE) and Gibraltar (GI). The profiles are of either a person (P) or an organization (O), and are distributed across radio stations, TV channels, newspapers and magazines. We consider only profiles that have at least one Twitter account. A profile can belong to more than one organization type and region (e.g., a journalist may have worked

Table 2: Distribution of Twitter accounts in media.info. P and O correspond respectively to accounts of persons (i.e., journalists) and news organizations.

	Magazine		Newspaper		**Radio**		Television	
	P	O	P	O	**P**	O	P	O
UK	83	311	314	290	**1529**	579	433	154
IE	4	22	112	40	222	83	19	11
GI	1	4	5	6	8	3	2	2

in a UK magazine before moving to an IE TV channel). We found that five Twitter accounts belong to both personal and organizational profiles. We consider these to be errors in the dataset and we exclude them. Table 2 shows the distribution of the remaining 4,137 Twitter accounts. For our experiments we consider positive examples to be Twitter accounts of the 1,529 British radio journalists (i.e, UK Radio P), including those who have also worked in other venues, and as negative examples the other 2,608 Twitter accounts within this dataset. Examples of such negative accounts include the Twitter accounts of a UK radio station, a UK newspaper editor, and a GI radio broadcaster. We chose that positive/negative split for two reasons. First, this task is more focused than that of finding any English speaking or even British journalists. If we succeed at our narrower task, we would expect to also succeed at more general tasks. Second, the number of British radio journalists is the largest subset identified in Table 2. This allows us to explore the impact of the broadest range of seed set sizes.

We want to evaluate our four algorithms using the labels of this dataset. We also want to keep track of the impact of the seed-set size on the performance of the systems. Two evaluation design issues need to be addressed. First, we need to choose a measure that relies only on known labels. Inspired by the work of Sakai [32], in which he defines $AveP'$ as the average precision on a ranked list after removing documents with unknown judgements, we define $P'@N$ as the precision at rank N on a list that has all of the unknown accounts removed. $P'@N$ is exactly $P@N$ when we know the labels of all of the first N accounts. The fewer unknown accounts we have, the closer $P'@N$ will likely be to $P@N$. We wish to keep $P'@N$ close to $P@N$, which is the measure we really care about. Thus, we avoid choosing a high value for N (even though we will want to retrieve hundreds of journalists in our ultimate application), as doing so would increase the number of unknown accounts excluded from the measurement. Hence, we use $P'@10$. Second, we need to measure on a stable test set if we are to compare results from different seed-set sizes. We therefore randomly draw a single held out test set and sweep across seed sets drawn from the remaining items.

As our test set we first randomly draw 500 accounts (without replacement) from the set of all 4,137 Twitter accounts. About 185 of these will typically be positive examples, leaving at most just over 1,300 positive examples for training. We then randomly order these remaining positive examples, and draw 27 nested samples with sizes between 5 and 1,300 as seed sets of British radio journalists. We then run our person search algorithms, obtaining one ranked list for each seed-set size from each algorithm. These ranked lists will typically find many Twitter accounts from outside the test set. Rather than judging those results manually, we evaluate each ranked list by going down the ranked list from the top

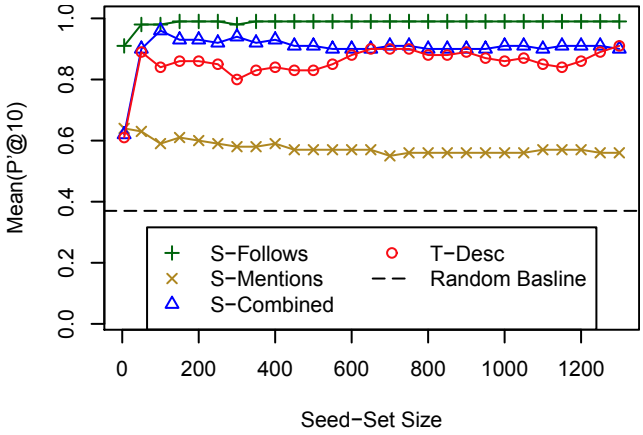

Figure 2: The mean of the precisions at 10, across ten random runs, for 27 seed-set sizes, using only the labels of accounts within test sets of size 500.

until 10 accounts that exist in the test set have been found, ignoring all other unknown accounts. If 10 known accounts are not found, we stop at the maximum number of accounts returned. We then compute the precision on just these first 10 known accounts. For cases in which fewer than 10 known positive examples are found anywhere in the ranked list, we still divide the number found by 10. This results in a single decile score (0.0, 0.1, 0.2, ..., 1.0) for each seed-set size and every algorithm. To obtain more fine-grained scores, we repeat the entire process—randomly drawing another test, randomly ordering the remaining positive examples, forming the 27 nested seed sets, running every algorithm for every seed set, and scoring the results—a total of ten times. We then average the resulting $P'@10$ values across the ten repetitions for each seed-set size and every algorithm.

Figure 2 depicts the results for each of our four person search algorithms. For reference, we plot a baseline $P'@10$ at 0.37, which is what would result from simply selecting 10 random accounts from the test set. We observe that S-Follows, S-Combined and T-Desc all yield fairly good results (mean $P'@10 \geq 0.8$) for seed-set sizes of at least 50. S-Mentions does least well, retrieving the largest number of detected false positives. To characterize the effect of ignoring unknown accounts, we can compute the mean of the ranks at which the first 10 known items in the test set were found. We take this mean across both the 10 items and the 10 repetitions. The lowest possible value for this mean is $avg(1..10) = 5.5$, which would be achieved if all of the top 10 accounts were in the test set. Taking a seed-set size of 200 as an example, the (rounded) average depths are 48, 49, 93 and 217 for S-follows, S-Combined, S-mentions, and T-Desc, respectively. We thus see that our three methods that rely on social features seem to behave more similarly than our one method (T-Desc) that relies on profile content features. Our results for Arabic in Section 7 help to explain this observation. To foreshadow that result, we believe that T-Desc is better able to find less prominent journalists than are our methods that are based on social features.

These formative evaluation results suggest that both social and profile content features can be useful. We therefore next evaluate our algorithms using a seed set of Arab journalists and newly created annotations.

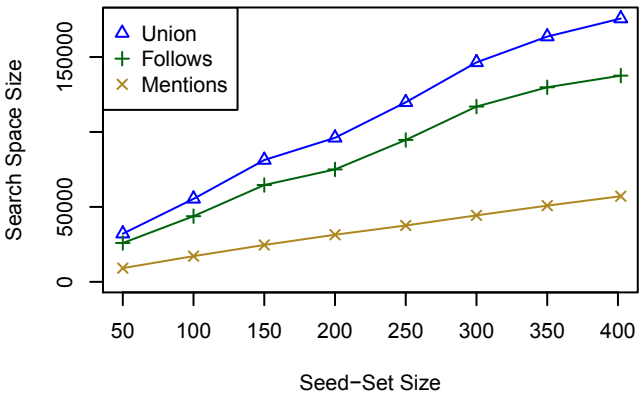

Figure 3: The average size of the search space as a function of the seed-set size. The limit of 3,200 tweets per account enforced by Twitter API makes the search space size of the mentions graph grow linearly but slowly. It grows at a higher speed for the follows graph, but with a diminishing return.

6. ARAB JOURNALISTS COLLECTION

Although Twitter accounts of many British radio journalists had been conveniently found, no such lists exist for Arab journalists. Twitter does allow users to construct lists of Twitter accounts, and several English news outlets do take advantage of this feature. For example, CNN maintains a list of 272 Twitter accounts, apparently of its own journalists[3] and Associated Press maintains a general list of 866 AP staff,[4] along with several more focused lists (e.g., of 73 AP photographers.[5]). Aljazeera Arabic, by contrast, offers no such list.[6] We therefore manually collected a seed set of Arab journalists, relying mainly on Wikipedia. After running our algorithms to find additional Arab journalists, we hired three annotators to evaluate our results.

6.1 Seed Journalists and Search Space

To build the seed set of Arab journalists, the first author looked at all names appearing under the category "Arab journalists" (including all its subcategories) in the Arabic version of Wikipedia.[7] We restricted our list to people who are still alive. The resulting set includes persons for whom Wikipedia indicates more than one title, such as writers, poets, politicians who are editing partisan newspapers, professors who are also newspaper columnists, news anchors, TV reporters, and retired journalists. Some pages contain direct links to a journalist's official website or Twitter account. If found, the first author visited the link to assess the authenticity of the account. Otherwise, two queries were issued to a commercial search engine that supports transliteration: (1) *name* site:twitter.com, and (2) *name* تويتر.[8] Because the name of a journalist might be shared with many other people, we sometimes needed to examine as many as 30 results, although often the candidate was found in the first 10.

[3] `http://twitter.com/CNN/lists/cnn-news`
[4] `http://twitter.com/AP/lists/ap-staff`
[5] `http://twitter.com/AP/lists/ap-photographers`
[6] `twitter.com/AlJazeera/lists/al-jazeera-arabic` contains only two organizational accounts.
[7] `http://ar.wikipedia.org/wiki/تصنيف:صحفيون_عرب`
[8] Twitter in Arabic

Table 3: List of labels available to the annotators to choose from. If the exact label was difficult to assess, a catch-all label for each category could be used (i.e., other).

Category	Labels
Journalist	Editor, photojournalist, magazine writer, news producer, broadcaster, correspondent, columnist, editor-in-chief, TV/radio host, other journalist
Cannot decide	Cannot verify (media person), cannot verify (other), account unavailable
Not a journalist	Writer, activist, oppositionist, poet, artist, spokesperson, blogger, politician, fan, bot, fake account, newspaper, station, channel, news org., news aggregator, other organization, other non-journalist

Table 4: Number of accounts assessed by both the first author, and each of the three independent annotators, with the corresponding Cohen's Kappa value.

	A1	A2	A3
Accounts /	317	418	20
Cohen's κ	0.78	0.75	0.78

To assess whether an account is actually associated with a specific journalist, rather than with a fan or with someone else with the same name, we relied on factors such as the name, the username, the profile content and picture, the number of tweets, the content of the most recent tweets (including links embedded in those tweets), the number of followers, a match between the age of the journalist as indicated by Wikipedia and by the profile, and an indication by Twitter that the account has been verified. No strict rules were set for any of these criteria. Instead, the first author made an individualized judgment in each case. Occasionally, in an ad-hoc way, he also considered some accounts that were followed by a journalist's account (e.g., their colleagues) as a way of expanding the list of journalists. The final result was a set of 402 Twitter accounts that we believe are quite clearly owned by Arab journalists. We use this set only for training; our evaluation data has been judged by independent annotators.

The union of the two bidirectional graphs of followers and mentions contains a set of 175,643 unique Twitter accounts. To study the effect of the seed-set size on the methods we describe below, we also define some nested subsets of our 402-journalist seed set. We do this by randomly sampling 50 journalists without replacement, then adding another 50, then another, and so on, up to 350. We repeat the graph construction process for each seed set, generating a subset of the full graph for each of the smaller seed sets. We perform this process three times, and we average our reported results over the three random sets. As an example of this process, Figure 3 shows how the average number of Twitter accounts in each graph grows with the seed set size.[9]

6.2 Annotations

To conduct our experiments we generated a pool of 1,441 annotations from the output of various systems and configurations (Section 4). We then hired three independent anno-

[9] These results are actually averaged over 10 points in each case because no annotation is required; results in Section 7 are averaged over three points to limit annotation costs.

tators (who are not aware of any details of our algorithms). We refer to these annotators as A1, A2 and A3; they assessed 710, 646 and 123 Twitter accounts, respectively. We asked them to judge whether an account is a journalist, and whether it is an Arab. To help them more consistently judge whether an account is a journalist, we gave them a list of labels, partitioned into three categories (journalist, cannot decide, not a journalist) that we allowed to grow incrementally based on their feedback (Table 3). For the results reported in this paper we consider as positive all accounts under the category journalist, and as negative all the other accounts. We asked the annotators to verify whether an account does actually belong to the corresponding person (by examining the profile and by running a Web search), and whether that person had ever practiced journalism (in a manner described by any of the provided titles). We emphasized that confusing labels within the same category was tolerable, but they were strongly encouraged to try their best not to confuse labels across different categories. It took, on average, three minutes to assess an account.

To compute inter-annotator agreement, the first author of this paper also annotated 723 accounts, doing so before examining the results of the independent annotators. Table 4 shows that there is high agreement with each annotator, with Cohen's chance-corrected Kappa values between 0.75 and 0.78. The first author's annotations were used only to compute inter-annotator agreement; the results in Section 7 are based solely on the independent annotations. In the few cases in which more than one independent annotator annotated the same account, we chose annotations from A1 or A3 over those of A2 for use as the gold standard.

7. EXPERIMENTS

We present our experiment design and discuss the results with respect to the number of journalists found, the diversity of retrieved accounts, and the trade-off between gathering a large seed set and annotating more retrieved accounts.

7.1 Baselines

We propose two baselines that a user who is trying to find Arab journalists might reasonably have tried. The first one is a keyword search: we issue a query through Twitter API to search for users using the word "journalist" in Arabic (صحفي). We name this method T-Baseline. The second method relies on Twitter's who-to-follow recommendations. For a given seed set, we create a Twitter account and configure its country to be Egypt, and its timezone to be that of Cairo (UTC +2 hours). We then set it to follow the accounts in the full 402-journalist seed set. To allow adequate time for any background processing performed by Twitter, we wait 24 hours before crawling the page of recommendations. We denote this method S-Baseline.

7.2 Performance

As described in Section 6, we sample nested subsets from the full seed set, repeating the process three times. We run our four systems on each subset and for the full set. We run S-Baseline only on the full set, and T-Baseline independently from any seed set. We then pool the top 50 accounts retrieved by each system and run, removing duplicates, ordering randomly, and partitioning arbitrarily among the three annotators. As our principal evaluation measure, we use precision at 50 (averaged, for the subsets, over the three samples) because annotations to depth 50 are sure from this process to be available. Figure 4 shows these results.

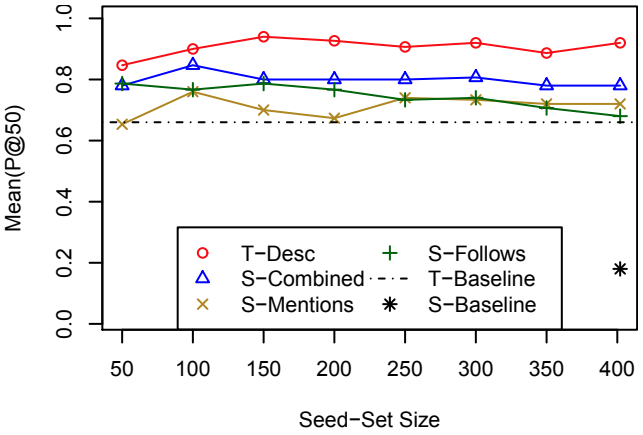

Figure 4: The mean, across three random runs, of the P@50 for our four methods (S-Follows, S-Mentions, S-Combined and T-Desc) on eight nested seed sets. The baselines are computed over one run, on seed-set size 402 for S-Baseline, and independently from the seed sets for T-Baseline.

We observe that all four methods are relatively insensitive to the seed set size, although S-Follows seems to be slowly decreasing as the seed-set size increases, perhaps because its search space is increasing the most quickly (Figure 3). No clear preference between S-Mentions and S-Follows is evident, but intersection-based reranking (S-Combined) does indeed seem to be doing better than either of the methods that it combines. The clear winner, however, is T-Desc, which achieves outstanding results (precision at 50 of 0.94 at a seed-set size of 150, with only 3/50 false positives). This method does indeed extract a core vocabulary that one would expect to be related to Arab journalists such as *writer*, *journalist*, *editor*, *Sky* and *Arabia*, in addition to terms that we did not anticipate such as *opinions* and *endorsement*, which are often used when journalists want to separate their personal opinions from their employers, or emphasize that retweets are not endorsements.

For reference, we plot the precision at 50 for each of the proposed baselines. The keyword based baseline (T-Baseline), which is independent of the seed set, finds 33 Arab journalists in the top 50. This is markedly worse than our text based method (T-Desc) which finds, on average, no less than 42 journalists (which is the average value for a seed set size of 50). This baseline is close to what S-Mentions and S-Follows find, but is outperformed by S-Combined. The Twitter who-to-follow baseline (S-Baseline) finds nine Arab journalists in the top eleven retrieved accounts, and then starts to recommend accounts of celebrities and organizations in the region where we ran our experiments. This suggests that the eleven recommendations are based on the network of friends, while the other ones are recommended given the IP address of our machine. We do not know whether Twitter could have recommended additional journalists, but even if it were to have done so at the same rate (9/11=0.81) down to rank 50 (which seems to us a bit optimistic), it still would have been beaten by T-Desc.

P@50 is a useful measure of how accurate a method is, but to characterize completeness we need to look deeper in the

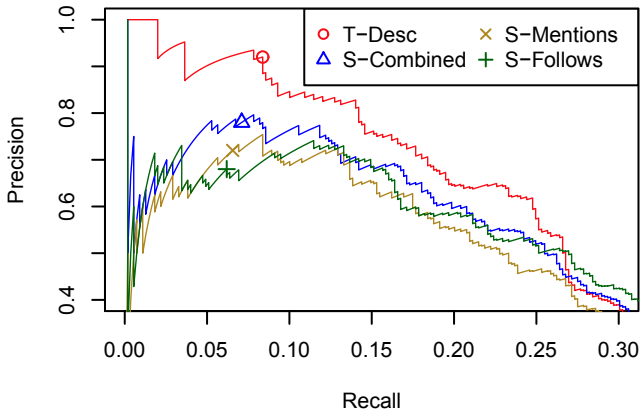

Figure 5: A recall / precision curve for the full seed set. We use all of the known annotations generated for three random runs, eight seed-set sizes and four systems. We assume all of the unassessed accounts to be negative.

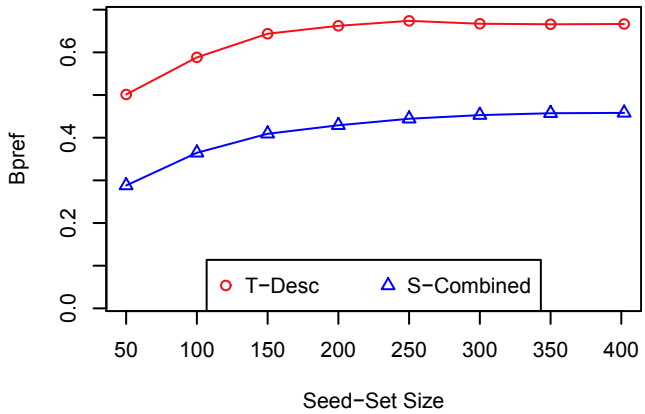

Figure 6: The mean of *bpref*, across three random runs, for eight seed-set sizes, using all of the annotations of the first 50 accounts, in addition to the annotations of seven accounts centered around the depths 50, 100, 250, 500, 750 and 1000.

ranked list. We can do this by computing a precision-recall plot. As is usual in such cases, we make the simplifying assumption that any unannotated account (i.e., any account that was never in the top 50 for any result set) is not a journalist. This tends to understate both precision and recall, but at relatively low computed recall levels (e.g, below 0.25) relative comparisons should remain reasonably accurate. In Figure 5, T-Desc is dominant over S-Combined over a large region, and there is almost no region in which S-Combined is dominated by either S-Mentions or S-Follows. For the remainder of this paper, we therefore focus on T-Desc and S-Combined.

Another way of looking deeper in the ranked list is to compute *bpref*, a ranked retrieval measure that is suitable for making system comparisons with incomplete judgments [6]. *bpref* is a ranking measure that scores a system based on its ability to rank known relevant items above the known irrelevant items; it is defined as:

$$bpref = \frac{1}{J}\sum_{j} 1 - \frac{|j \text{ ranked higher than } n|}{min(J, N)},$$

where j is an account retrieved from the set of known Arab journalists of size J, and n is retrieved from the set of accounts known to be not of Arab journalists (of size N).

On the positive side, *bpref* is able to consider the entirety of each ranked list; on the negative side its numerical value is not as easily interpreted as, for example, precision or recall. For computing *bpref* we also use the additional "point precision" annotations that we describe below in Section 7.4. Figure 6 illustrates the average *bpref* across three random runs for eight seed-set sizes, including annotations gathered at lower depths. Consistent with our other results, T-Desc does better at ranking journalists ahead of other accounts than S-Combined with any of the seed-set sizes that we tried, but they both tend to plateau at a seed-set size of about 250.

7.3 Retrieval Diversity

The results in Section 7.2 clearly indicate that between T-Desc and S-Combined, T-Desc is the better choice. This is, however, a false choice, since T-Desc and S-Combined find largely disjoint sets of journalists. To see this, we first look at the journalists retrieved by both methods based on the complete seed set (i.e., with 402 accounts). We find that T-Desc and S-Combined retrieve 46 and 39 journalists respectively within the first 50 accounts returned. Interestingly, only one journalist is returned by both of them. That is, out all of the top 50 journalists we can find with each method based on this seed set, only 1.19% are returned by both of these methods. Next, for each method independently, we look at all of the journalists retrieved using any of the eight seed sets and the three random runs. T-Desc finds a total of 261 journalists, while S-Combined finds 153. Only 12 journalists are retrieved by both. That is, less than 3%. Finally, we do not limit ourselves to the top 50 accounts, and we go as deep as we can (we cut off our analysis at 1000 returned accounts); for this analysis we also use the additional "point precision" annotations that we describe below in Section 7.4. We find that the intersection of the 660 and 541 journalists retrieved by T-Desc and S-Combined respectively is 260, 27.63% of the total.

Clearly, T-Desc and S-Combined are not finding the same journalists. This observation leads us to investigate the differences in the populations of journalists returned by these two methods. Figure 7 plots, on a log scale, some attribute values for the first 50 accounts returned using the full seed set. The filled circles and triangles correspond to the Arab journalists correctly retrieved by T-Desc or S-Combined, respectively (true positives), while the empty symbols correspond to the accounts retrieved that are not Arab journalists (false positives). We selected these attributes as indicators of the presence and activity of the accounts. As the top three plots for the Listed, Followers, and Favorites features show, journalists correctly found by S-Combined are more prominent than the ones returned by T-Desc.[10] Taking Followers as example, the median number of followers for journalists correctly detected by T-Desc is 1,328, while it is 53,540 for journalists correctly detected by S-combined.

These results suggest that different goals in the creation of lists of journalists may call for different methods. If the goal is to maximize the number of journalists, both techniques

[10]Listed counts the number of lists to which the account was added by other users; lists are a way of organizing tweets. Favorites counts how many times a journalist "favorites" another users' tweet as a way of calling attention to it.

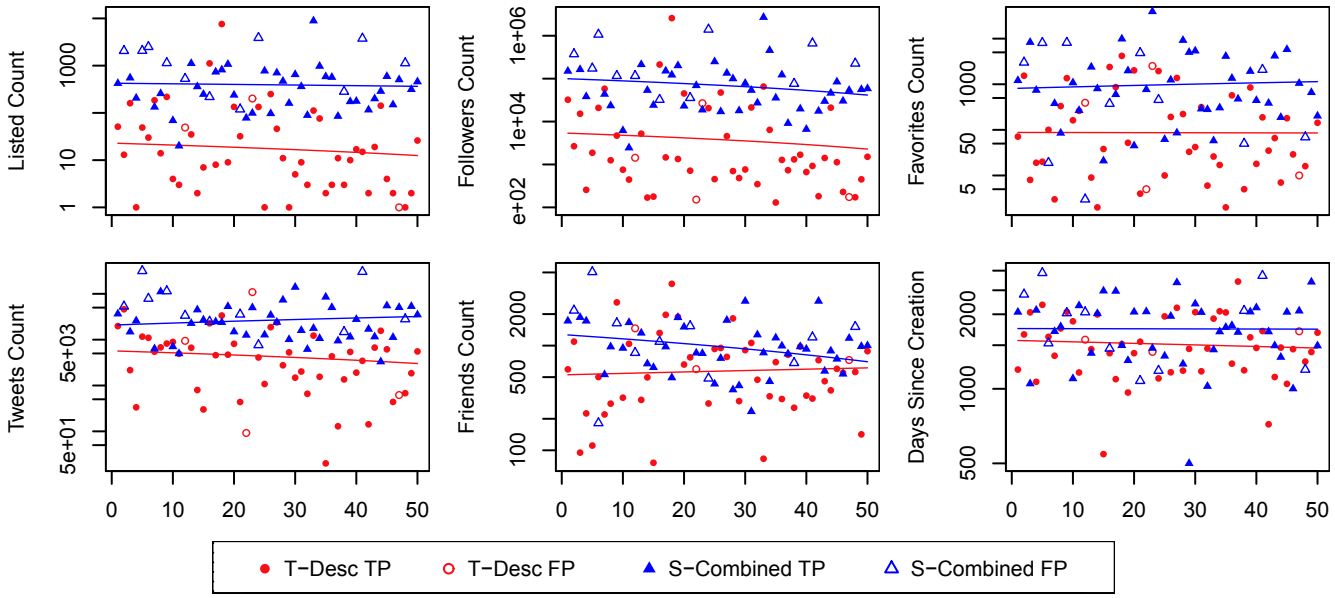

Figure 7: Numerical features of retrieved accounts (in a log scale) as a function of the retrieval rank (x-axis) and the ranking method. The retrieved accounts are either true or false journalists (TP and FP respectively). The fitting line are plotted after excluding the false journalists as well as the outliers (which are values outside the range $[q_1 - 1.5 \times IQR, q_3 + 1.5 \times IQR]$, where IQR is the difference between the first and the third quartiles q_1 and q_3 respectively).

should be used. If, however, the goal is to find journalists who are representative of a broad population, T-Desc alone may be the better choice because using S-Combined risks biasing the set in favor of prominent and prolific journalists. Of course, it is prominent journalists that we wish to study, then S-Combined would be an excellent choice.

7.4 Annotation Cost

We have so far focused on the performance of the various systems ignoring the cost of obtaining the initial seed set. Indeed, if this set was freely accessible and known a priori to be accurate, then focusing on the precision and yield (recall × cutoff) of the resulting systems would suffice. However, if this seed were to be expensive to gather and manually check, then we might be better off acquiring a smaller seed set and annotating more deeply in the ranked list. We now turn to the question of how best to balance annotation costs and the quality of the results. This is particularly useful when we want to find some minimum number of journalists, across the seed set and retrieved list. Here we model cost by annotation time using, for example, the average of three minutes per annotation that we reported above in Section 6.2.

We formalize this trade-off by estimating the number of retrieved journalists at any depth (down to 1000), as a function of the seed-set size. For each of a list of five seed-set sizes between 50 and 402, we compute the "point precision" (i.e., the precision near some point in the ranked list) at 50, 100, 250, 500, 750 and 1000. We do so by computing the ratio of accounts that are journalists within a radius of 3 of these specific points in the ranked list (i.e., count(journalists) / 7). The area under the curve defined by these depths and the corresponding point precisions approximates the number of retrieved journalists down to the depth 1000.

Figure 8 shows one way in which we can use these results, looking in this case at the estimated yield in the top 1000.

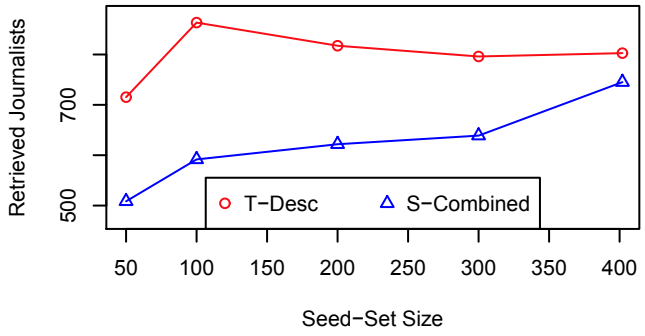

Figure 8: Number of retrieved journalists at depth 1000. For a given seed-set size, we estimate this number by computing the point precision at depths 50, 100, 250, 500, 750 and 1000, and calculating the area under the curve of the precision = f (depth). We then average over three random runs.

As with our earlier results for precision at 50, there is no evidence beyond seed set sizes of 100 that T-Desc does better with larger seed-set sizes. It appears that the core vocabulary words are learned adequately with even smaller seed-set sizes. On the other hand, S-combined does find more journalists as the seed set size increases, but with a diminishing return. Curves of this type, computed at different depths, can serve as a guide for balancing between training (i.e., acquiring more seed journalists) and testing (i.e., gathering more journalists by applying the ranking method) when the cost of training annotations is significant [1].

8. CONCLUSIONS AND FUTURE WORK

We have introduced the problem of using a seed set to find journalists in Twitter. We suggested two families of methods to solve this problem. The first relies on the bidi-

rectional social links of friends/followers or mentions which, when combined, tends to retrieve prominent journalists with high precision. The second, based on text features from the profile description, retrieves journalists that are less active on Twitter, with even higher precision. As we have discussed, the preference for one approach over the other, or for using both, depends on the characteristics of the set of journalists that we wish to construct. Considering the total annotation budget for both building the seed set and assessing the retrieved ranked list, our results clearly indicate that we can get along rather well with fairly small seed sets.

This work can be extended in several directions. First, our Institutional Review Board approval to conduct this research and to disseminate our annotated list of Arab journalists requires us to reverify that each account that we identify in this way is associated with a public attribution of the account as being owned by a journalist. We were able to satisfy this condition for 1,1231 out of the 1,231 journalists in our seed set or for which we have independent annotations.[11]

Other sources of text features might also be exploited in future work (e.g., tweet content, hashtag use, or URL content). Other types of social relationships (e.g., physical proximity) might also be tried. In addition, we can extend the search space to the two-hop neighbors of the seed-set. We might also productively explore other ways of combining social evidence, and of combining social and content evidence. One more aspect that needs examination is the geographical and topical distribution of the journalists in the seed-set. A diversified sampling can lead to results different from simple random sampling, or using Wikipedia to construct the seed-set. Finally, we might also try these ideas with other populations. We expect to find that some professions (e.g., professors) might yield results quite like that which we have seen with journalists, while perhaps for others (e.g., airline pilots) with different patterns of Twitterati fame, we might see very different results.

ACKNOWLEDGMENT

This work was made possible by NPRP grant# NPRP 6-1377-1-257 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

9. REFERENCES

[1] M. Bagdouri, W. Webber, D. D. Lewis, and D. W. Oard. Towards minimizing the annotation cost of certified text classification. In *CIKM*, 2013.

[2] S. Bergsma, M. Dredze, B. V. Durme, T. Wilson, and D. Yarowsky. Broadly improving user classification via communication-based name and location clustering on Twitter. In *NAACL*, 2013.

[3] S. Bergsma, P. McNamee, M. Bagdouri, C. Fink, and T. Wilson. Language identification for creating language-specific Twitter collections. In *LSM*, 2012.

[4] P. Bhattacharya, S. Ghosh, J. Kulshrestha, M. Mondal, M. B. Zafar, N. Ganguly, and K. P. Gummadi. Deep Twitter diving: Exploring topical groups in microblogs at scale. In *CSCW*, 2014.

[5] S. Bowman and C. Willis. We media: How audiences are shaping the future of news and information. `hypergene.net/wemedia`, 2003. Accessed: 2015-01-19.

[6] C. Buckley and E. M. Voorhees. Retrieval evaluation with incomplete information. In *SIGIR*, 2004.

[11] `http://cs.umd.edu/~mossaab/files/arab-journalists.tgz`

[7] Z. Cheng, J. Caverlee, H. Barthwal, and V. Bachani. Who is the barbecue king of Texas?: A geo-spatial approach to finding local experts on Twitter. In *SIGIR*, 2014.

[8] R. Cohen and D. Ruths. Classifying political orientation on Twitter: It's not easy! In *ICWSM*, 2013.

[9] W. B. Croft and D. J. Harper. Using probabilistic models of document retrieval without relevance information. *JDoc*, 35(4), 1979.

[10] D. Dailey and K. Starbird. Journalists as crowdsourcerers: Responding to crisis by reporting with a crowd. *CSCW*, 23(4-6), 2014.

[11] T. El-Ganainy, W. Magdy, and A. Rafea. Hyperlink-extended pseudo relevance feedback for improved microblog retrieval. In *SoMeRA*, 2014.

[12] C. Fink, J. Kopecky, and M. Morawski. Inferring gender from the content of tweets: A region specific example. In *ICWSM*, 2012.

[13] G. P. C. Fung, J. X. Yu, H. Lu, and P. S. Yu. Text classification without negative examples revisit. *TKDE*, 18(1), 2006.

[14] S. Ghosh, M. B. Zafar, P. Bhattacharya, N. Sharma, N. Ganguly, and K. Gummadi. On sampling the wisdom of crowds: Random vs. expert sampling of the Twitter stream. In *CIKM*, 2013.

[15] B. Han, P. Cook, and T. Baldwin. Text-based Twitter user geolocation prediction. *JAIR*, 2014.

[16] S. Han, D. He, J. Jiang, and Z. Yue. Supporting exploratory people search: A study of factor transparency and user control. In *CIKM*, 2013.

[17] M. Hu, S. Liu, F. Wei, Y. Wu, J. Stasko, and K.-L. Ma. Breaking news on Twitter. In *CHI*, 2012.

[18] W. Huang, I. Weber, and S. Vieweg. Inferring nationalities of Twitter users and studying international linking. In *HT*, 2014.

[19] J.-H. Kang and K. Lerman. Using lists to measure homophily on Twitter. In *ITWP*, 2012.

[20] B. Liu, W. S. Lee, P. S. Yu, and X. Li. Partially supervised classification of text documents. In *ICML*, 2002.

[21] W. Meng, L. Lanfen, W. Jing, Y. Penghua, L. Jiaolong, and X. Fei. Improving short text classification using public search engines. In *IUKM*, volume 8032. Springer, 2013.

[22] T. Miyanishi, K. Seki, and K. Uehara. Improving pseudo-relevance feedback via tweet selection. In *CIKM*, 2013.

[23] E. Mohammady and A. Culotta. Using county demographics to infer attributes of Twitter users. In *Joint Workshop on Social Dynamics and Personal Attributes in Social Media*, 2014.

[24] F. Mordelet and J. P. Vert. A bagging SVM to learn from positive and unlabeled examples. *PRL*, 2014.

[25] D. Nguyen, R. Gravel, D. Trieschnigg, and T. Meder. "How old do you think I am?" a study of language and age in Twitter. In *ICWSM*, 2013.

[26] J. H. Parmelee. Political journalists and Twitter: Influences on norms and practices. *JMP*, 14(4), 2013.

[27] M. Pennacchiotti and A.-M. Popescu. A machine learning approach to Twitter user classification. In *ICWSM*, 2011.

[28] S. Phuvipadawat and T. Murata. Breaking news detection and tracking in Twitter. In *WI-IAT*, 2010.

[29] Q. Qiu, Y. Zhang, J. Zhu, and W. Qu. Building a text classifier by a keyword and Wikipedia knowledge. In *ADMA*, 2009.

[30] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta. Classifying latent user attributes in Twitter. In *SMUC*, 2010.

[31] K. Sadamitsu, K. Saito, K. Imamura, and G. Kikui. Entity set expansion using topic information. In *HLT*, 2011.

[32] T. Sakai. Alternatives to bpref. In *SIGIR*, 2007.

[33] E. Zak. 4 questions with Liz Heron, the New York Times social media editor. `adweek.com/fishbowlny/-/250679`, 2012. Accessed: 2015-01-13.