POLYMORPHIC SBD PREPROCESSOR: A PREPROCESSING APPROACH FOR SOCIAL BIG DATA

Amit K. Jadiya

Research Scholar, Institute of Engineering and Technology,
Devi Ahilya Vishwavidyalya, Indore, Madhya Pradesh, India
amitjadiya@gmail.com

Archana Chaudhary

Assistant Professor, School of Computer Science and IT,
Devi Ahilya Vishwavidyalya, Indore, Madhya Pradesh, India
Archana_scs@yahoo.in

Ramesh Thakur

Associate Professor, International Institute of Professional Studies,
Devi Ahilya Vishwavidyalya, Indore, Madhya Pradesh, India
R_thakur@rediffmail.com

Abstract

In recent years, the social media has become a powerful tool for sharing people thoughts and feelings. As a result data is being generated, analyzed and used with a tremendous growth rate. The data generated by numerous updates, comments, news, opinions and product reviews in social websites is very useful for getting insights. As there are multiple sources, the size, speed and formats of the gathered data affects the overall quality of information. To achieve quality information, preprocessing step is very important and decides future roadmap for efficient big data analysis approach. In context to social big data we are addressing the preprocessing phase which includes cleaning of data, identifying noise, data normalization, data transformation, handling missing values and data integration. In this paper we have proposed a new approach polymorphic SBD (Social Big Data) preprocessor which provides efficient results with multiple social big data sets. Also available data preprocessing methods for big data are presented in this paper. After efficient and successful data preprocessing steps, the output data set will be efficient, well formed and suitable source for any big data analysis approach to be applied afterwards. The paper also presents an example case and evaluates min-max normalization, z-score normalization and data mapping for the case presented.

Keywords: Social big data ; Preprocessing ; Data normalization ; Data mapping ; SBD preprocessor.

1. Introduction

Data is basic source of knowledge and in case of big data [16, 17, 18, 19], it travels through four different phases in its life cycle as shown in Fig. 1. These phases are data generation, data acquisition, data storage, and data analytics. There are large number of data sources like various social websites, news contents, blogs and many more, which come under data generation phase. Data collection phase consists of processes and techniques which are useful to extract and gather data from respective data sources. Collected data is preprocessed and stored for further analysis. Due to tremendous data growth and available disparate data sources, a huge amount of structured, semi-structured and unstructured data is generated which is varied, anomalous and complex in nature. Out of these, it is essential that low quality and irrelevant data should be detected and removed as in prior stages in the data generation and acquisition phases in order to avoid wastage of storage space and processing time [2]. Out of mentioned data sources, now a days social media is very popular among people and billions of people all over the world are generating the data. The social media data is a combination of structured, unstructured and semi-structured formats which have 6 V's property - Volume, Velocity, Variety, Veracity, Value and Variability. So it comes under big data and termed as 'social big data' [1].

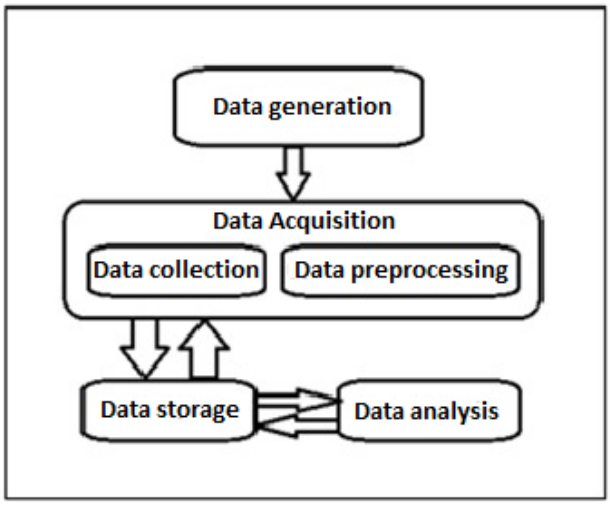


Fig. 1. Big data Lifecycle [2]

Data preprocessing for social media data is a crucial research topic since most of the data is highly impacted by negative elements like presence of noisy data, inconsistent data, missing values and superfluous data. Data preprocessing is a set of techniques and processes which are used prior to the application of data for analysis [2]. In the paper we are combining all these steps and presenting new approach which takes input raw data from multiple sources and integrates it in uniform structured form which can be further used for analytical processes and solutions.

But there are challenges which need to be handled in preprocessing steps while working with Social big data [7]. These challenges are mentioned below.

1.1 Noise and irrelevant data

1.1.1. HTML characters

Initial data might have html tags which are embedded with the original data. For example: > is converted to symbol “>” and " is converted to symbol for double quotation mark.

1.1.2. Different data formats

Converting information from complex characters and symbols to the simple and easy understandable characters is known as data decoding. Normally text data is available in different forms Latin, UTF8, UTF16 etc. and for further analysis phases it is important to keep all data into one standard encoding format.

1.1.3. Apostrophe lookup

When apostrophes are present in the text, the chances of disambiguation also increases. It needs to enable proper structured format and follow the rules of context free grammar in order to get the correct mean. Example “it’s" should be "it is" or "it has”.

1.1.4. Removal of stop-words and punctuation

Commonly used words (like a, an, the etc.) which should be ignored for enterprise searching and indexing are called stop words. These are natural language words which do not have significant meaning, so these words can be ignored. Approaches like creating a list of stop words and using predefined specific libraries available in programming language can be used. Also the punctuation symbols need to be dealt properly according to the priorities. For example: ",", “.” and ”?” are important punctuation which should be retained and other punctuation symbols should be removed.

1.1.5. Removal of expressions

In social media, normally the textual data contains human expressions for laughing, crying, happy, sad etc. Usually these expressions are not relevant and can be removed. This can be done by normal regular expressions.

1.1.6. Split attached words

Normally social data have the text data, which might be informal. Many comments and tweets are posted with few attached words like ColdDay, PlayingInTheRainnySeason etc. Such attached words need to split into normal forms.

1.1.7. Removal of URLs

It is a very common practice to remove URLs and hyperlinks in text data.

1.2 Handling missing values

Real-world data can have missing values due to various reasons like few data field values not recorded due to unavailability or non applicability. Values might be missed due to data corruption. For the social media data it might be possible that number of likes or dislikes etc. are not available for few sources. So handling the missing data in the initial phase is important.

1.3 Data normalization

As there are multiple sources, data may contain attribute values with different scales like rupee or dollar for currency, grams or kilograms for weight, centimeters or meters for length etc. Rescaling real numeric attribute values into the same scale with range of 0 to 1 is called normalization.

1.4 Data transformation

The process of converting the data into a standard format is known as data transformation. Data transformation primarily involves mapping of source data elements for the respective destination.

1.5 Data integration

The goal of data integration is to provide unified access to data which resides in multiple, autonomous data sources. While this goal is easy to state, achieving this goal has proven notoriously hard, even for a small number of sources that provide structured data. For the scenario of unstructured data, it is more challenging [4].

2. Social big data issues

Big data processing differs from traditional data processing due to its dimensions volume, velocity, variety, veracity, value and variability. In case of social big data, the first issue is that data sources not only contains huge volume of data, but the number of data sources are also in thousands. Second, most of the data sources are very dynamic and growing with a tremendous rate which makes it different from traditional data. Third, data sources are very heterogeneous in nature. Fourth, the data sources have different qualities and significant differences in the area coverage, timing and accuracy of collected data from the sources [4, 20].

To overcome above mentioned issues, the data quality in social big data needs well-defined structure which can be accessed by lightweight processes with the capability to run in parallel where its storage system is distributed [2]. In section-3, we presents related work and section-4 portray proposed Polymorphic SBD preprocessor algorithm. Also we presented an example case which shows the challenges with social big data preprocessing and efficient approaches for their solutions using proposed algorithm.

3. Related work

Huadong et al. [10] have proposed and implemented a big data preprocessing system based on Hadoop which had four modules: resource monitoring, task distributing, task processing and input analysis. Authors have explained the process using the word count example where in the map phase, all nodes read the input data files line by line and split each line into words by using a specific delimiter. Then each node prepares key/value pair which is stored in temporary file. In the reduce phase, it summarize the partial result of the map phase and counts each word's total frequency.

Rachida et al. [2] have proposed a big data processing quality framework for solving various data quality issues in large data sets. The key components of suggested framework are preprocessing activity selection, technique selection, data quality profile optimization, execution, quality control and data quality profile adapter. They have used EEG dataset as big data since it fulfills most of the big data characteristics and this raw data was preprocessed to remove filtrated noise and artifacts. The selected DQP (Data Quality Profile) is generated as an XML file which contains all the information related to the preprocessing activity to EEG algorithms and the targeted data quality.

Jin et al. [11] have proposed an efficient multi dimensional fusion algorithm. Authors have partitioned the big data with higher dimensions into a number of relatively small data subsets. In the suggested algorithm, the preprocessing block is responsible for the two tasks - first is to normalize the data so that the data can be easily compared. Second is to replace the missing values with "*" to successfully process the missing data in the fusion algorithm.

Arputhamary et al. [12] highlighted the importance of data integration in big data world and considered two categories in big data integration. First is integration of multiple big data sources in the big data environment. Second is the integration of unstructured big data sources. They have suggested techniques for meeting big data integration challenges like schema mapping, record linkage and data fusion.

Luai et al. [13] proposed a matrix to check for the best normalization method based on the factors and their priorities. They have applied normalization by using three methods: min-max normalization, z-score normalization and normalization by decimal scaling and provided comparative analysis. After analysis on their results, they have suggested that choosing the min-max normalization provides better normalized dataset. They have worked with HSV data set having 122 examples, which was downloaded by UCI repository.

Other than above, many researchers worked with the tools like R, Python, Scala, Apache Spark, Apache Hive and Apache Pig which are helpful in achieving goals of preprocessing tasks.

4. Methodology

In this paper, we propose a new approach Polymorphic SBD preprocessor which provides efficient preprocessed output dataset and we can perform efficient analytical solutions in further steps. We consider there are total n different social media data sources S = {S_1, S_2, S_3, ..., S_n} and collected data set D = {D_1, D_2, D_3, ..., D_n}, where D_1 is data set collected from social media source S_1 and so on. D_1 has F_1 = {F_{1a}, F_{1b}, F_{1c}....}, D_2 has fields F_2 = {F_{2a}, F_{2b}, F_{2c}....} and so on. As set F = {F_1, F_2, F_3, ..., F_n} have different attributes, scale values etc. before moving further to data analysis, preprocessing is utmost important. Data sets D_1, D_2, D_3,...., D_n should not have noisy data, missing values etc. and data normalization, transformation and mapping need to be done before data Integration. We consider output preprocessed data sets PD = {PD_1, PD_2, PD_3, ..., PD_n}. The pseudo-code of Polymorphic SBD preprocessor approach is shown in Algorithm 1.

Algorithm 1. Polymorphic SBD preprocessor.

Input : $S = \{S_1, S_2, \ldots, S_n\}$ // Set of social media data for n sources.

Output: $PD = PD_1 \cup PD_2 \cup \ldots\ldots \cup PD_n$ // Aggregated pre-processed dataset for n sources

Method:

step 1. Extract data from different sources as
$D_1 \leftarrow S_1, D_2 \leftarrow S_2, \ldots . D_n \leftarrow S_n$.

step 2. ***for*** $i=1$ *to* n ***do***
- **2.1.** ***if*** D_i contains noisy data and missing values ***then***
  - **2.1.1.** Perform data cleaning, noise removal, handle missing value approaches and store result in DC_i.
  - ***end***
- **2.2.** From D_i and DC_i, identify set of fields. $F_i = \{F_{i1}, F_{i2}, \ldots, F_{ip}\}$ // p is number of fields which need to be normalized, transformed and integrated.
- **2.3.** ***for*** $j=1$ *to* p ***do***
  - **2.3.1** On F_{ij} perform required approach min-max/z-score normalization, transformation or data mapping.
  - ***end***
- **2.4.** Store result in PD_i // Preprocessed data

end

step 3. Integrate $PD = PD_1 \cup PD_2 \cup \ldots\ldots \cup PD_n$.

4.1 Design and functionality of Polymorphic SBD preprocessor

The proposed approach is used to prepare preprocessed dataset PD which contains integrated uniform data from multiple n sources. The design of Polymorphic SBD preprocessor approach is shown in Fig. 2. First, extract data D from source S then identify noisy data and missing values by applying approaches as discussed in section 5.2. Cleaned dataset DC is prepared after removal of noisy and missing values and prepare a set of all fields F which need to be normalized, transformed and integrated for final preprocessed dataset PD. As these fields F_1, F_2, . . .F_n will not be in unformed structure which can be integrated, apply suitable normalized approach min-max or z-score normalization, transformation and data mapping approaches and save result data set as PD. These approaches are discussed in following sections. After completing above processes integrate result dataset for all sources.

5. Data set preparation and steps for proposed approach

For our study, in data preparation section we have selected social media dataset D_1 from Facebook, which is downloaded by website data.world.com [15]. The proposed approach involves data normalization, data transformation and data integration which are discussed in the following sections. It is also required to use few

common preprocessing steps like noise identification, data cleaning and handling missing values which can be achieved by using available packages and functions in programming languages.

5.1 Preparation of data

For making benefits of data availability for researchers, social media companies provide their Application Programming Interfaces (APIs). An API is an interface through which we may connect new add-ons to the available services and collect data of a given social media service for further work or analysis [5]. There are two types of APIs. First REST API, which work as request-response fashion. Here Client sends the request to server and server responds. Second is streaming API where server sends response to the client whenever any updates are available.

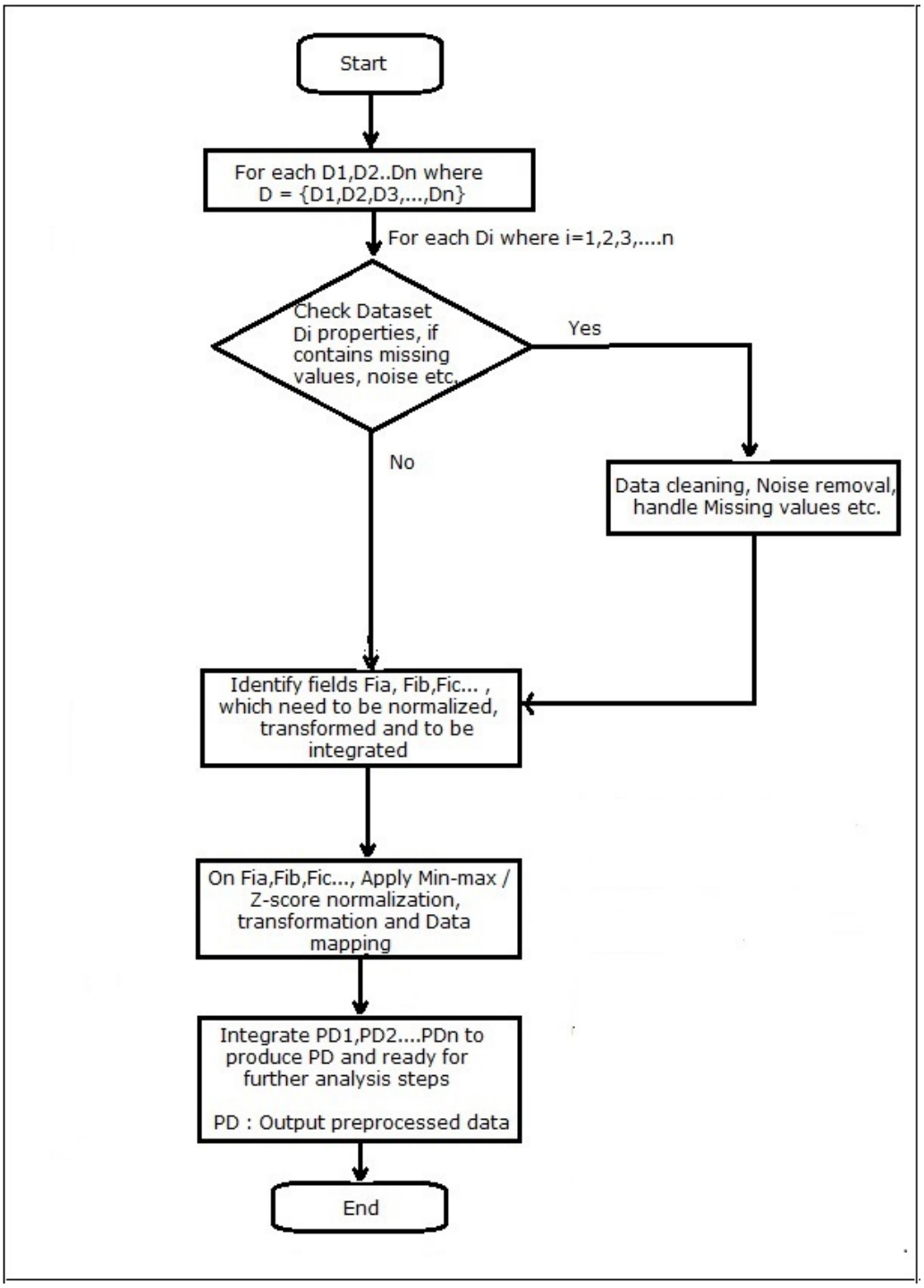
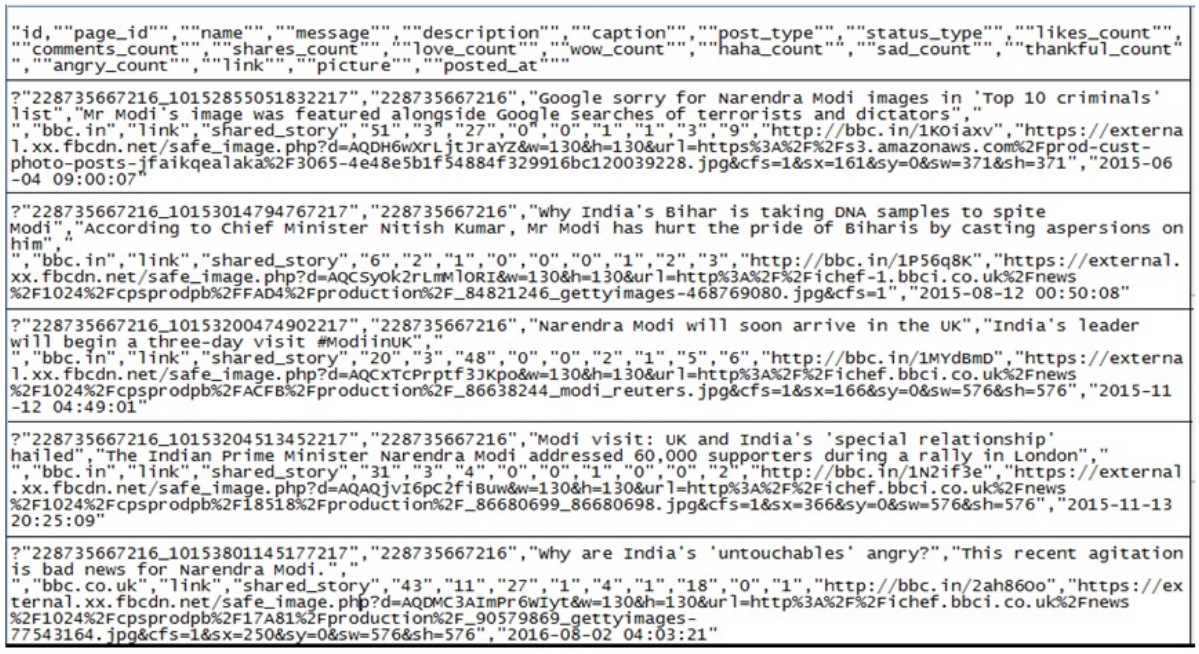


Fig. 2. Design of Polymorphic SBD preprocessor

As shown in Fig. 3, we are taking dataset from Facebook which have attributes: “Id”, "page_id", "name", "message","description", "caption","post_type", "status_type", "likes_count", "comments_count", "shares_count", "love_count", "wow_count", "haha_count", "sad_count", "thankful_count“, "angry_count“, "link", "picture“ and "posted_at“. Process of getting Facebook data through API is shown below.

(1) Generate access token from Facebook for developers.
(2) Import libraries related to Facebook like import com.restfb.FacebookClient.
(3) Use the access token to get field values.
(4) Print the values.



```
"id,""page_id"","name"",""message"","description"","caption"",""post_type"","status_type"",""likes_count"",
""comments_count"","shares_count"",""love_count"","wow_count"",""haha_count"","sad_count"",""thankful_count"
","angry_count"","link"",""picture"","posted_at"""
?"228735667216_10152855051832217","228735667216","Google sorry for Narendra Modi images in 'Top 10 criminals'
list","Mr Modi's image was featured alongside Google searches of terrorists and dictators","
","bbc.in","link","shared_story","51","3","27","0","0","1","1","3","9","http://bbc.in/1KOiaxv","https://externa
l.xx.fbcdn.net/safe_image.php?d=AQDH6WXrLjtJraYZ&w=130&h=130&url=https%3A%2F%2Fs3.amazonaws.com%2Fprod-cust-
photo-posts-jfaikqealaka%2F3065-4e48e5b1f54884f329916bc120039228.jpg&cfs=1&sx=161&sy=0&sw=371&sh=371","2015-06
-04 09:00:07"
?"228735667216_10153014794767217","228735667216","Why India's Bihar is taking DNA samples to spite
Modi","According to Chief Minister Nitish Kumar, Mr Modi has hurt the pride of Biharis by casting aspersions on
him","
","bbc.in","link","shared_story","6","2","1","0","0","0","1","2","3","http://bbc.in/1P56q8K","https://external.
xx.fbcdn.net/safe_image.php?d=AQCSyOk2rLmMlORI&w=130&h=130&url=http%3A%2F%2Fichef-1.bbci.co.uk%2Fnews
%2F1024%2Fcpsprodpb%2FFAD4%2Fproduction%2F_84821246_gettyimages-468769080.jpg&cfs=1","2015-08-12 00:50:08"
?"228735667216_10153200474902217","228735667216","Narendra Modi will soon arrive in the UK","India's leader
will begin a three-day visit #ModiinUK","
","bbc.in","link","shared_story","20","3","48","0","0","2","1","5","6","http://bbc.in/1MYdBmD","https://externa
l.xx.fbcdn.net/safe_image.php?d=AQCxTcPrptf3JKpo&w=130&h=130&url=http%3A%2F%2Fichef.bbci.co.uk%2Fnews
%2F1024%2Fcpsprodpb%2FACFB%2Fproduction%2F_86638244_modi_reuters.jpg&cfs=1&sx=166&sy=0&sw=576&sh=576","2015-11
-12 04:49:01"
?"228735667216_10153204513452217","228735667216","Modi visit: UK and India's 'special relationship'
hailed","The Indian Prime Minister Narendra Modi addressed 60,000 supporters during a rally in London","
","bbc.in","link","shared_story","31","3","4","0","0","1","0","0","2","http://bbc.in/1N2if3e","https://external
.xx.fbcdn.net/safe_image.php?d=AQAQjvI6pC2fiBuw&w=130&h=130&url=http%3A%2F%2Fichef.bbci.co.uk%2Fnews
%2F1024%2Fcpsprodpb%2F18518%2Fproduction%2F_86680699_86680698.jpg&cfs=1&sx=366&sy=0&sw=576&sh=576","2015-11-13
20:25:09"
?"228735667216_10153801145177217","228735667216","Why are India's 'untouchables' angry?","This recent agitation
is bad news for Narendra Modi.","
","bbc.co.uk","link","shared_story","43","11","27","1","4","1","18","0","1","http://bbc.in/2ah860o","https://ex
ternal.xx.fbcdn.net/safe_image.php?d=AQDMC3AImPr6WIyt&w=130&h=130&url=http%3A%2F%2Fichef.bbci.co.uk%2Fnews
%2F1024%2Fcpsprodpb%2F17A81%2Fproduction%2F_90579869_gettyimages-
77543164.jpg&cfs=1&sx=250&sy=0&sw=576&sh=576","2016-08-02 04:03:21"
```

Fig. 3. Example case Data Set from FacebookAPI [15].

5.2 Noise identification, data cleaning and handling missing value

For handling HTML characters, decoding data, Apostrophe lookup, removing stop-words, punctuation, expressions and URLs, it can be directly deleted by using appropriate packages and modules available in programming language (like html parser in python). To handle the missing values, there are few methods which can be used for social big data as well. First is to remove all rows which contain missing values [8]. This is the simplest way to handle missing data and can be done by using utility packages in programming like use Pandas DataFrame in Python. Second method is to impute missing values [8]. This is to replace missing values with sensible values. Imputing means using a model to replace missing values. We hereby consider few options where replacing a missing value is needed. These are –

(1) A constant value for which values are already defined within the field domain but having a distinct value from all others such as 0.
(2) A value which is selected randomly on the basis of other column values.
(3) A mean, median or mode value for the attributes or columns.
(4) Any value derived from any other predictive model.

There are few algorithms which can accept missing data as well. For example, in such cases k-nearest neighbor’s algorithm can ignore a column from a distance measure. Also there are few algorithms like classification and regression trees, which can use the missing value as a unique and different value for building the predictive model.

5.3 Data normalization

Main data normalization methods are min-max normalization and z-score normalization [13]. Min-max normalization implements a linear transformation to the real data and it is calculated as -

$$V' = ((V - Min_p) / (Max_p - Min_p)) * (NewMax_p - NewMin_p) + NewMin_p \quad (1)$$

Where Min-max normalization maps a value V to V’ in the boundary [NewMaxp to NewMinp].

In z-score normalization, values are normalized on the basis of mean and standard deviation of the attribute [13]. It is also called zero mean normalization and it is calculated as -

$$d' = d - Mean(P) / Std(P) \quad (2)$$

Where z-score normalization find the z-score value d' of the value d. Here Mean(P) is mean of all attribute values of P and Std(P) is calculated standard deviation of all values of P [6].

Each source has different parameters like Facebook have "likes_count", "comments_count" etc. and Twitter has "followers_count". So before integrating data, normalization is important. Table 1 shows min-max normalization and table 2 shows z-score normalization approach on likes_count and angry_count fields in sample dataset discussed in section-5.1. Here "**..**" represents many other fields.

Table 1. Min-max normalization on likes_count and angry_count fields.

..	..	**likes_ count**	**angry_ count**	..	..
..	..	51	9	..	..
..	..	06	3	..	..
..	..	20	6	..	..
..	..	31	2	..	..
..	..	43	1	..	..

Min-max normalization

..	..	**likes_ count**	**angry_ count**	..	..
..	..	1	1	..	..
..	..	0	0.29	..	..
..	..	0.33	0.71	..	..
..	..	0.57	0.14	..	..
..	..	0.83	0	..	..

Table 2. Z-score normalization on likes_count and angry_count fields.

..	..	**likes_ count**	**angry_ count**	..	..
..	..	51	9	..	..
..	..	06	3	..	..
..	..	20	6	..	..
..	..	31	2	..	..
..	..	43	1	..	..

Z-score normalization

..	..	**likes_ count**	**angry_ count**	..	..
..	..	1.3	0.56	..	..
..	..	-1.5	-0.14	..	..
..	..	-0.64	0.21	..	..
..	..	0.05	-0.26	..	..
..	..	0.8	-0.37	..	..

Although the data is being collected from different sources as presented in SBD preprocessor, obtained resulting values from min-max and z-score normalization are efficiently used for social big data. It provides uniform values having meaningful information whether data is stored to distributed clusters.

5.4 Data transformation

The phases of data transformation are data mapping and code generation [9]. Data mapping is to set elements from the source to the destination for capturing occurred transformations. This becomes more tedious task when transformations are complex like one-to-many or many-to-one. Different sources have different fields and characteristics like Facebook D_1 and twitter data D_2, which can be mapped as Fig. 4. Code generation is to create the program for actual transformation logic. The data specification is used to develop the program and the program executes on the computer system.

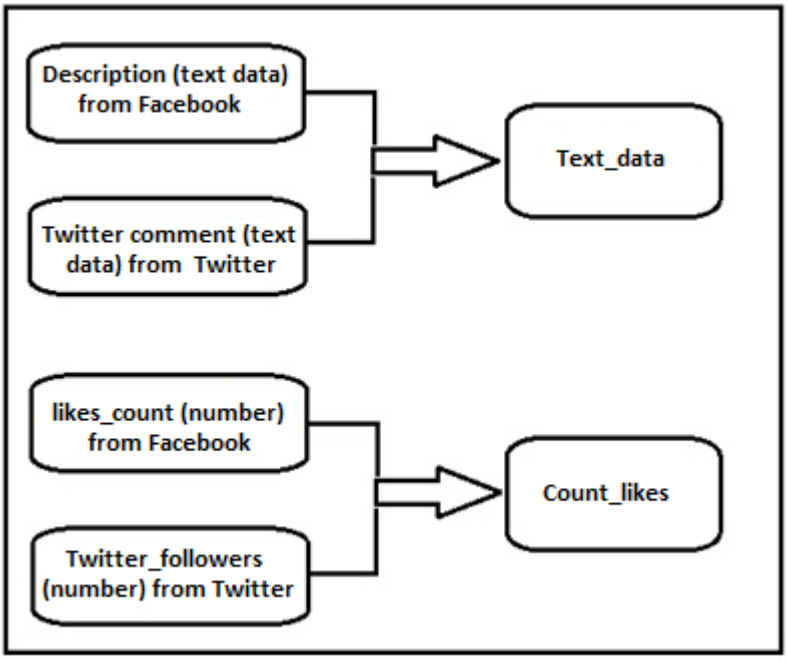


Fig. 4. Data mapping.

5.5 Data integration

To integrate multiple autonomous data sources and removing data ambiguity, it is quite difficult which need manual effort to get the semantics of the data for each source. As shown in Fig. 5, there are three major steps in traditional data integration: Schema alignment, Record linkage and Data fusion.
The aim of schema alignment is to find out which attributes have the same meaning and which have different meanings. Semantic ambiguity is the main challenge which is addressed by schema alignment. Record linkage is the task of identifying records that have similar meaning across various data sources [3]. When different sources provide conflicting values, which value can be used in integrated data, this challenge is addressed by data fusion. Data fusion is to combine and correlate data which belongs to a single subject from different sources. This achieves deriving additional insights from the data [4, 9].

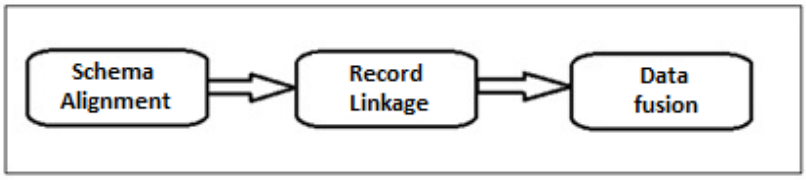


Fig. 5. Data Data Integration Architecture[12].

6. Results and discussion

Proposed approach produce efficient integrated dataset from multiple social media sources. For implementation of modules like normalization and data mapping in proposed approach, we have used packages/functions provided in programming language like Python and Java. Currently we have presented SBD preprocessor steps with dataset by considering few fields or properties of social media source. As the result, output dataset PD (Preprocessed dataset) is a uniform, well formed and structured data set which shows that SBD preprocessor is a efficient approach and very important for further analysis steps. Later when we will work with many sources and fields, we will extend the implementation with Hadoop MapReduce based algorithms for preprocessing. Hadoop MapReduce is a software framework for distributed processing which can work efficiently in parallel fashion.

7. Conclusion

This paper presented new efficient approach through Polymorphic SBD preprocessor which provides solution approaches for preprocessing in context to social big data. The result shows that proposed algorithm provides

output data with high accuracy and aggregated features, which is best source for further data processing applications. Here authors worked with data collected by limited source and fields only, but the same approach can be applied with multiple sources and fields. Further we will extend our work on more different types of data sources with parallel processing and system based on Hadoop and analyze the result.

References

[1] Bello-Orgaz, G.; Jung, J; Camacho, D. (2015). Social big data: Recent achievements and new challenges. Information Fusion, pp. 1-15.
[2] Taleb, I.; Dssouli, R.; Serhani, M.A. (2015): Big Data Pre-Processing: A Quality Framework. 2015 IEEE International Congress on Big Data, pp. 191-198.
[3] Arputhamary, B.; Arockiam, L. (2014): A Review on Big Data Integration. IJCA Proceedings on International Conference on Advanced Computing and Communication Techniques for High Performance Applications ICACCTHPA, pp. 21-26.
[4] Dong, X.L.; Srivastava, D. (2013): Big Data Integration. Proceedings of the 2013 IEEE International Conference on Data Engineering, pp. 1245-1248.
[5] Lomborg, S.; Bechmann, A. (2014): Using APIs for Data Collection on Social Media. The Information Society 30, pp. 256-265.
[6] Saranya, C.; Manikandan, G. (2013): A study on Normalization Techniques for Privacy Preserving Data Mining. International Journal of Engineering and Technology 5, pp. 2701-2704.
[7] García, S.; Ramírez-Gallego, S.; Luengo, J.; Benítez, J.M.; Herrera, F. (2016): Big data preprocessing: methods and prospects. Big Data Analytics 1(9), pp. 1-22.
[8] Jason Brownlee (2017) Python Machine Learning para. 5. https://machinelearningmastery.com/handle-missing-data-python/. Accessed: 10 Oct 2020.
[9] Huh, J.; Grundy, J.; Hosking, J.; Liu, K.; Amor, R. (2009): Integrated Data Mapping for a Software Meta-tool. Proceedings of the 2009 Australian Software Engineering Conference, ASWEC, pp. 111-120.
[10] Dai, H.; Zhang, S.; Wang, L.; Ding, Y. (2016): Research and Implementation of Big Data Preprocessing System Based on Hadoop. 2016 IEEE International Conference on Big Data Analysis (ICBDA), pp. 1-5.
[11] Zhou, J.; Hu, L.; Wang, F.; Lu, H.; Zhao, K. (2013): An Efficient Multidimensional Fusion Algorithm for IoT Data Based on Partitioning. Tsinghua Science and Technology 18(4), pp. 369-378.
[12] Arputhamary, B.; Arockiam, L. (2015): Data Integration in Big Data Environment. Bonfring International Journal of Data Mining 5: pp. 1 - 5.
[13] Shalabi, L.A.; Shaaban, Z. (2006): Normalization as a Preprocessing Engine for Data Mining and the Approach of Preference Matrix. International Conference on Dependability of Computer Systems, pp. 207-214.
[14] Bhadani, A.K.; Jothimani, D. (2016): Big Data: Challenges, Opportunities and Realities. Effective Big Data Management and Opportunities for Implementation, Pennsylvania, USA, IGI Global, pp. 1-24.
[15] Martinchek (2016). 2012-2016 Facebook Posts Retrieved from https://data.world/martinchek/2012-2016-facebook-posts
[16] Chaudhary, A.; Kolhe, S.; Kamal, R. (2016): A hybrid ensemble for classification in multiclass datasets: An application to oilseed disease dataset. Computers and Electronics in Agriculture 124, pp. 65–72.
[17] Chaudhary, A.; Kolhe, S.; Kamal, R. (2016): An improved random forest classifier for multi-class classification. Information Processing in Agriculture 3, pp. 215–222.
[18] Chaudhary, A.; Kolhe, S.; Kamal, R. (2013): Machine learning classification techniques: A comparative study. International Journal on Advanced Computer Theory and Engineering 2(4), pp. 21–25.
[19] Chaudhary, A.; Kolhe, S.; Kamal, R. (2013): Machine Learning Techniques for Mobile Devices: A Review. International Journal of Engineering Research and Applications 3(6), pp. 913–917.
[20] Jadiya, A.K.; Thakur, R. (2019): Efficient Workflow for Social Big Data Processing. Proceedings of Recent Advances in Interdisciplinary Trends in Engineering & Applications (RAITEA).