Journal of Computer Science

Original Research Paper

A Hierarchical Clustering Approach for DBpedia based Contextual Information of Tweets

[1]Venkatesha Maravanthe, [2]Prasanth Ganesh Rao,
[3]Anita Kanavalli, [2]Deepa Shenoy Punjalkatte and [4]Venugopal Kuppanna Rajuk

[1]Department of Computer Science and Engineering, VTU Research Resource Centre, Belagavi, India
[2]Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bengaluru, India
[3]Department of Computer Science and Engineering, Ramaiah Institute of Technology, Bengaluru, India
[4]Department of Computer Science and Engineering, Bangalore University, Bengaluru, India

Article history
Received: 21-01-2020
Revised: 12-03-2020
Accepted: 21-03-2020

Corresponding Authors:
Venkatesha Maravanthe
Department of Computer Science and Engineering, VTU Research Resource Centre, Belagavi, India
Email: venkatesha.uvce@gmail.com

Abstract: The past decade has seen a tremendous increase in the adoption of Social Web leading to the generation of enormous amount of user data every day. The constant stream of tweets with an innate complex sentimental and contextual nature makes searching for relevant information a herculean task. Multiple applications use Twitter for various domain sensitive and analytical use-cases. This paper proposes a scalable context modeling framework for a set of tweets for finding two forms of metadata termed as primary and extended contexts. Further, our work presents a hierarchical clustering approach to find hidden patterns by using generated primary and extended contexts. Ontologies from DBpedia are used for generating primary contexts and subsequently to find relevant extended contexts. DBpedia Spotlight in conjunction with DBpedia Ontology forms the backbone for this proposed model. We consider both twitter trend and stream data to demonstrate the application of these contextual parts of information appropriate in clustering. We also discuss the advantages of using hierarchical clustering and information obtained from cutting dendrograms.

Keywords: Context Modeling, DBpedia, Extended Contexts, Hierarchical Clustering, Short Text Clustering, Twitter Data Mining

Introduction

Adoption of Social Web as the platform for people to exchange opinions lead to a high volume of user-generated content. The availability of affordable smart devices and easy access to the internet via wireless mediums such as WiFi, 4G, etc., has greatly aided this transition. Smartphone adoption, in particular, has resulted in a further increase in the user base for social applications. The technology shift has promoted the evolution of multiple social media platforms, prominent of which are Facebook, Twitter, Instagram, LinkedIn, etc. Users freely exchange information over these platforms via text, images and videos. All the social platforms provide the necessary security and privacy features. The accessibility to the information is largely defined and controlled by the users themselves. In the microblogging space, Twitter is the most popular social media website allowing users to assimilate concise information directly from the genesis. More than half of the world's population today are active Internet users with around 1.3 billion twitter users. Recent statistics presented an average of 500 million tweets posted by 100 million active users everyday[a].

Twitter works on the concept of followee and follower. Most content on twitter is publicly accessible. Some users however have opted for private accounts and their activities are often not available. Abundance and simplified access of tweets leads to a problem of finding the right content and mining useful information from a given user's perspective. Human tendency has always been to stay curious about diverse topics and inability to find tweets suiting their interests can be discouraging. Hence there is a need for efficient frameworks that identify topics, tag contexts that eventually help define a foundation for development of context aware applications. We do find numerous Apps in popular Marketplaces working on information derived

[a] https://www.omnicoreagency.com/twitter-statistics/

by processing twitter data. Such apps include recommender systems, in-app marketing, sentiment analysis, market insights, crisis management etc.

DBpedia[b] knowledge-base is a crowd-sourced project developed by extracting contextual data from Wikipedia. It uses Linked Open Data standard consisting of 3 billion Resource Description Framework (RDF)[c] triplets and provides a mechanism to query the relevant content and links mainly Wikipedia and Yet Another Great Ontology (YAGO)[d] categories along with many external web pages. Multilingual support is also provided by DBpedia. (Lehmann *et al*., 2009; 2015) provide more insights into design of this framework for building the Ontology and maturity of the available content. They also describe the mechanisms to access information via online interfaces.

Hierarchical clustering is one of the cluster analysis methods in data mining to group similar data points to clusters. Hierarchical clustering works with the help of cluster dissimilarity and cluster linkage which are explained in subsequent sections. This form of clustering is advantageous to visualize meaningful taxonomies and nested clusters. There are two different ways to perform hierarchical clustering:

- *Agglomerative (bottom-up) approach* where each sample is a single cluster then merged form a single cluster
- *Divisive (top-down) approach* which starts as a single cluster which is broken down until one cluster of each sample is left

The first part of this work presents a model for extracting valid contexts from tweets using DBpedia followed by demonstrating how clustering can be applied to these contexts. Section Literature Survey provides an overview of existing similar models and the applicability of their contributions to our work. Section Problem Definition talks about the problem addressed in this work. Section Methodology outlines our optimized approach to determine contexts and clustering of the same. Section Experimental Results presents the results and Section Conclusions and Future Work wraps up the finding while noting the scope for future enhancements.

Literature Survey

Usage of Wikipedia and DBpedia as a knowledge base for mining text data has been long part of Semantic Mining. Bontcheva and Rout (2012) survey the various approaches for making sense of social media data streams. Authors highlight the importance of Linked Open Data resources, entity linking with Wikipedia articles and usage of Wikipedia categories. Gabrilovich and Markovitch (2007) further show the versatility of concepts derived from Wikipedia and proposes 'Explicit Semantic Analysis (ESA)' for computing semantic relatedness in natural language texts. Ramanathan and Kapoor (2009) propose a model for creating user profiles with the help of Wikipedia. Framework by Genc *et al*. (2011) discuss leveraging Wikipedia to map tweet to its semantic space, to calculate distance between tweets, helping better classification. Muñoz García *et al*. (2011) describe a topic recognition scheme by linking keywords to a ranked list of DBpedia resources. Authors in (Hamdan *et al*., 2013) utilized DBpedia along with WordNet and SentiWordNet as a combination for sentiment classification.

User interest modeling is an important application of Semantic Mining. Initially Michelson and Macskassy (2010) use 'Named Entity Recognition (NER)' for getting entities and disambiguates leveraging Wikipedia for generating Twopics. Wikipedia concept linking in Lu and Lam (2012), put forth expansion of user's interest and results show better recommendation using these interests. Kapanipathi *et al*. (2014) process a hierarchy on 'Wikipedia Concept Graph (WCG)' to come up with 'User Interest Generator' and 'Interest Hierarchy Generator', mapping user's primitive interests to Wikipedia hierarchy. Shah *et al*. (2018) propose enrichment technique using DBpedia Ontology, generating niche interest and inferred general interest and works for least active users as well. Interests identified using DBpedia aggregates into a user profiling framework in (Orlandi *et al*., 2012).

Contexts or Topic identification promoted many recommender systems using other knowledge-bases than DBpedia. Abel *et al*. (2011) represent user modeling with entity identification via OpenCalais[e]. A news recommendation system has been built on top of user modeling and considers the temporal dynamics of profile changes. Initiation from work (Pla Karidi, 2016) manifests to a complete recommender architecture in (Pla Karidi *et al*., 2017) suggests both tweets and followees. This system takes advantage of Alchemy API[f] for deriving contexts to build a Knowledge Graph of 1092 nodes and 1323 edges. A super set of 1092 concepts may not be sufficient in specific areas and DBpedia offers an alternative to explore.

Papneja *et al*. (2018) propose a content recommender related to user interest. DBpedia Spotlight[g] serves purpose to find mapping between domain ontology and DBpedia classes. Romero and Becker (2017) describe a classification framework, taking advantage of DBpedia for enriching semantic features. DBpedia spotlight connects terms to their respective URI for semantic enrichment.

[b] https://wiki.dbpedia.org/
[c] https://www.w3.org/RDF/
[d] https://en.wikipedia.org/wiki/YAGO_(database)
[e] http://www.opencalais.com
[f] https://www.ibm.com/watson/developercloud/alchemylanguage
[g] https://www.dbpedia-spotlight.org/

DBpedia and Wikipedia based solutions can be found in combination with clustering. Szczuka *et al*. (2012), authors have used DBpedia dictionary and matched against respective concepts for converting texts from scientific documents. Here it is clearly concluded that the DBpedia concept representation of clusters are in line with manually assigned cluster labels. Likewise, in (Schuhmacher and Ponzetto, 2013) web search results are processed with DBpedia Spotlight for snippet semantification and topic assignment leading to better quality of clusters formed. Hu *et al*. (2009) present a method to cluster different sets of documents by generating document-category matrix built on top of Wikipedia term-concept matrix. Results show that Wikipedia category information yields better cluster output along with hierarchical clustering methods.

In the review work, Alnajran *et al*. (2017) have compared 13 different research works on applications of clustering for mining twitter dataset. Although the performance is low for hierarchical clustering, quality of clusters is pointed out to be much better. Twitter event detection using hierarchical clustering after computing pairwise distances of tweet-by-term matrix is proposed in (Ifrim *et al*., 2014). Experiments have shown hierarchical clustering can process 24 h stream data in 1-hour time-frame with an accuracy of 80%.

Flisar and Podgorelec (2018) frames a classification model for tweets using DBpedia and is similar to our effort for identifying contexts. This work makes use of DBpedia Spotlight and queries DBpedia ontology for enrichment of data. In our prior work (Venkatesha *et al*., 2019), we attempt to find extended contexts and provide a scalable framework along with relevant data filtering. Vicient and Moreno (2015) have recommended hierarchical clustering for topic discovery in tweets. As a first step, semantic annotations are done on hashtags of tweets with the help of WordNet and Wikipedia categories. These annotated hashtags are the input for bottom-up hierarchical clustering procedure using complete linkage identical to what we are proposing. Saraçli *et al*. (2013) provide a detailed comparison of hierarchical clustering methods and help to determine right distance measures, thus guiding us for better decision making on clustering approach.

Problem Definition

Given the large set of tweets, model a framework to generate and cluster contexts for those tweets. Framework should consider perform the below outlined objectives:

1. Read and process the set of tweets resolving ambiguities
2. Generate primary context(s) from tweets
3. Get the extended context(s) for every primary context obtained
4. Cluster the primary/extended context(s) to visualize extracted metadata
5. Discover associated context(s) information at different levels of clusters

First consideration should be a proper tool to work with Named Entity Recognition and disambiguation to avoid confusion between different contexts. Long sentence or paragraph input can result in one or more applicable contexts or categories. Hence the second step should consider all the applicable contexts of text input. Third step is to find metadata or hidden data around the primary contexts helping to derive more meaningful information about the text. These additional information about primary contexts are termed as extended contexts. Considering the amount of text data generated on twitter every day, framework for extracting primary and extended contexts should scale for larger datasets. For better learning of tweets, fourth and fifth steps attempt to cluster the contextual knowledge of group of tweets.

Methodology

The proposed framework for generating contexts is illustrated in Fig. 1 and subsequent clustering is given in Fig. 4. We commence the process by acquiring data i.e., Tweets. Tweets can be stored in multiple formats. We have taken JSON[h] file format to store the input data for the ease of use. Java nio package[i] is primarily used to read files and Jackson[j] open source library is utilized to process JSON data. Every tweet in the input data goes through 'Primary Context Generator' and 'Extended Context Generator' and the outcome is stored in JSON format.

We use text "Roaming around amazon forest is a great experience" to understand the working of intended framework.

Primary Context Extractor

Section 3 highlights the difficulties with ambiguous context. We chose open source DBpedia spotlight (Mendes *et al*., 2011; Daiber *et al*., 2013) for handling ambiguity and deriving primary contexts:

- DBpedia Spotlight: DBpedia spotlight is an annotating tool built on top of DBpedia resources. It comprises of built-in disambiguation resolution on the phrases extracted from text. Results in Mendes *et al*. (2011) shows that DBpedia disambiguation evaluation has an accuracy of 80.52% for Spotlight Mixed approach. Daiber *et al*. (2013) further

[h] https://www.json.org
[i] https://docs.oracle.com/javase/7/docs/api/java/nio/packagesummary.html
[j] https://github.com/FasterXML/jackson

extends the model to multiple languages. This tool has both Web and Web-services based interfaces. In this paper, we rely on RESTful based web-services exposed connecting to '/candidates' endpoint. Interlinking of annotated term to DBpedia resources with a unique URI string is an upper hand of this tool. URIs can be directly connected to either DBpedia or Wikipedia resources.

Output of this step is a set of URIs based on the input text. Hence for a given tweet *t*, the output can be defined as set of URI/s termed as primary contexts:

$$PC = \{pc_1, pc_2, ...pc_n\} \quad (1)$$

For all $t \in T$, where T consists of multiple texts/tweets.

Sample text produces *Amazon_rainforest* as URI. Ambiguous tagging of Amazon as a company is avoided in API endpoint. Response from API contains additional attributes tagged to every URI such as *contextualScore*, *support*, *priorScore* and *finalScore*. This supplementary information is captured and stored, however not used in this work.

Extended Context Generator

Extended Context Generator defined in Fig. 1 deals with finding additional metadata for the extracted primary contexts. Extended contexts are queried through DBpedia Ontology applying SPARQL[k]. DBpedia SPARQL endpoint[l] is used to run respective queries and response is collected in JSON format. Query is modified to fetch resource class type of primary context in DBpedia Ontology. These types are in turn mapped to multiple named space schemas (e.g. *dbo*, *dul*, *yago* etc). Each primary context is mapped with valid response from SPARQL endpoint. Resultant data is subjected to filtering to extract only Wikipedia based categories. Java stream filters[m] are used to keep up with performance. Filtered results are termed as extended contexts:

$$EC = \{ec_1, ec_2, ..., ec_m\} \quad (2)$$

For all $pc \in PC$ and $m > n$ for tweets' set *T*. JSON object representation of primary context and extended contexts for the sample text is depicted in Fig. 2. Java Executors[n] capability is utilized to enable multiple threads reading tweets and to connect to two sources in parallel.

[k] https://www.w3.org/TR/rdf-sparql-query/
[l] http://dbpedia.org/sparql
[m] https://docs.oracle.com/javase/8/docs/api/java/util/stream/packagesummary.html
[n] https://docs.oracle.com/javase/tutorial/essential/concurrency/exinter.html

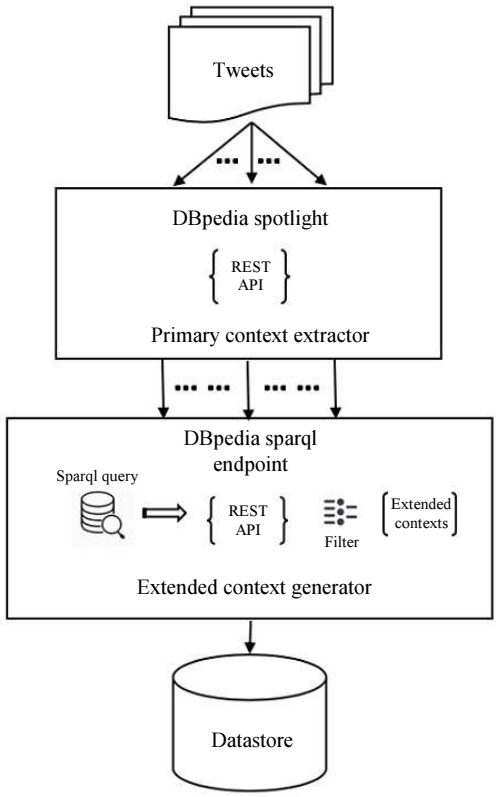


Fig. 1: Proposed architecture for context generator

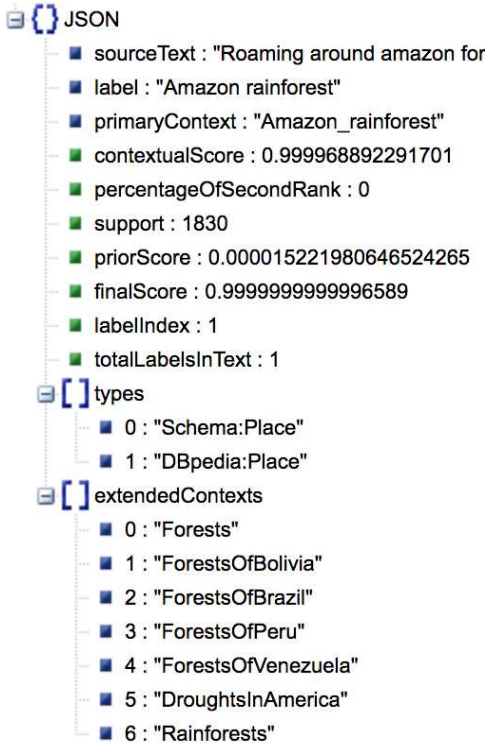


Fig. 2: Sample Text JSON Object Representation

Cosine Similarity Calculator

Generated extended contexts are validated using Cosine Similarity using PC and EC. Similarity measure calculation is dependent on vector representations of text data. We have considered Term Frequency-Inverse Document Frequency (TF-IDF) method to generate vectors. A term is an entry from either PC or EC.

Term Frequency is calculated for every document, by considering number of occurrences of a term in that document:

$$tf(t,d) = \frac{n_{t,d}}{\sum_d T} \tag{3}$$

IDF is calculated for every term w.r.t. the entire set of documents:

$$idf(t) = 1 + \log_e\left(\frac{N}{n_t}\right) \tag{4}$$

where, N is the total number documents and n_t number of documents where a particular term t appears.

TF-IDF weight w of term t in document d is calculated as:

$$w(t,d) = tf(t,d) * idf(t) \tag{5}$$

where, d is the document consisting of either primary contexts or extended contexts. TF-IDF weight w for each term is represented in vector format for every document.

Similarity measure is calculated on multiple documents extracted after sampling vectors of primary contexts or extended contexts. Representation of this approach is shown in Fig 3.

Given two n-dimensional vectors $W1$ and $W2$ of TFIDF weights, cosine similarity between these two vectors are represented as:

$$\cos(W1,W2) = \frac{W1 \cdot W2}{\|W1\| \cdot \|W2\|} \tag{6}$$

where, $W1$ and $W2$ are the vectors consisting of TF-IDF weights. Cosine similarity in (6) can be elaborated as:

$$\cos(W1,W2) = \frac{\sum_{i=1}^{n} W1_i W2_i}{\sqrt{\sum_{i=1}^{n}(W1_i)^2}\sqrt{\sum_{i=1}^{n}(W2_i)^2}} \tag{7}$$

where, $W1_i$ and $W2_i$ are components of $W1$ and $W2$ respectively. Higher cosine similarity value indicates more similar vectors. If vectors in comparison are exactly the same because of the underlying text, then the similarity value would be 1 which corresponds to the maximum possible value.

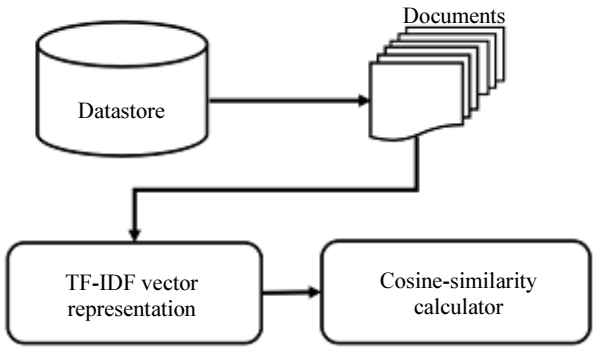


Fig. 3: Cosine similarity calculator

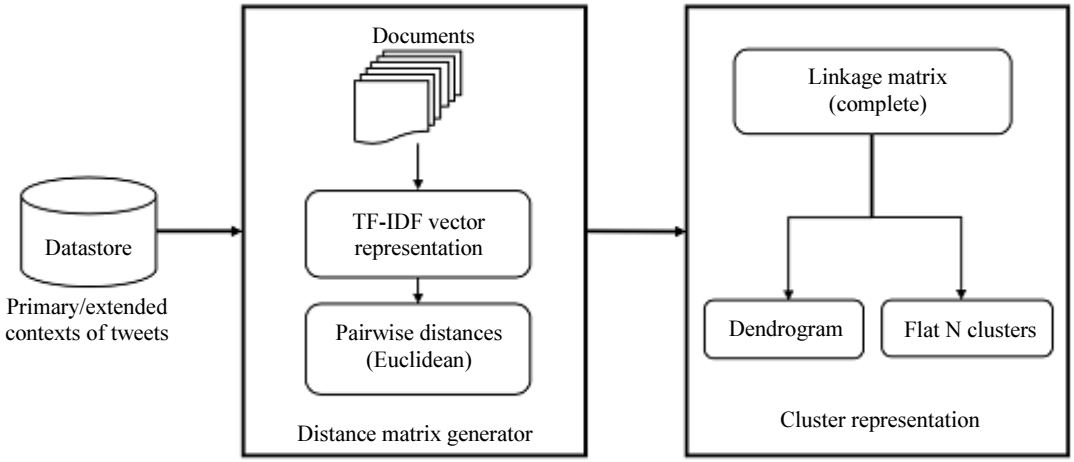


Fig. 4: Approach followed for clustering the contexts

Clustering Approach

Having validated the usefulness of contexts obtained by computing similarities, we further try to feed these metadata to a clustering method. The clustering step is mainly to identify useful patterns. Input data for clustering varies based on the type and size of tweets being sampled. The nature of input data makes it difficult to decide on the number of clusters or widow size, which are required with standard K-Means or Mean-Shift clustering approaches respectively. Similarly, algorithms like DBSCAN may not perform better if we have varied density clusters because of unrelated texts/tweets. Given these circumstances and the need for a generic approach, we chose to go with the bottom-up approach of hierarchical clustering i.e., Agglomerative Clustering.

In this study, we have followed the steps given in Fig. 4 for clustering data. Even for clustering, first we need to convert primary/extended contexts to vectors and we use the TF-IDF representation as explained in section 4.3. Next step is to select a distance measure to find dissimilarity between data points. Euclidean distance measure has been picked up after referring to the results of (Saraçli *et al*., 2013).

Here Euclidean distance is calculated as pairwise distances between two vectors. Hence the Euclidean distance between a pair of row vector x and y is given as:

$$dist_{euclidean}(x,y) = \sqrt{(x\cdot x) - 2*(x\cdot y) + (y\cdot y)} \quad (8)$$

For 2-dimensional sets x and y Euclidean distance can also be represented as:

$$dist_{euclidean}(x,y) = \| x - y \|_2 = \sqrt{\sum_i (x_i - y_i)^2} \quad (9)$$

The distance matrix from 8 is then used to generate a linkage matrix. Linkage is to determine the proximity of two clusters. For a large number of samples, it is favorable to apply complete linkage with most of the distance measures as indicated in (Saraçli *et al*., 2013). This is calculated at every iteration of cluster merge and suppose there are $|u|$ original observations $|u[0], u[1],\ldots..u[|u|-1]|$ in cluster u and $|v|$ original objects $|v[0], v[1],\ldots..,v[|v|-1]|$ in cluster v, then the complete linkage can be defined as:

$$L_{complete}(u,v) = \max\left(dist\left(u[i], v[j]\right)\right) \quad (10)$$

Clustering starts with computing distance matrix between each data point and merging two closest clusters until a single cluster is formed. For the implementation purpose we have used python SciPy[o] library along with python machine learning package scikit-learn[p].

Experimental Results

In order to evaluate the model proposed in section 4, we have considered already extracted set of tweets as the data source. Recent updates to twitter API policies in July 2018 and subsequent difficulties faced with API rate limits in our prior work (Rao *et al*., 2018), made us choose standardized readily available data. Existing data has been used and more focus is given to the tweets' context information extraction.

Datasets

We have considered two different datasets; one is containing tweets specific to trends and the other one being a stream of tweets.

- Tweets data dump from Kaggle consisting of top trends in October 2017 and corresponding tweets pulled from Twitter API throughout the month. Dataset is approximately 450MB, available in JSON format with hierarchy of date, trend and the tweets. Retweets are attributed to the count of respective tweets for organizing data duplication
- USA geolocated tweets dataset[q] comprising 200,000 tweets. Data was collected over a period of 48 hours and made available in excel file format. It also consists of Top 100 tweets in different groups such as retweets, favorites etc.

Contexts and Similarity

"#CatalanReferendum" trend from 1st day of October month is selected for generating contexts. There are totally 1297 unique texts tagged to this trend. Extracted tweets are randomly split into 5 equal sized chunks. These 5 sub lists are employed as input for the context generator and respective 5 documents store the primary and extended contexts. Merging these two documents result in 5 more documents leading to 3 different sets of 5 documents each involving primary, extended and (primary + extended) contexts.

Results of similarity scores considering all 3 contexts are given in Table 1. With the outcome, it is evident that either extended contexts or both the contexts give much better results than using only the primary contexts. Random split of specific trend might result in disjoint sets. In case of comparison between disjoint sets, the similarity measure after extending is a lower value e.g., Doc 1 v/s Doc 4 in Table 1.

[o] https://www.scipy.org/
[p] https://scikit-learn.org/stable/
[q] http://followthehashtag.com/datasets/free-twitter-dataset-usa-200000-free-usa-tweets/

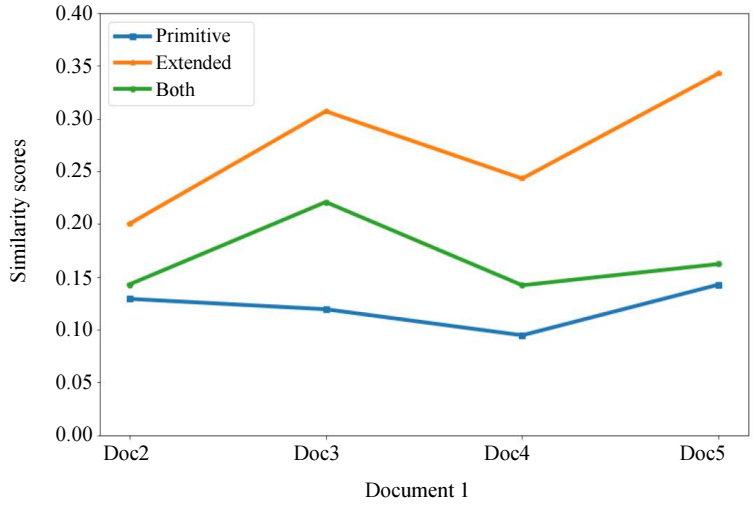


Fig. 5: Document 1 v/s other docs (#CatalanReferendum)

Table 1: Cosine similarity scores (#CatalanReferendum)

	Primary	Extended	Both
Doc 1			
Doc 2	0.12918	0.20017	0.14276
Doc 3	0.11931	0.30692	0.22088
Doc 4	0.09465	0.24314	0.14204
Doc 5	0.14270	0.34280	0.16220
Doc 2			
Doc 1	0.12918	0.20017	0.14276
Doc 3	0.12114	0.16489	0.29984
Doc 4	0.11502	0.15107	0.12784
Doc 5	0.14951	0.13948	0:21714
Doc 3			
Doc 1	0.11931	0.30692	0.22088
Doc 2	0.12114	0.16489	0.29984
Doc 4	0.11615	0.17085	0.19169
Doc 5	0.17495	0:24029	0:28183
Doc 4			
Doc 1	0.09465	0.24314	0.14204
Doc 2	0.11502	0.15107	0.12784
Doc 3	0.11615	0.17085	0.19169
Doc 5	0.16243	0.14718	0.15353
Doc 5			
Doc 1	0.14270	0:34280	0.16220
Doc 2	0.14951	0.13948	0.21714
Doc 3	0.17495	0.24029	0.28183
Doc 4	0.16243	0.14718	0.15353

For better understanding of the obtained results, plotting of Document 1 v/s Other documents is illustrated in Fig. 5. Cosine similarity for the entire #CatalanReferendum trend splits are represented in Fig. 6. Better results are acquired with extended contexts in most of the scenarios and both contexts in few cases.

We have experimented the model with few more trends extracted from dataset, outlined in Table 2. Similarity scores for these trends are preferable with extended and both contexts identical to #CatalanReferendum.

For fine-tuning the performance, number of threads to process tweets are kept configurable. Figure 7 shows time taken by the proposed system for Cristiano Ronaldo trend consisting of 794 tweets. With 250 threads program took approximately 30 sec for execution. On an average, 6000[r] tweets are posted every second. Applying filtering of specific trend to the live stream of tweets we might end up with roughly 1000 tweets per trend for processing. Proposed system can be calibrated with appropriate number of threads and can complete processing within 30 sec. Running the framework on a multi-core server should provide even better scalability.

When we compare our approach with (Vicient and Moreno, 2015), our effort overcomes the difficulties involved with semantic annotation of hashtags by using DBpedia. By observing the classification results on top of DBpedia based enriched data in (Flisar and Podgorelec, 2018), we intended to experiment clustering of contexts. Outcomes of chosen hierarchical clustering are explained in the subsequent section.

Cluster Representation

Clustering was carried out on both of the datasets to observe what kind of patterns will emerge. We had to do a bit of processing as pre and post steps while generating contexts:

- Removal of RT prefix from the tweets as DBpedia terming this as a separate context and linking it with category: RT (TV network)[s]
- Generated contexts are URIs which might contain commas in the string label, e.g., Hilo, Hawaii[t]. Therefore, a different delimiter had to be used to split the data to get actual context labels.

[r] http://www.internetlivestats.com/twitter-statistics/
[s] https://en.wikipedia.org/wiki/RT_(TV network)
[t] https://en.wikipedia.org/wiki/Hilo,_Hawaii

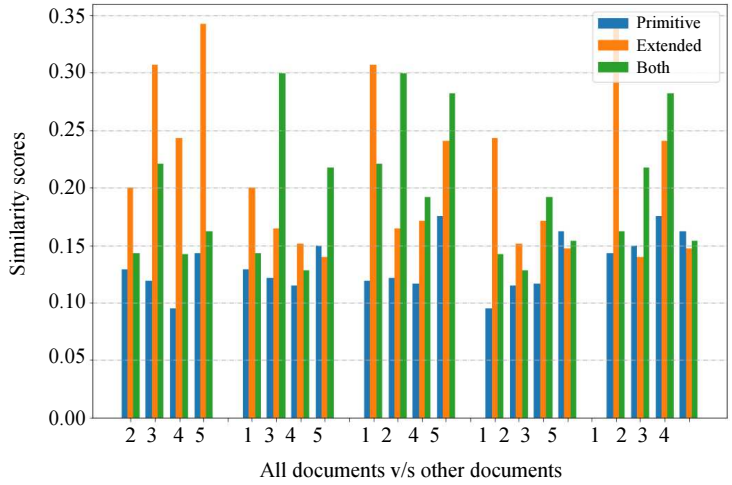


Fig. 6: Comparison of all documents (#CatalanReferendum)

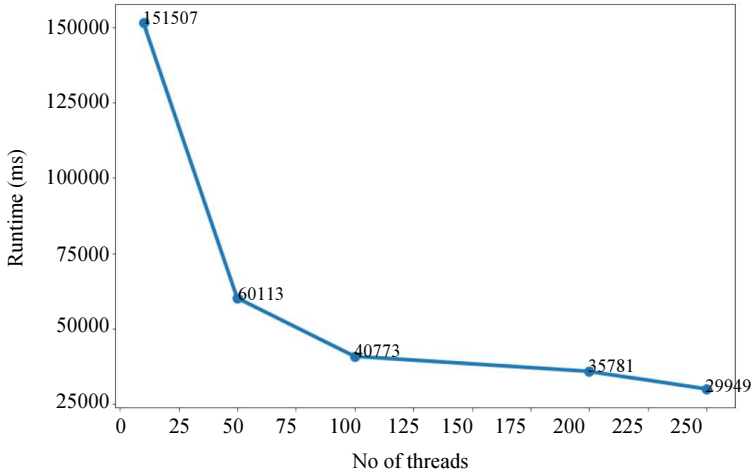


Fig. 7: Runtime analysis for Cristiano Ronaldo trend

Table 2: Details of tweets extracted from datasets

Dataset type	Trend name	No. of unique tweets	Date
Trend	Barcelona	1326	Oct 01 2017
	Jason Aldean	1391	Oct 02 2017
	Kazuo Ishiguro	1840	Oct 05 2017
	Cristiano Ronaldo	794	Oct 23 2017
Stream	–	1000	Apr 16 2014

Unlike the step mentioned in previous section, contexts Generation was carried out for the entire set of tweets and not the equal sized chunks.

Firstly, #CatalanReferendum trend was sampled with generated primary contexts. The dendrogram result is depicted in Fig. 8. Along the same lines we experimented clustering for extended contexts of #CatalanReferendum. Dendrogram for this is shown only for last 10 cluster merges and is given in Fig. 9.

To get insights into the contexts associated, related context labels from dendrogram is printed in user friendly format in Fig. 10a and 10b. As we can see, though a trend is predominately a single context, we observe one cluster showing other categories emerged out of contexts. These clusters formed with primary and extended contexts are comprehensive information about the hidden patterns of the trend.

As a next step, we experimented the same clustering for stream data. Here the objective was to find patterns within the corpus of unrelated tweets. We selected the first 1000 tweets from the stream and Fig. 11 visual representation of the dendrogram. Selected cluster labels of primary and extended contexts for the stream of tweets are provided in Fig. 12a and 12b.

Table 3: Details of Flattened Clusters of (#CatalanReferendum)

No. of Clusters		No. of Items	Full Set or Selected 5 Context Labels from Clusters
Default (3)	Cluster 1	1	Catalonia
	Cluster 2	7	Catalonia, Civil Guard (Spain), Spain, Juvenal, Democracy
	Cluster 3	523	Si La people, S´I Sørv´agur, Ramon Llull, El perro, Baton (law enforcement)
5	Cluster 1	1	Catalonia
	Cluster 2	7	Spain, Democracy, Civil Guard (Spain), Juvenal, Catalan language
	Cluster 3	16	Quebec sovereignty movement, Catalonia, Human rights, Canada, Venezuela
	Cluster 4	5	Spain, Riot police, Catalan language, Catalonia, Barcelona
	Cluster 5	518	CAE Inc., Las Palmas, Coca, Andalusia, Asco (art collective)
7	Cluster 1	1	Catalonia
	Cluster 2	2	Spain, Catalonia
	Cluster 3	7	Juvenal, Catalan language, Spain, Civil Guard (Spain), Democracy
	Cluster 4	16	Democracy, European Union, Venezuela, North Dakota, Quebec sovereignty movement
	Cluster 5	5	Barcelona, Riot police, Catalan language, Catalonia, Spain
	Cluster 6	15	Press TV, Military police, Falange, Spain, Shocker (wrestler)
	Cluster 7	515	Ido (language), Miss Spain, CAMBIA, Derecho, Correcto

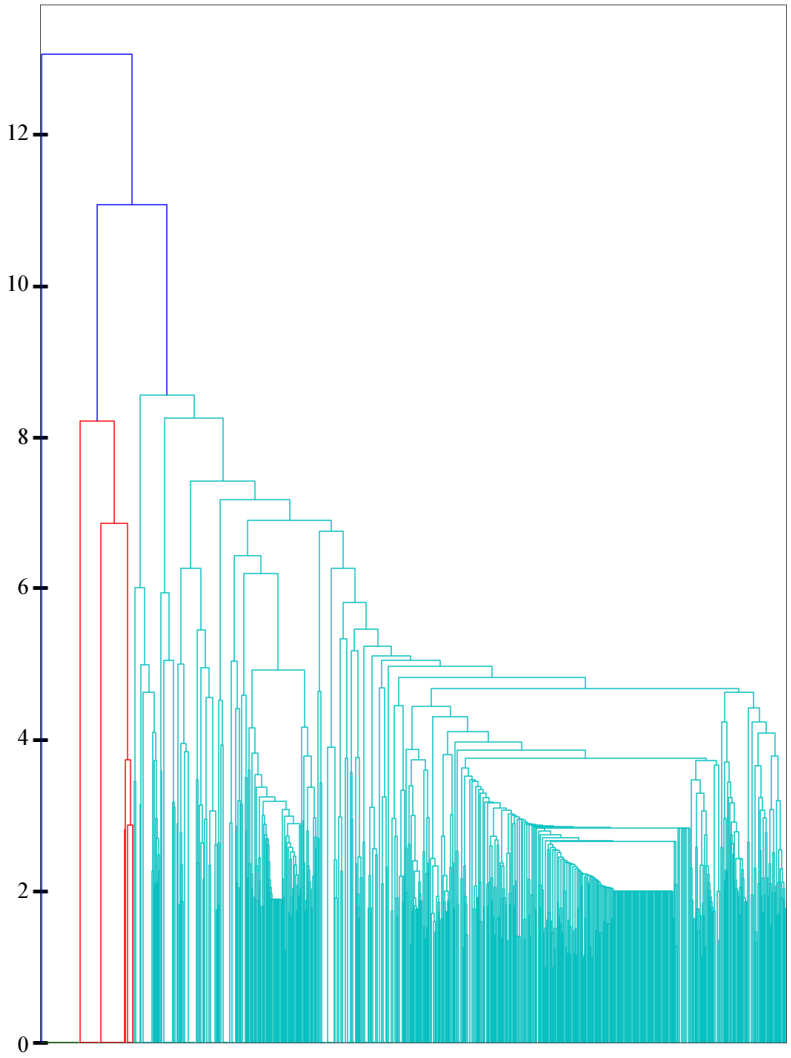


Fig. 8: Dendrogram of all clusters with Primary Contexts for #CatalanReferendum

DOI: 10.3844/jcssp.2020.330.343

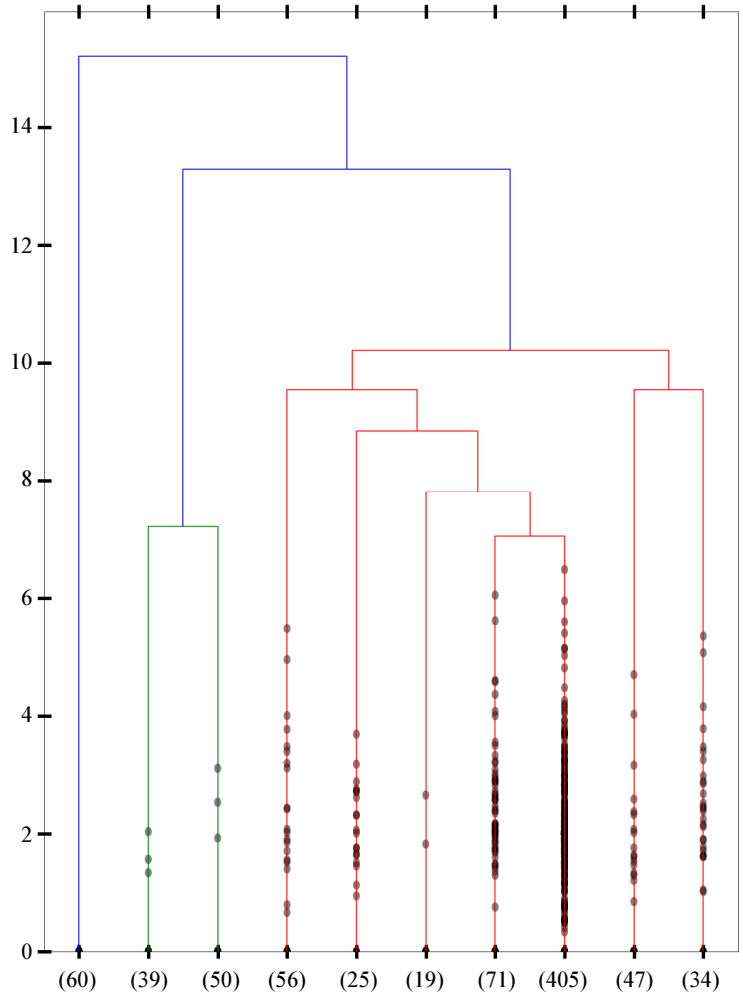


Fig. 9: Dendrogram of Top 10 Merged Clusters with Extended Contexts for #CatalanReferendum

g	{'Catalonia'}
r	{'Catalan_language', 'Catalonia', 'Spain', '4', 'Democracy', 'Civil_Guard_(Spain)', 'Juvenal'}
c	['Vea', 'Llevant', 'La 2 (Spain)', 'Girona', 'Desmond Tutu', 'Ukraine', 'Madrid', 'Press TV', 'Todavía', 'Shocker (wrestler)', 'Burgos CF', 'Pig', 'Lost (TV series)', 'Lisbon', 'Roto', 'Piccadilly Circus', 'Hostiaz', 'Airport', 'Utopia', 'Jacint_Verdaguer', 'Palo_(flamenco)', 'Poca__West_Virginia', 'Force', 'Deben_(unit)', 'Persian_language']

(a)

g	{'StatesAndTerritoriesEstablishedIn1979', 'UnrecognizedOrLargelyUnrecognizedStates', 'NUTS2StatisticalRegionsOfTheEuropeanUnion', 'DividedRegions', 'StatesAndTerritoriesEstablishedIn2006', 'SoloPianoPieces', 'StatesAndTerritoriesEstablishedIn1932', 'RegionsOfEuropeWithMultipleOfficialLanguages'}
r	['CountriesInEurope', 'RomanceCountriesAndTerritories', 'MemberStatesOfNATO', 'StatesAndTerritoriesEstablishedIn1979', 'PeninsulasOfAsia', 'MemberStatesOfTheCouncilOfEurope', 'MemberStatesOfTheUnitedNations', 'UnrecognizedOrLargelyUnrecognizedStates', 'IslandCountries', 'AfricanCountries', 'MemberStatesOfTheUnionForTheMediterranean', 'Continents', 'MemberStatesOfTheEuropeanUnion', 'StatesAndTerritoriesEstablishedIn1932', 'MunicipalitiesInTheProvinceOfBarcelona', 'LawEnforcementUnits', 'IndependenceMovements', 'LawsOfWar', 'MemberStatesOfTheCommonwealthOfNations', 'Spanish-speakingCountriesAndTerritories', 'SoloPianoPieces', 'MunicipalitiesInTheProvinceOfTarragona', 'NUTS2StatisticalRegionsOfTheEuropeanUnion', 'English-speakingCountriesAndTerritories', 'BritishIslands']
c	['GenresOfPoetry', 'AmericanTelevisionTalkShows', 'LanguagesOfTheCaribbean', 'JewishAmericanWriters', 'BladeWeapons', '1980sAutomobiles', 'TreatiesOfCameroon', 'SlavicCountriesAndTerritories', 'WargameCompanies', 'MunicipalitiesInTheProvinceOfValencia', 'Occitan-languageWriters', 'NationalAnthems', 'SpaTowns', '1936Songs', 'FrenchWriters', 'Subject–object–verbLanguages', 'ChileanRockMusicGroups', 'KnightsOfChrist(papacy)', 'EurovisionSongsOf1976', 'Encyclopedists', '18th-centuryFrenchWriters', 'ElectricLocomotivesOfSweden', 'CitiesInScotland', '14th-centuryPhilosophers', 'Idols(TVSeries)Judges']
b	{'StatesAndTerritoriesEstablishedIn1979', 'UnrecognizedOrLargelyUnrecognizedStates', 'NUTS2StatisticalRegionsOfTheEuropeanUnion', 'DividedRegions', 'StatesAndTerritoriesEstablishedIn2006', 'SoloPianoPieces', 'StatesAndTerritoriesEstablishedIn1932', 'RegionsOfEuropeWithMultipleOfficialLanguages'}

(b)

Fig. 10: Selected Clusters Output for #CatalanReferendum Trend; (a) Primary Contexts from the Clusters; (b) Extended Contexts from the Clusters

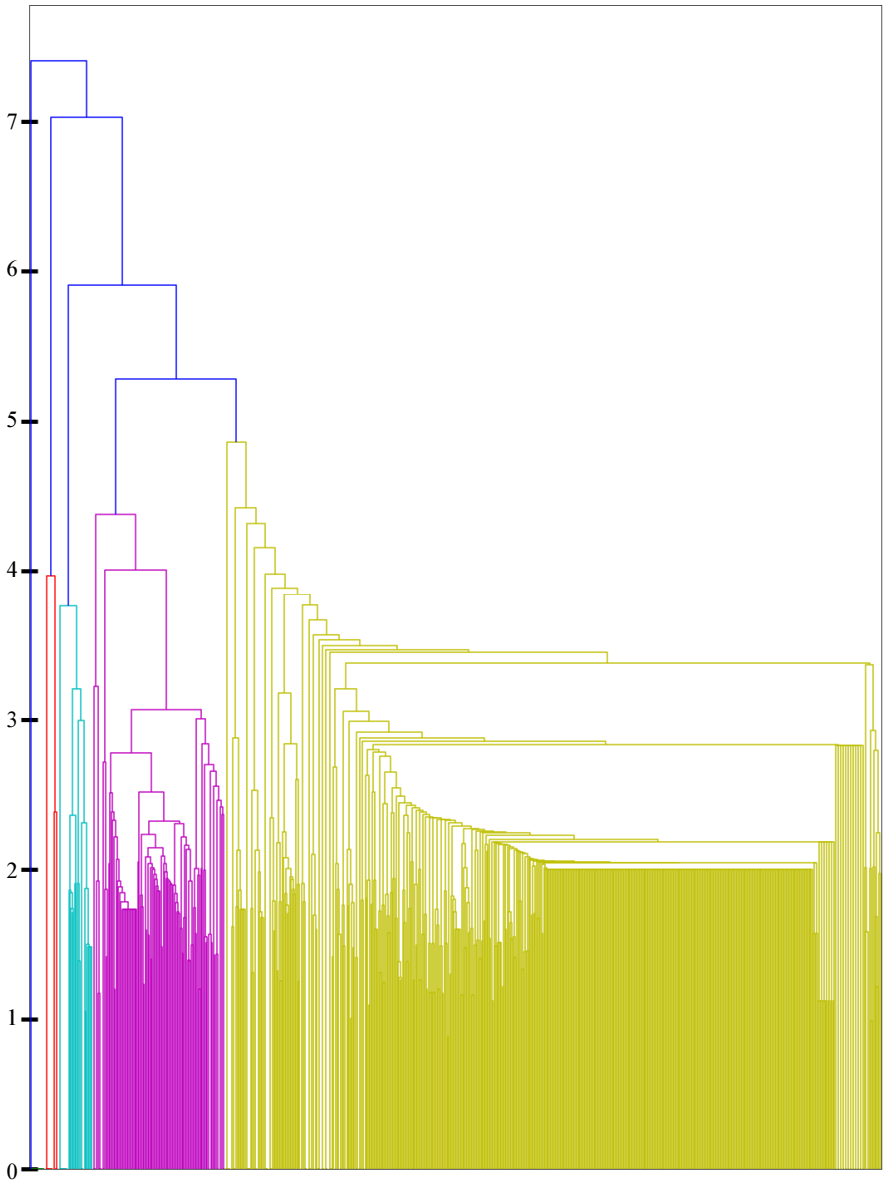
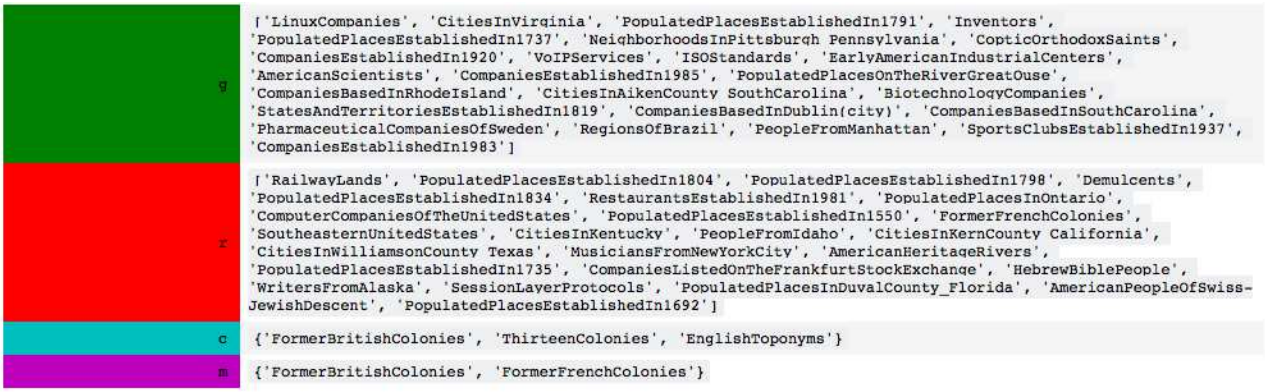


Fig. 11: Dendrogram for Random 1000 Tweets with Primary Contexts

g	{'New_York'}
r	{'Job_(biblical_figure)', 'Starbucks', 'New_York'}
c	{'Ohio', 'OhioHealth', 'Middletown Connecticut', 'Employment', 'Sales', 'Registered nurse', 'Canal Park (Akron Ohio)', 'Trotwood Ohio', 'Highland (council area)', 'Lockbourne Ohio', 'Westlake Ohio', 'Chico California', 'Gahanna Ohio', 'Molina_Healthcare', 'Cleveland', 'Kentucky', 'Maurices', 'Campbell_Hill_(Ohio)', 'Ohio_Stadium'}
m	['Lubbock Texas', 'Boston', 'Registered nurse', 'Wyoming Michigan', 'Guadalajara', 'Maurices', 'Dearborn Michigan', 'TD Bank N.A.', 'Edison New Jersey', 'Bethesda Maryland', 'Indianapolis', 'Columbia Maryland', 'Ashland Kentucky', 'London Ontario', 'Sales', 'Florida State Road 436', 'Delaware North', 'ADTRAN', 'X-ray computed tomography', 'Edinburg Texas', 'Turkish_language', 'South_Carolina', 'Newburgh_(city)__New_York', 'Bros', 'BAYADA_Home_Health_Care']
y	['Freehold Borough New Jersey', 'Tlalpan', 'Reston Virginia', 'Dover', 'Oxford Alabama', 'Peterson Air Force Base', 'Southfield Michigan', 'Kali', 'Tribeca', 'LexisNexis', 'Littlerock__California', 'Sabre', 'Islandia New York', 'Kitten', 'Fort Wayne Indiana', 'Chapultepec', 'Miss Independent (Kelly Clarkson song)', 'Registered nurse', 'Ottawa', 'Rhinelander__Wisconsin', 'Octal', 'Bonsai', 'Buford__Georgia', 'Ridgewood__New_Jersey', 'Golf']

(a)



(b)

Fig. 12: Selected Clusters for Stream Data of 1000 Tweets; (a) Primary Contexts from the Clusters; (b) Extended Contexts from the Clusters

Hierarchical clustering also provides the flexibility of knowing clusters at any of the merged levels. This could be of great use if we have to drill down to specific levels to find clusters. Table 3 displays details of few flattened clusters for #CatalanReferendum trend.

Conclusion and Future Work

In this work, the context modeling framework for tweets and usage of those generated contexts in clustering has been elaborated. Since we use a proven knowledge-base DBpedia which efficiently handles ambiguities in text contexts, we ensured relevant contexts are generated. Similarity scores tabulated above demonstrates the quality of disambiguated primary contexts and extended contexts. The architecture takes care of scalability aspects, handling fairly large datasets in multiple threads. In the second part of this paper, we have shown how this contextual information is useful for uncovering additional information about tweets. We have also presented flattened cluster data from various levels of hierarchy. Many of the existing works focus on specific areas and misses to come up with a comprehensive solution. We wanted to build a generic approach and have presented the same with two different types of tweet datasets. Overall, we have contributed to designing a scalable framework using open source knowledge-base/tools and shown clustering of this contextual data which can be a backbone for specialized problem domains.

In the future, we want to sample this model to cluster users and build a generic recommender system. Accurate users' interests are supportive in designing a strong recommender framework capable of suggesting tweets, topics, users or external contents. Based on the problem domain, we can either pick up primary or extended contexts and configure the number of clusters to address over-recommendation or over-specialization issues. We also intend to test the prototype with real-time streaming data to design an end-to-end framework. We wish to extend the model for a specific domain and compare with existing methods. Different combinations of distance measure and linkage can also be explored.

Author's Contributions

All the authors have contributed to conceptualization, to finalize the approaches, write-up and review of the manuscript. Venkatesha and Prasanth have assisted with the implementation and experimentation of the proposed framework.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and there are no ethical issues involved.

References

Abel, F., Q. Gao, G.J. Houben and K. Tao, 2011. Analyzing user Modeling on Twitter for Personalized News Recommendations. In: User Modeling, Adaption and Personalization, Konstan, J.A., R. Conejo, J.L. Marzo and N. Oliver (Eds.), Springer, Berlin, Heidelberg, pp: 1-12.

Alnajran, N., K. Crockett, D. McLean and A. Latham, 2017. Cluster analysis of twitter data: A review of algorithms. Proceedings of the 9th International Conference on Agents and Artificial Intelligence, Feb. 24-26, Portugal, pp: 239-249. DOI: 10.5220/0006202802390249

Bontcheva, K. and D. Rout, 2012. Making sense of social media streams through semantics: a survey. Semantic Web J., 5: 373-403. DOI: 10.3233/SW-130110

Daiber, J., M. Jakob, C. Hokamp and P.N. Mendes, 2013. Improving efficiency and accuracy in multilingual entity extraction. Proceedings of the 9th International Conference on Semantic Systems, Sept. 4-6, ACM, USA, pp: 121-124. DOI: 10.1145/2506182.2506198

Flisar, J. and V. Podgorelec, 2018. Document enrichment using DBpedia ontology for short text classification. Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics, Jun. 25-27, ACM, Novi Sad, Serbia, pp: 1-9.

Gabrilovich, E. and S. Markovitch, 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. Proceedings of the 20th International Joint Conference on Artificial Intelligence, (CAI' 07), Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp: 6-12.

Genc, Y., Y. Sakamoto and J. Nickerson, 2011. Discovering context: Classifying tweets through a semantic transform based on wikipedia. Proceedings of the International Conference on Foundations of Augmented Cognition, Jul. 9-14, Springer, Orlando, FL, pp: 484-492. DOI: 10.1007/978-3-642-21852-1_55

Hamdan, H., F. Bechet and P. Bellot, 2013. Experiments with DBpedia, WordNet and SentiWordNet as resources for sentiment analysis in micro-blogging. Proceedings of the 2nd Joint Conference on Lexical and Computational Semantics and 7th International Workshop on Semantic Evaluation, (WSE' 13), The Association for Computer Linguistics, Atlanta, Georgia, USA, pp: 455-459.

Hu, X., X. Zhang, C. Lu, E. Park and X. Zhou, 2009. Exploiting wikipedia as external knowledge for document clustering. Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Jun. 28-Jul. 1, ACM, Paris, France, pp: 389-396. DOI: 10.1145/1557019.1557066

Ifrim, G., B. Shi and I. Brigadir, 2014. Event detection in twitter using aggressive filtering and hierarchical tweet clustering. Proceedings of the CEUR Workshop, (EUR' 14), pp: 33-40.

Kapanipathi, P., P. Jain, C. Venkataramani and A. Sheth, 2014. User interests identification on twitter using a hierarchical knowledge base. Proceedings of the 11th International Conference on Semantic Web: Trends and Challenges, May 25-29, Springer, Anissaras, Crete, Greece, pp: 99-113. DOI: 10.1007/978-3-319-07443-6_8

Lehmann, J., C. Bizer, G. Kobilarov, S. Auer and C. Becker *et al*., 2009. DBpedia – a crystallization point for the web of data. J. Web Semant., 7: 154-165. DOI: 10.1016/j.websem.2009.07.002

Lehmann, J., R. Isele, M. Jakob, A. Jentzsch and D. Kontokostas *et al*., 2015. DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. Semantic Web J., 6: 167-195. DOI: 10.3233/SW-140134

Lu, C. and W. Lam, 2012. User modeling and tweets recommendation based on wikipedia concept graph. Intelligent Techniques for Web Personalization and Recommender Systems AAAI Technical Report WS-12-09.

Mendes, P.N., M. Jakob, A. Garcıa-Silva and C. Bizer, 2011. DBpedia spotlight: Shedding light on the web of documents. Proceedings of the 7th International Conference on Semantic Systems, Sept. 7-9, ACM, USA, pp: 1-8. DOI: 10.1145/2063518.2063519

Michelson, M. and S.A. Macskassy, 2010. Discovering users' topics of interest on twitter: A first look. Proceedings of the 4th Workshop on Analytics for Noisy Unstructured Text Data, Oct. 26-26, ACM, Toronto, Ontario, Canada, pp: 73-80. DOI: 10.1145/1871840.1871852

Muñoz García, O., A. García-Silva, O. Corcho, M. de la Higuera Hernandez and C. Navarro, 2011. Identifying topics in social media posts using DBpedia. Proceedings of the Networked and Electronic Media Summit, Sept. 27-29, Torino, Italy, pp: 81-86.

Orlandi, F., J. Breslin and A. Passant, 2012. Aggregated, interoperable and multi-domain user profiles for the social web. Proceedings of the 8th International Conference on Semantic Systems, Sept. 5-7, ACM, Graz, Austria, pp: 41-48. DOI: 10.1145/2362499.2362506

Papneja, S., K. Sharma and N. Khilwani, 2018. Context aware personalized content recommendation using ontology based spreading activation. Int. J. Inform. Technol., 10: 133-138. DOI: 10.1007/s41870-017-0052-5

Pla Karidi, D., 2016. From user graph to topics graph: Towards twitter followee recommendation based on knowledge graphs. Proceedings of the IEEE 32nd International Conference on Data Engineering Workshops, May 16-20, IEEE Xplore Press, Helsinki, Finland, pp: 121-123. DOI: 10.1109/ICDEW.2016.7495629

Pla Karidi, D., Y. Stavrakas and Y. Vassiliou, 2017. Tweet and followee personalized recommendations based on knowledge graphs. J. Ambient Intell. Humanized Comput., 9: 2035-2049. DOI: 10.1007/s12652-017-0491-7

Ramanathan, K. and K. Kapoor, 2009. Creating user profiles using wikipedia. Proceedings of the 28th International Conference on Conceptual Modeling, Nov. 9-12, Gramado, Brazil, pp: 415-427. DOI: 10.1007/978-3-642-04840-1_31

Rao, P.G., M. Venkatesha, A. Kanavalli, P.D. Shenoy and K.R. Venugopal, 2018. A micromodel to predict message propagation for twitter users. Proceedings of the International Conference on Data Science and Engineering, Aug. 7-9, IEEE Xplore Press, Kochi, India. DOI: 10.1109/ICDSE.2018.8527807

Romero, S. and K. Becker, 2017. Improving the classification of events in tweets using semantic enrichment. Proceedings of the International Conference on Web Intelligence, Aug. 23-26, ACM, Leipzig, Germany, pp: 581-588. DOI: 10.1145/3106426.3106435

Saraçli, S., N. Doğan and I. Doğan, 2013. Comparison of hierarchical cluster analysis methods by cophenetic correlation. J. Inequalities Applic. DOI: 10.1186/1029-242X-2013-203

Schuhmacher, M. and S. Ponzetto, 2013. Exploiting DBpedia for web search results clustering. Proceedings of the Workshop on Automated Knowledge Base Construction, Oct. 27-28, ACM, California, San Francisco, USA, pp: 91-96. DOI: 10.1145/2509558.2509574

Shah, B., A.P. Verma and S. Tiwari, 2018. User interest modeling from social media network graph, enriched with semantic web. Proceedings of International Conference on Computational Intelligence and Data Engineering, (IDE' 18), Springer, Singapore, pp: 55-64. DOI: 10.1007/978-981-10-6319-0_5

Szczuka, M., A. Janusz and K. Herba, 2012. Semantic Clustering of Scientific Articles with Use of Dbpedia Knowledge Base. In: Intelligent Tools for Building a Scientific Information Platform, Bembenik, R., L. Skonieczny, H. Rybiński and M. Niezgodka (Eds.), Springer, Berlin, pp: 61-76.

Venkatesha, M., P.G. Rao, A. Kanavalli, P.D. Shenoy and K.R. Venugopal, 2019. Detecting extended contexts in tweets using DBpedia. Proceedings of the IEEE Region 10 Symposium, Jun. 7-9, IEEE Xplore Press, Kolkata, India, pp: 168-173. DOI: 10.1109/TENSYMP46218.2019.8971256

Vicient, C. and A. Moreno, 2015. Unsupervised topic discovery in micro-blogging networks. Expert Syst. Applic., 15: 6472-6485. DOI: 10.1016/j.eswa.2015.04.014