A NOVEL SIMILARITY APPROACH FOR ONLINE SENTIMENT TEXT

Zar Zar Hnin[1,*], Ei Ei Mon[2], Cho Cho Khaing[3]

[1] *Faculty of Computer Science, University of Computer Studies, Mandalay, UCS-MDY, Myanmar*
[2,3] *Faculty of Computer Science, University of Computer Studies (Loikaw), Kayah State,Myanmar*

ABSTRACT

The content and context of social network websites become crucial to know what the people are interested in and what kinds of information are spread among them depending on all commissions, comments, and actions need to analyze. Consequently, it becomes important that the brands listen carefully to what is said about their online business. Additionally, it demands more challenges to know whether the conversation leads to positive or negatives so that the impact of social network opinions can be measured to apply back in the real word problems. This paper finds the similar groups of social network activities, especially comments and posts of the users who shares about the same context depending on a specific topic. For this purpose, this paper introduces how to deal with finding the similarity between the contextual text of the users in semantic ways by filling the gap of syntactic measures in text similarity. Regarding the datasets, Twitter dataset, which is a popular dataset for sentiment analysis is used Respecting to the performance results, the proposed system achieves promising results with higher accuracy rate but lower error rate for both datasets available from online.

Keywords: *social network, text similarity, semantic analysis, syntactic measures*

1. INTRODUCTION

Sentiment analysis is also useful when analysis the opinions of human by extracting features keywords of user input text. [1-5]. In a world where we generate 2.5 quintillion bytes of data every day, sentiment analysis has become a key tool for making sense of that data. This has allowed academic and business area to get key insights and automate all kind of processes.

Investigation the belief of social network users as a social-learning agenda is an important part of any social media monitoring plan so that social network websites help the users finding the information they want to receive and inform the advertisements or useful jobs vacancies depending on the information they browse in SNS. For example, one of popular social network websites, Facebook watches the activities of social network users depending on the posts or comments they write, they attract the users to provide more information they need. To do so, they first need to understand what a person feels behind a social media post. Having known the opinions extracted from a post can give us an important matter regarding how we can continue and respond [6, 7, 12].

Sentiment Analysis, which is alternatively known as Opinion Mining is a field within Natural Language Processing (NLP) that builds systems that try to identify and extract opinions within text such as positive, negative or neutral sense from the user's social network sharing and posts.

Similarity measures play a key role in classification of texts in several fields, including signal processing, natural language processing, statistics and information retrieval and also sentiment analysis. This measure is needed to retrieve the documents relevant to user query.

Regardless of querying final sentiment decisions, this paper perform the analysis procedures for similarity measures between comments groups of different users so that they can further be used for further analysis by corresponding business area such as public policy making, online shopping, etc.[12,13,16,18].

Sentiment analysis, just as many other NLP problems, can be modeled as a classification problem where two sub-problems must be resolved:

- Classifying a sentence as subjective or objective, known as subjectivity classification.

- Classifying a sentence as expressing a positive, negative or neutral opinion, known as polarity classification

This paper digs a deep hole for proposing a new text similarity measures by means of phrase-wise and word-wise similarity to reveal the similar groups of writing styles with different word usage and styles. We even explore embedded words using the help of WordNet so that we can explore more similarity by semantically rather than syntactic matching in contemporary matching processes.

The remainder of the paper is organized as follows. Literature review is studied in Section 2, problem domain is discussed in section 3. The proposed method is presented in Section 4 and the conclusion is presented in Section 5.

2. LITERATURE REVIEW

The earlier research reports focus on consideration of similar patterns or semantics information among documents, concepts or phrases. In recent natural language processing applications, they demonstrate the stronger need to find effective methods to measure semantic similarities between variable length texts, and general methods are suggested for these people [1-5].

Sentiment lexicons can be organized in three types, attending to which information is contained in them [5], [6]: (i) those who contain only sentiment words (a list of words), (ii) the ones that are formed by both sentiment words and polarity orientations (a list of words with only positive and negative annotations), and (iii) the lexicons that offer sentiment words with orientation and intensity [7] (a list of words with scalar numerical values).

The most popular approach that makes use of sentiment lexicons is keyword matching, also called keyword spotting [8]. Basically, this technique consists in detecting the presence of certain sentiment bearer words, thus obtaining the sentiment estimation as an aggregate of the associated sentiment values. Although this method is certainly simple and computationally cheap, it is also limited, as it happens in the case of domain adaptation. Sentiment words can display variations in their polarity values depending on the contextual domain [9], language [10] or even context [11], causing lexicon-based approaches to decrease their classification performance.

Checking that semantic analogy texts provides the correct path to understand, compare and study the concepts that are subject to each term in the test texts for both of them. The words in two different test texts may not necessarily mean the same concepts. The correct concept of word mapping and word comprehension is required according to the study [2] with the use of evaluation text refers to rating predictions.

Some of the existing works [3] and [4] use texts of change by conducting an assessment of the user's opinion that is reflected in their texts of change, to improve personalized recommendations. Only a few methods combine the rules of character level and token in the researches [5,6,7,8] These methods are called important steps. According to the study [7] soft cosine, both the character level and the token level appear to match the name. Soft cosine has cosine to match tokens and bigrams to match the character level.

Regarding the chain-based similarity, the paper [8] proposed a standardized and modified version of the string-matching algorithm of the Longest Common Undercurrent (LCS) to measure the similarity of the text. It works together with a corpus-based measure, its methods achieved a very competitive result.

3. PROBLEM DOMAIN

3.1 Sentiment Analysis

The AI system should understand similar identifiers from users and provide a consistent response. The emphasis on semantic objectives is to create a system that identifies language and word patterns to generate responses similar to how human conversations work. This paper organizes the similar user group for online analyzers so that they can make further decision in a timely manner without needing individual comparison on every online comments of the social newt wok users to extract the following information.

Polarity: if the speaker expresses a positive or negative opinion,

Subject: the thing that is being talked about,

Opinion holder: the person, or entity that expresses the opinion

With the help of sentiment analysis systems, this unstructured information could be automatically transformed into structured data of public opinions about products, services, brands, politics, or any topic that people can express opinions about. This data can be very useful for commercial applications like marketing analysis, public relations, product reviews, net promoter scoring, product feedback, and customer service.

Sentiment analysis can be applied at different levels of scope:

- Document level sentiment analysis obtains the sentiment of a complete document or paragraph.
- Sentence level sentiment analysis obtains the sentiment of a single sentence.
- Sub-sentence level sentiment analysis obtains the sentiment of sub-expressions within a sentence.

3.2 Types of Sentiment Analysis

There are many types and flavors of sentiment analysis and SA tools range from systems that focus on polarity (positive, negative, neutral) to systems that detect feelings and emotions (angry, happy, sad, etc) or identify intentions (e.g. interested v. not interested). In the following section, we'll cover the most important ones.

3.2.1 Fine-grained Sentiment Analysis

Sometimes you may be also interested in being more precise about the level of polarity of the opinion, so instead of just talking about positive, neutral, or negative opinions you could consider the following categories: Very positive, Positive, Neutral, Negative, Very negative.

This is usually referred to as fine-grained sentiment analysis. This could be, for example, mapped onto a 5-star rating in a review, e.g.: Very Positive = 5 stars and Very Negative = 1 star.

Some systems also provide different flavors of polarity by identifying if the positive or negative sentiment is associated with a particular feeling, such as, anger, sadness, or worries (i.e. negative feelings) or happiness, love, or enthusiasm (i.e. positive feelings).

3.2.2 Emotion detection

Emotion detection aims at detecting emotions like, happiness, frustration, anger, sadness, and the like. Many emotion detection systems resort to lexicons (i.e. lists of words and the emotions they convey) or complex machine learning algorithms.

One of the downsides of resorting to lexicons is that the way people express their emotions varies a lot and so do the lexical items they use. Some words that would typically express anger like shit or kill (e.g. in your product is a piece of shit or your customer support is killing me) might also express happiness (e.g. in texts like This is the shit or You are killing it).

3.2.3 Aspect-based Sentiment Analysis

Usually, when analyzing the sentiment in subjects, for example products, you might be interested in not only whether people are talking with a positive, neutral, or negative polarity about the product, but also which particular aspects or features of the product people talk about. That's what aspect-based sentiment analysis is about. In our previous example:

"The battery life of this camera is too short."

The sentence is expressing a negative opinion about the camera, but more precisely, about the battery life, which is a particular feature of the camera.

3.2.4 Multilingual sentiment analysis

Multilingual sentiment analysis can be a difficult task. Usually, a lot of preprocessing is needed and that preprocessing makes use of a number of resources. Most of these resources are available online (e.g. sentiment lexicons), but many others have to be created (e.g. translated corpora or noise detection algorithms). The use of the resources available requires a lot of coding experience and can take long to implement.

An alternative to that would be detecting language in texts automatically, then train a custom model for the language of your choice (if texts are not written in English), and finally, perform the analysis.

4. Proposed Semantic-based Similarity Measures

4.1 Types of Sentiment Analysis

In this paper, we use pre-trained sentence encoders as combination of Smooth Inverse Frequency and referenced model Google Sentence Encoder.

The method for estimating the semantic analogy between a pair of sentences is to average the words of the inlays of all the words in two sentences and calculate the cosine between the resulting inlays. Obviously, this simple baseline leaves a considerable space for diversity.

Taking the average of word inlays in a sentence tends to give too much weight to words that are quite irrelevant, semantically speaking. Smooth Inverse Frequency (SIF) tries to solve this problem in two ways:

- Weighting the key terms: using term-frequency and inverse-document frequency (tf-idf)
- Common component removal: SIF computes the principal component of the resulting embedding for a set of sentences. It then subtracts from these sentences embedding their projections on their first principal component. This should remove variation related to frequency and syntax that is less relevant semantically.

SIF removes unimportant words, alternatively known as stopping words such as but, just, etc., and keeps the information that can expose the most to the semantics of the sentence.

Sentiment Analysis task is considered a sentiment classification problem. The first step in the SC problem is to extract and select text features. Some of the current features are [62]:

Terms presence and frequency: These features are individual words or word n-grams and their frequency counts. It either gives the words binary weighting (zero if the word appears, or one if otherwise) or uses term frequency weights to indicate the relative importance of features [63].

Parts of speech **(***POS***)**: finding adjectives, as they are important indicators of opinions.

Opinion words and phrases: these are words commonly used to express opinions including *good or bad*, *like or hate*. On the other hand, some phrases express opinions without using opinion words. For example: *cost me an arm and a leg*.

Negations: the appearance of negative words may change the opinion orientation like *not good* is equivalent to *bad*.

4.2 Proposed Architecture of Similarity Matching

By referencing the similarity matching models of InferSent [19] and the Google Sentence Encoder, we built a pre-trained encoder for phrase-level and word-level semantic matching of social network comments and opinion-contained short texts.

In this encoder, we use soft-max combination phases with three layers to organized the partial results obtained from phrase and word level similarity matching results.

To demonstrate how proposed system works on similarity matching upon the short text that can probably found on social networks.

In matchmaking process, word-level measures can be categorized into the following three classes:

- Exact match,
- Word Transformed Match, and
- Longest common substring (LCS)
- Ignorance of word sequence (IWS)

In this paper, we assume all types of similarity as similar so, we take the matching result as one, whereas combined phrase level or sentence level, with the help of WordNet, we find the sparsity of a word and find their semantic meanings, in this case, the similarity values is regarded depending on the distance they are found.

5. Datasets and Implementation

5.1 Development Setting

To test the efficiency of proposed models and matching process, we use two alternative datasets so that the proposed system is shown to be independent of datasets. For every experimental works, we set different variations of words and sentence structure so that the overall average result is summarized and shown in the figures.

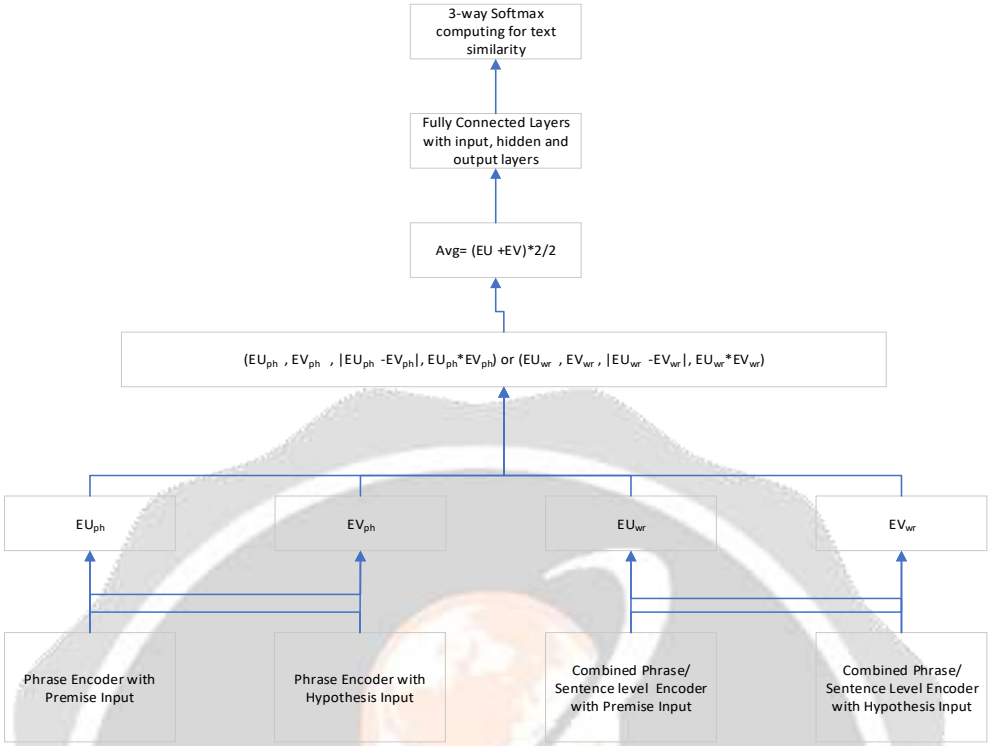

Fig -1 Pre-trained Encoder Model for Phrase and Setence Level Semantic Similarity Matching

5.2 Experimental Works

The experiments are performed in two setting aspects as shown in following subsections to measure the accuracy and error rate executed by the proposed system. The accuracy rate is used to determine if a value is accurate compare it to the accepted value. In an experiment observing a parameter with an accepted value of VA and an observed value VO, there are two basic formulas for percent accuracy:

Accuracy (%) = (VA - VO)/VA X 100

Percent error is the difference between a measured and known value, divided by the known value, multiplied by 100%.

5.2.1 Accuracy Measurement: This experiment is to measure how the proposed system accurately or mistakenly matches the text which is disguised in different synonyms and semantic relations. According to the results illustrated in Chart-1 our proposed semantic match achieves significant better results against traditional syntactic matching.

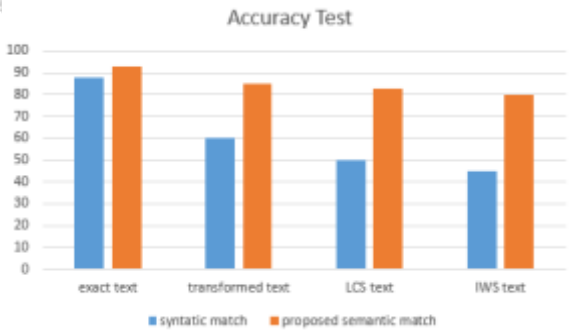

Chart -1 accuray measurement on text variations

5.2.2 Error rate Measurement: we propose the following definition of text entry error rate, given a presented text string (A) and a transcribed text string (B). The reason how we can get a more accurate sentiment score through this proposed system is that with the best social analyzing tools, we are getting data from online conversations all

over the web. The biggest trap social media monitoring tools fall in is that they only incorporate user data from social sites like Twitter. The error rate is measured accrodgin to the following equations depending on the input text (A) and transcribed text string (B). The results are shown in Chart 2 as follows.

$$ErrorRate = \frac{MSD(A,B)}{max(|A|,|B|)} \times 100\%$$

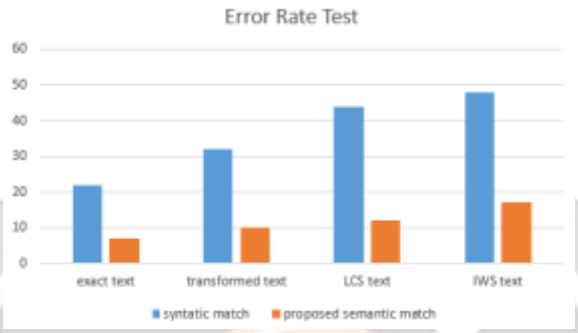

Chart -2 error rate measurement on text variations

6. CONCLUSIONS

In this paper, we search users of the group to share the same interests by studying their textual comments. The main purpose is to retrieve the user group by analyzing their context's semantically similarity so that the publishers' interest centers and, group interests can be revealed. This paper presents a text similarity in semantic ways using pre-defined encoders in both word and phrase level matching and gains far better results compared with other approaches. As future work, we have plan to extend this approach with better promising techniques so that better results could be revealed.

7. REFERENCES

[1] Cohen, W., Ravikumar, P., & Fienberg, S. (2003a). A comparison of string metrics for matching names and records. In Kdd workshop on data cleaning and object consolidation, 3, (pp. 73-78).

[2] A semantic similarity-based perspective of affect lexicons for sentiment analysis, Knowledge-Based Systems Volume 165, 1 February 2019, Pages 346-359

[3] Jimenez, S., Gonzalez, F., & Gelbukh, A. (2010). Text comparison using soft cardinality. In International Symposium on String Processing and Information Retrieval (pp. 297-302). Springer Berlin Heidelberg.

[4] Ryan MichaelsKeshavan, RaviDavid, Adamo,Jr , A Text Similarity Approach to Sentiment Classification (of Movie Reviews) using SentiWordNe, 4(1), 1–17. https://doi.org/10.1145/2414425.2414434

[5] Leung, C. W., Chan, S. C., & Chung, F(2006), Integrating Collaborative Filtering and Sentiment ANalysis: A Rating Inference Approach. ECAI 2006 Workshop on Recommender Systesm, 62-66, https://doi.org/10.1.1.69.1870

[6] Gabrilovich Evgeniy, & Markovitch Shaul. (2007). Computing Semantic Relatedness Using Wikipedia-based Explicit Semantic Analysis. Paper presented at the IJCAI.

[7] Cohen, W., Ravikumar, P., & Fienberg, S. (2003b). A comparison of string distance metrics for name-matching tasks. In II Web (pp. 73-78).

[8] Masashi Hadano, Kazutaka Shimada, Tsutomu Endo, Aspect Identification of Sentiment Sentences Using A Clustering Algorithm, Procedia - Social and Behavioral Sciences Volume 27, 2011, Pages 22-31

[9] Fattah Mohamed Abdel, & Ren Fuji. (2008). Automatic text summarization. World Academy of Science, Engineering and Technology, 37, 2008.

[10] Liu Yang, Sun Chengjie, Lin Lei, & Wang Xiaolong yiGou: A Semantic Text Similarity Computing System Based on SVM. Paper presented at the Proceedings of the 9th International Workshop on Semantic Evaluation, pages 80-84, Denver, Colorado, USA.

[11] Manning Christopher D, Raghavan Prabhakar, & Schütze Hinrich. (2008). Introduction to information retrieval (Vol. 1): Cambridge university press Cambridge.

[12] N Gali, R Mariescu-Istodor, D Hostettler , Framework for syntactic string similarity measures, Expert Systems with Applications, 2019.

[13] Sultan Md Arafat, Bethard Steven, & Sumner Tamara. (2014b). DLS@CU: Sentence Similarity from Word Alignment. Paper presented at the Proceedings of the 8th International Workshop on Semantic Evaluation, pages 241-246, Dublin, Ireland

[14] Zhao, X., Niu, Z., & Chen, W. (2013b). Opinion-based collaborative filtering to solve popularity bias in recommender systems. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) (Vol. 8056 LNCS, pp. 426–433). https://doi.org/10.1007/978-3-642-40173-2_35

[15] Mikolov Tomas, Chen Kai, Corrado Greg, & Dean Jeffrey. (2013). Efficient estimation of word representations in vector space. Paper presented at the ICLR, 2013.

[16] Leacock, C., & Chodorow, M. (1998). Combining Local Context and WordNet Similarity for Word Sense Identification. WordNet: An Electronic Lexical Database., (JANUARY 1998), 265–283. https://doi.org/citeulike-article-id:1259480

[18] Meng Lingling, Huang Runqing, & Gu Junzhong. (2013). A review of semantic similarity measures in wordnet. International Journal of Hybrid Information Technology, 6(1), 1-12.