BUILDING SEMANTIC CONTEXT: AN APPROACH FOR LEARNING CONTEXT FROM THE WEB FOR FACILITATING TWITTER ANALYSIS

NAZURA JAVED¹, B. L. MURALIDHARA²

¹ Department of Computer Science and Applications, Bangalore University, Bangalore, India
² Department of Computer Science and Applications, Bangalore University, Bangalore, India

ABSTRACT:

Tweets are cryptic and often laced with insinuation. Interpretation of user tweets cannot be done in isolation. Human beings can interpret the tweets because they possess the ‘Background/Contextual knowledge’. This knowledge enables them to understand the context of tweets and interpret the text. Emulating this interpretation ability in machines requires the machine to acquire ‘Contextual Knowledge’. In this paper we propose a novel technique of harnessing the contextual knowledge from the online sources and building a Labeled Context Corpus (LCC). Our contribution in this paper is a methodology that discovers trending topics and automates the construction of a LCC. Since most of the user tweets do not contain hashtags and associating the them with a topic label is a challenge; LCC can be leveraged for analysis of tweets even when they lack reference URLs or hashtags. Mining of tweets can provide valuable insight into societal sentiment.

Keywords: contextual corpus, text mining, machine learning, tweet analysis

[1] INTRODUCTION

Twitter has become a platform for individuals to express views and opinions on vital societal and political issues. The tweet text though limited to 140 characters can speak volumes, subject to the fact that it is interpreted with reference to the context. The tweets by themselves are highly cryptic and sparse. Hence applying mining techniques on them can be inconclusive.
Interpretation of tweets has been a topic of interest, challenge and research on account of the following reasons:

1. The tweet text is limited to 140 characters and hence is brief and sometimes cryptic.
2. The tweet text is unstructured and there is no adherence to grammatical syntax.
3. Tweet text comprises of slangs, incorrectly spelled words, abbreviations and phonetically spelled words.
4. Colloquial terms are liberally used.
5. Most of the tweets do not contain a hashtag. Thus determining the subject or topic of the tweet itself is a challenge.
6. Tweets are laced with insinuation and assume certain ‘World knowledge’ or contextual knowledge.

Table -1 shows some typical user tweets. Approximately 85% of the tweets do not contain hashtags. The URLs contained in the tweets are not complete; the last few characters of the URL are lost because of the 140 character limit on tweet text.

| User Tweets                                                                                                                                  |
| @AwanBr111 tweeted: JNU Scholar Claims Her Degree Blocked Due To Afzal Guru Row: A PhD scholar at the Jawaharlal University today ... https://t.co/4x4Lo2pHzw |
| @Imdipti_ tweeted: RT @Congress2019: Vision less, Only 1.8% got 150 days of work in drought-hit states: MGNREGA data @JhaSanjay @Mumbai_Congress https://t.co… |
| In times of rural distress, MGNREGA has been recognised as the most effective means of providing the healing touc…NDA destroying MGNREGA https://… |
| @IndianExpress tweeted: #ExpressFrontPage Vijay Mallya’s plea against ED’s claim rejected, court issues non-bailable warrant https://t.co/S9f4REhbe1 |

We propose a novel technique of context generation that would facilitate tweet interpretation. The methodology proposed by us harnesses the online resources and obtains contextual data. It enables the machine to learn all the context terms for a topic. The relevant and associated terms are extracted and a 3 tuple $U_i = (O_i, CT_{ij}, W_{ij})$ is constructed comprising of popular topic $O_i$ and a table containing context terms $CT_{ij}$ and weights $W_{ij}$ for topic $O_i$. There is one such 3 tuple $U_i$ for each ‘popular topic’. The tuples $U_1...U_n$ are combined to form a LCC. This LCC can be leveraged for learning the reference for a user tweet and tagging it.

Our contribution in this paper is a methodology that discovers trending topics and automates the construction of LCC. Since most of the user tweets do not contain a hashtag; identifying the associated topic or event is a challenge. LCC, a resource of related, relevant topic-based terms can be leveraged for topic classification and analysis of tweets.

The remainder of the paper is structured as follows: In section 2 we overview the related literature. Section 3 proposes the methodology for Corpora building and tagging. Section 4 describes the experiments and the results thereof. This paper concludes by examining the scope/ boundary of the proposed techniques and future scope for enhancement and research.

[2] RELATED WORK
Twitter can be used for reporting news, sharing information, discussing/debating issues and expressing sentiments. Since Twitter is easily accessed by wireless devices, the reach of this medium is expanded. The potential of drawing in citizens through online consultations and crowd-sourcing are discussed in the following works. Twitter can enable many novel and powerful uses for government applications. The relevance of twitter messages and their analysis are highlighted in [1]. Publishing policy decisions and synthesis of opinion regarding the policies for the purpose of decision making are investigated in [2].

Social media analytics involves a three-stage process: capture, understand, and present. The capture stage involves obtaining relevant social media data by monitoring or “listening” to various social media sources, archiving relevant data and extracting pertinent information. All the data extracted is not useful and hence the understand stage selects relevant data for modeling, removes noisy, low quality data, and employs various advanced data analytic methods to analyze the data retained and gain insights from it [3].

Social media analytics is concerned with developing and evaluating informatics tools and frameworks to collect, monitor, analyze, summarize, and visualize social media data, usually driven by specific requirements from a target application. However, social media analytics faces several challenges such as semantic inconsistency/inaccuracies, misinformation and lack of structure. Also the dynamic nature and the sheer size of social media content impede analysis [4]. The limited length of a tweet and no restrictions on its writing styles results in grammatical errors, misspellings, and informal abbreviations [5]. Tweets contain highly irregular syntax and nonstandard use of English. Normalizing Twitter posts and converting them into a more standard form of English, so that standard machine translation (MT) and natural language processing (NLP) techniques can be more easily applied to them was researched in [6]. Tweets are first preprocessed to remove noise. Then they are fed into a machine translation model to convert them into Standard English. All the above works thus highlight the relevance of cleaning and structuring tweets for reliable semantic analytics.

The web is a valuable resource of open source tools, knowledgebases (KB), online news and contents which can be accessed using API libraries. It is therefore imperative to leverage these resources and use them for analysis. Stanford University’s NER [7], WordNet [8], ConceptNet [9], DBpedia and Yago [10] are some such resources utilized by researchers for text analytics. A novel Yago-based Summarizer that relies on ontology to capture the actual meaning and context of the document sentences and achieve disambiguation was proposed in [11]. A Question Answering system which takes the input in natural language, converts it into SparQL queries and retrieves the relevant contents from Yago Knowledgebase was developed in [12]. DEANNA, a framework for natural language question answering translates questions into a structured SPARQL query that can be evaluated over KBs such as Yago, DBpedia, Freebase and other Linked Data sources [13]. IBM’s Watson project impressively demonstrated a new level of deep QA, by winning the popular quiz show Jeopardy against two human champions [14]. The significance of the online resources and tools are highlighted in the above works.

Classification of short texts and the problems therein are discussed in [15, 16]. Short texts do not provide sufficient word occurrences and hence traditional classification methods such as "Bag-Of-Words" have limitations. Domain specific features are extracted to enable the classification of text to a predefined set of generic classes such as news, events, opinions, deals, and private messages. The use of supervised techniques for solving the problem of linguistic
variability in tweets is demonstrated in [16]. This paper focuses on the linguistic variability in the information available on social media and solves the variability problem by transforming information into generalized features. Both the above works use domain feature extraction for aiding classification.

The problem of tweet classification and the different approaches to address the same are discussed in the following works. Tweets are sparse in content and context. Hence context of the tweet is obtained by mapping the content onto most similar Wikipedia pages in [17]. Distances between pages are used as a proxy for the distances between messages. This helps to perform a more accurate classification of a set of twitter messages. A distantly supervised approach for classifying tweets is used in [18]. The tweets contain a plethora of distinctive named entity types (companies, products, bands, movies, and more). Almost all of these are infrequent and hence even a large sample of annotated tweets is not a good sample for training the classifier. Hence a distantly supervised approach using large dictionaries of entities gathered from Freebase are used for obtaining the entity’s context. Our work is similar to the above approach with respect to context building. We capture context data for the tweet by tapping the online resources and news articles on the web. We add descriptive topic labels to this context and build a LCC which can be used for tweet classification.

N-gram summaries from the tweets are generated using internal, external and mined internal content [19]. This paper uses external sources like news articles and other web a content to enrich the tweet and facilitate a broader interpretation.

Extracting facets from tweets for designing appropriate faceted search strategies on twitter are discussed in [20]. This work extracts facets from the tweets by considering the contents of the URLs mentioned in them. These facets yield an adaptive search strategy for search and exploration of twitter content. This paper highlights the selection of facets for effective search. In our work of construction of LCC, discovering and harnessing the pertinent web content is important. Hence building an appropriate search phrase that enables retrieval of relevant context is important.

External resources and KBs can help guide the interpretation of tweets. Enhanced classification and mining can be achieved by using these resources. The different approaches and techniques that leverage the additional resources for a more optimized result are discussed in [21, 22, 23, 24]. The classification of twitter messages into different topics is researched in [22]. This work extracts the features from the words in the tweets, from the URLs mentioned in the tweet and words from the user profile. Leveraging the web resources for enriching the existing internal business data records and using them for analysis such as customer segmentation, competitive intelligence and fraud detection are explored [23].

The significance of a KB and its effective utilization for tweet interpretation are highlighted in [24]. It is difficult to interpret some tweets in isolation. But if the context of this tweet is available in the KB, then inference and interpretation is possible. KB is at the heart of any powerful machine learning system. Social Genome, a large real-time social knowledge base was built. This KB was the heart of Kosmix project and was used to power most of its applications. Social Genome was built, using Wikipedia, a set of other data sources, and social media data. Our Context building approach differs from the above works in the following respects:
• We do not build context from the relatively fixed resources like dictionaries and ontology. Since our primary objective is to tag and later analyze the public tweets relating to current affairs, political events and societal scenarios, fixed ontology and dictionaries are neither appropriate nor adequate. Hence we build context by tapping the web resources like news articles and the links included in tweet text etc. which are dynamic in nature.

• We structure the web content and build a Labeled Corpus (LCC). This corpus has the potential of being leveraged for classification and further analysis of user tweets.

[3] METHODOLOGY

The objective of this work is to build a Labeled Context Corpus which would enable a semantic interpretation of tweets. Human beings interpret tweets and understand them because of their ‘Context knowledge’. They acquire this knowledge through sources like newspaper, television or online channels. In our work, we discover the popular or trending topics from the tweets of the popular political leaders. After discovering the topics, we perform web scraping to collect the relevant and associated web contents. We finally build a ‘Labeled Context Corpus’ (LCC) by crunching these contents. Each tuple of $LCC$ contains a popular topic, relevant context terms and their associated weights. The weights reflect the significance of the terms with respect to the topic. This LCC provides scope for mining of user tweets even when they lack hashtags or other direct references. Figure-1 provides an overview of the proposed technique. The following sections describe the stages involved.

[3.1] Collection of Popular Tweets:

The political leaders with more than a million twitter followers and belonging to different political parties of India are considered as popular leaders. The twitter messages tweeted by them are termed as ‘Popular tweets’. Our rationale behind collecting these tweets is stated hereunder:

(1) Political entities or leaders tweet about the current and relevant events or issues. These tweets are more meaningful than the other user tweets and can be parsed more easily. Most of these tweets contain hashtags. Parsing these tweets enables us to extract hashtags, entities and other specific domain terms.

(2) Parsing popular tweets yield meaningful topics. Collecting contextual information about these subjects/topics and developing a contextual corpus can facilitate interpretation of cryptic public tweets.
[3.2] Pre-process Popular Tweets:

The ‘Popular Tweets’ are subjected to pre-processing. The following preliminary processing is done with a view of facilitating further processing and analysis [25].

1. Translation of abbreviations into correct English words.
2. Separation of hashtags, @tags and URLS from the tweet text.
3. Extraction of metadata like username, location, retweet count, followers count etc.
4. Extraction of Domain terms that represent topics or events. These terms are not common entities like locations, and cities. They are extracted used Named Entity Recognition (NER) and can be entities like companies or institutions for e.g. Kingfisher Airlines, JNU (Jawaharlal Nehru University) or policies /schemes like MGNREGA – Mahatma Gandhi National Rural employment Guarantee Act or colloquial terms like ‘Mann ki Baat’

Table 2 shows the sample pre-processed tweets of popular political leaders.

<table>
<thead>
<tr>
<th>User Tweet</th>
<th>Username</th>
<th>Cleaned Tweet</th>
<th>#tags</th>
<th>location</th>
<th>URL</th>
<th>Terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>ret @sonalndtv: the real reason kashmiris who actually raised anti-india slogans in jnu are not in jail: <a href="https://t.co/juktkaquy3">https://t.co/juktkaquy3</a></td>
<td>ArvindKejriwal</td>
<td>retweet: sonalndtv: the real reason Kashmiris who actually raised anti india slogans in Jnu are not in jail</td>
<td></td>
<td></td>
<td><a href="https://t.co/juktkaquy3">https://t.co/juktkaquy3</a></td>
<td>Kashmir JNU</td>
</tr>
<tr>
<td>want to know more about #budget2016 &amp; its positive impact? check out this site. #vikasakbudget <a href="https://t.co/tn2ivddpg">https://t.co/tn2ivddpg</a></td>
<td>NarendraModi</td>
<td>Want to know more about budget2016 its positive impact check out this site</td>
<td>#budget2016</td>
<td></td>
<td><a href="https://t.co/tn2ivddpg">https://t.co/tn2ivddpg</a></td>
<td></td>
</tr>
</tbody>
</table>

1 ‘Mann ki Baat is an Indian radio program hosted by the Prime Minister of India, Narendra Modi in which he addresses the people of the nation on radio, DD National and DD News channels.
budget2016 lacks both vision & conviction. A list of new promises w/o any account of the failure of tall promises made in last 2 budgets!

OfficeOfRG

budget2016 lacks both vision conviction. List of new promises without any account of the failure of tall promises made in last 2 budgets

India; Timezone: Chennai

budget 2016

[3.3] Discovering Popular Topics (PT):

The popular topics and events are discovered using the following steps:

1. Hashtags, if present, are extracted from the popular tweets. If the hashtag contains conjoined words for e.g. #modibudgettest; the hashtag is split.

2. A Document Term Matrix (DTM) with term frequencies (TF) is constructed for all the popular tweets and the terms with the highest frequencies are extracted.

3. ‘Popular Terms/topics’ are selected on the basis of the frequencies of the hashtagged (split) and other frequent terms.


The Popular topics discovered may be specific terms like ‘Budget2016’ or relatively general words like ‘JNU’. For building a Semantic Context Corpus, it is necessary to scrap only the topic-related web content. Search for general terms like ‘JNU’ may yield irrelevant content. Hence we generate a Search Phrase by appending the PT with the month, year of tweets and the country.

We use Google Trends for Suggestions regarding the above Search phrase and form a CSP, which is the concatenation of PT, month, year of tweets and the country.

[3.5] Web Scraping for Context:

We search for the related news articles and web content using the CSP. We then scrap out the contents from the retrieved URL sites. Web scraping is done in two ways: a) ‘Popular tweets’ containing PTs, sometimes contain links to the associated URL sites. Text content is scraped from these URL sites. b) Text content is scraped from the URL sites which are searched/retrieved using the CSP.

Figure-2 shows a diagrammatic representation of the Web scraping process. It depicts one instance of the web scraping process. This process takes one CSP and the embedded URLs as its input and yields one folder containing related documents/articles. A collection of such folders forms the ‘Context Corpus’ (CC). The CC can be defined as under:

\[
CC = \{ f d_1, f d_2, .. f d_n | f d_i = \{ d_1, d_2, .. d_m \} \}
\]

wherein \( f d_1, f d_2, .. f d_n \) are the \( n \) folders.

\( d_1, d_2, .. d_m \) are documents within one folder \( f d_i \).

Each document \( d_1, .. d_m \) contains articles for a topic.
[3.6] Generation of the Term Document Matrix:

The CC is converted into Term Document Matrix (TDM). TDM is generated after pre-processing like a) Removal of punctuations, special symbols and stop words; b) Standardization of Proper Nouns. For e.g. Prime Minister Narendra Modi, PM Mr. Modi, PM Modi, are the different ways in which Prime Minister Narendra Modi is represented. These nouns are translated into standard form like for e.g. ‘NarendraModi’ for Prime Minister Narendra Modi. Figure - 3 shows a document belonging to the CC. The different forms of the proper noun Narendra Modi are highlighted.

![Figure: 3. Document showing a Proper noun in its different forms](image)

Each folder of the CC is converted into one TDM (Table 3). Rows contain the ‘terms’ and columns represent the ‘documents’.

<table>
<thead>
<tr>
<th>Table 3. Term Document Matrix $M_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Terms 1 to $j$ in $M_i$</td>
</tr>
<tr>
<td>$term_1$</td>
</tr>
<tr>
<td>$term_2$</td>
</tr>
<tr>
<td>...</td>
</tr>
<tr>
<td>$term_j$</td>
</tr>
</tbody>
</table>

[3.7] Generation of LCC:
This process takes the TDMs $M_1,..,M_n$ as input and generates a LCC. The row sum is calculated for each row of TDM $M_i$. The row sum reflects the frequency of the usage of the term in the CC. The dimensionality of the TDM is reduced by filtering out the insignificant/infreqent terms using the Feature ranking method. The frequencies of terms are considered as ranks and the top ranked terms are selected using the Pareto analysis principle. This process yields a 3 tuple $U_i = (O_i, CT_{ij}, W_{ij})$. The process is algorithmically defined hereunder.

**Table 4. Terms/Notations used in the Algorithm**

<table>
<thead>
<tr>
<th>$M_1,..,M_n$</th>
<th>$M_1,..,M_n$ are TDMs for the Popular Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>$tf_{ijk}$</td>
<td>Term frequency of term $jk$ of Matrix $M_i$</td>
</tr>
<tr>
<td>$Rs_{ij}$</td>
<td>The rowsum (row sum) for each row of Matrix $M_i$ i.e. the total frequency of term $t_{ij}$ of matrix $M_i$</td>
</tr>
<tr>
<td>$CT_{ij}$</td>
<td>Context term; Term is selected using Pareto Analysis technique</td>
</tr>
<tr>
<td>$Ms_i$</td>
<td>Matrix sum (Sum of all term frequencies (tf) for Matrix $M_i$)</td>
</tr>
<tr>
<td>$W_{ij}$</td>
<td>Computed weight of context term $CT_{ij}$</td>
</tr>
<tr>
<td>$U_i = (O_i, CT_{ij}, W_{ij})$</td>
<td>3 tuple $U_i$ wherein $O_i$ is the Popular topic, $CT_{ij}$ and $W_{ij}$ are the context terms and weights for Popular topic $O_i$</td>
</tr>
<tr>
<td>$l_i$</td>
<td>Label $l_i$ for Popular topic $O_i$</td>
</tr>
<tr>
<td>$LCC$</td>
<td>Labeled Context Corpus built from tuples and their respective labels</td>
</tr>
</tbody>
</table>

**Algorithm buildLabeledCorpus (Matrices $M_1,..,M_n$ for each Topic $O_1..O_n$, LCC)**

Input: Matrices $M_1,..,M_n$; wherein each $M_1,..,M_n$ is a Term Document matrix $T \times D$ with $j$ rows and $k$ columns for topics $O_1..O_n$ respectively

Output: LCC

**for each Matrix $M_i$ of Matrices $M_1,..,M_n$**

**for each term $j$ of terms $1..j$ of Matrix $M_i$**

Compute $Rs_{ij} = \sum_{k=1}^{k} tf_{ijk}$

**end for**

Select terms $CT_{ij}$ using Feature ranking method

set $m = count(CT_{ij})$ [count of selected context terms]

Compute $Ms_i = \sum_{i=1}^{i} \sum_{k=1}^{k} tf_{ijk}$

**end for**

**for each context term $CT_{ij}$**

Compute $W_{ij} = Rs_{ij} / Ms_i \times 100$

**end for**

Generate Tuple $U_i = (O_i, CT_{ij}, W_{ij})$

**end for**

**for each Tuple $U_i$**

Assign label $l_i$ to $U_i$ such that $U_i \leftrightarrow l_i$

$LCC = LCC \parallel (l_i \parallel U_i)$

[LCC constructed by concatenating all tuples and their respective labels]

**end for**
[4] EXPERIMENTS AND RESULTS

We collected the tweets of the popular political leaders of India for e.g. Prime Minister of India, Mr. Narendra Modi, Chief Minister of Delhi and the national convener of AAP party, Mr. Arvind Kejriwal and the President of Congress party, Mr. Rahul Gandhi using Python twitter Search API. We collected 482 tweets of these popular political leaders. These mainly included the tweets made by them during the period of February 2016 to April 2016. We pre-processed the tweets and also extracted the metadata, hashtags, @tags and hyperlinks. The procedure for pre-processing, and building a clean corpus was automated. The Popular topics were discovered using the steps stated in section 3.3. In our experiment it was found that the terms ‘JNU’, ‘Mannkibaat’, ‘MGNREGA’ were frequently mentioned in the tweets. We derived the CSP and used Microsoft Bing search API to search for the CSP and capture the contextual content. Bing search yielded the URLs using its ranking algorithms. We selected the top ten URLs. We restricted our web scraping to only ten sites because our repeated experiments revealed that the top ten URLs yield adequate and optimum content for building LCC. We scraped the web contents from the URLs mentioned in the popular tweets. Thus for each popular topic we had one folder of documents. A collection of all the folders made up the CC. We used the R mining tool to pre-process the CC and build TDMs. The algorithm for building LCC was coded in Python. The JSON representation of the LCC is shown in Figure- 4. In our experiment, the PT ‘JNU’ was extracted from the popular tweets (Table 2). The CSP, ‘JNU February 2016 India, antinational slogans’ was derived and the context documents were collected.

The above experiment substantiates our proposed technique. It enables the machine to build the context for the trending/popular topics. The LCC generated at the end of this process provides scope for classification and analysis of tweets.

[5] CONCLUSION AND FUTURE SCOPE

Most of the user tweets do not contain a hashtag. They are cryptic and lack sufficient context. Hence it is difficult to tag them and analyze them. Though the untagged tweets are difficult to tag/classify per se, there is a scope of interpreting them if they are supported by the contextual data. This work demonstrates the methodology of learning context for the popular topics or events and structuring the same as LCC. The LCC contains the context terms for the popular, trending topics. This provides future scope for classification and interpretation of tweets that lack sufficient background, context or hashtag. The tweets that cannot be classified can be abandoned and not used for further mining. This work has the potential of being used for tagging and analysis of user tweets to gain insight into societal issues, public sentiments and opinions.
<table>
<thead>
<tr>
<th>Category</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freedom</td>
<td>0.010</td>
</tr>
<tr>
<td>Jnu</td>
<td>0.145</td>
</tr>
<tr>
<td>Kamhalyakumar</td>
<td>0.121</td>
</tr>
<tr>
<td>Police</td>
<td>0.039</td>
</tr>
<tr>
<td>Political</td>
<td>0.014</td>
</tr>
<tr>
<td>Poster</td>
<td>0.014</td>
</tr>
<tr>
<td>Sedition</td>
<td>0.058</td>
</tr>
<tr>
<td>Slogan</td>
<td>0.077</td>
</tr>
<tr>
<td>Speech</td>
<td>0.024</td>
</tr>
<tr>
<td>Students</td>
<td>0.072</td>
</tr>
<tr>
<td>Unarkhailid</td>
<td>0.106</td>
</tr>
</tbody>
</table>

Figure: 4. JSON representation of the LCC
REFERENCES


