Life Event Detection using Conversations from Social Media

Luis G. Moyano[1], **Paulo R. Cavalin**[1], **Pedro P. Miranda**[1]

[1]IBM Research
Av. Pasteur 138/146, Rio de Janeiro (RJ), Brazil

{lmoyano,pcavalin,pedropm}@br.ibm.br

Abstract. *In this work, we explore the possibility to detecting life events from Social Media by means of machine learning classification algorithms. One important difficulty of this kind of detection task is that, typically, Social Media posts are quite short, and there is not much context provided. This lack of context usually implies strong ambiguity leading to poor classification performance. Here, we propose the use of conversations as a means to augment context and improve classification performance. We evaluate single-post vs. conversation classification performance and compare different models for the conversations classifier. Finally, we describe the performance of the different classifiers in a large data set with 20,000 posts.*

Resumo. *Neste trabalho, investigamos a detecção de eventos de vida de mídias sociais através de algorítmos de classificação de aprendizado de máquina. Um grande desafio está relacionado ao fato que as postagens de mídias sociais são normalmente curtas, o que dificulta a captura do um contexto associado a elas. Esta falta de contexto normalmente implica em grande ambiguidade, e consequentement, em desempenho ruim na classificação. Neste trabalho propomos o uso de conversas como uma maneira de aumentar o contexto disponível e assim melhorar o desempenho da classificação de eventos de vida. Comparamos o desempenho na classificação de postagens contra conversas, e discutimos diferentes maneiras de construir classificadores para conversas. Finalmente, descrevemos o desempenho de diferentes tipos de classificadores em um conjunto de dados de teste de 20,000 postagens.*

1. Introduction

Social Media Networks (SMNs) have been attracting millions of people worldwide. Twitter, a microblogging platform and one of the most popular SMNs, currently reports about 284 million active users, posting 500 million 'tweets' on average every day[1]. Facebook, another very widely used SMN, reports more than 1.32 billion active users[2]. The scale at which these SMNs operate makes them very useful for user analysis.

On SMNs, users are able to submit posts that may contain text, images and videos. The content of these posts can be either general, about a given subject (for instance weather, politics, TV shows, etc.) [Kwak et al. 2010], or it can also be personal, related to something that has happened, or is happening, in the user's or a friends'

[1]https://about.twitter.com/company

[2]http://www.dailymail.co.uk/sciencetech/article-2703440/Theres-no-escape-Facebook-set-record-stock-high-results-beats-expectations-1-32-BILLION-users-30-mobile.html

life, to which we refer as life event [Atefeh and Khreich 2013, Ehrlich and Shami 2010, Eugenio et al. 2013]. Birthdays, graduations, buying a house, getting a new job, are all examples of relevant life events [De Choudhury et al. 2013].

Although both posts containing more general or more personal topics can be helpful to understand people in SMNs, the latter offers the possibility of understanding events that are associated specifically with a given user. By properly detecting life events, people's profiles can be enriched with information that might not be available in other data sources [Li et al. 2014, Hernandez et al. 2013]. As a consequence, enriched profiles can lead companies to approach clients in a more focused and effective way. For example, a bank could offer to a client a loan once it is detected that he/she is getting married or getting a new job, or a real state agency could offer a larger house to a person that is having a baby.

The most straight-forward approach to detect life events is by analyzing individual posts [Khobarekar 2013, Choudhury and Alani 2014]. In this case, the goal is to process a given text using Natural Language Processing (NLP) techniques, use some approach (Machine Learning, Rules, etc) to detect if it is about a life event or not, and possibly extract additional relevant information (who?, when?, where?, etc). However, this task can be very challenging. Messages on social media can typically be noisy and ambiguous, making it difficult to understand the real meaning of the post. Moreover, a single post might not contain all the necessary information about the life event. This happens especially on microblogging platforms such as Twitter, where posts consist of very short messages.

The use of conversations can improve the way life events are evaluated. A conversation consists of a set of messages connected with each other, posted by more than one user, and referring to the same subject. An example of conversation is when someone that is celebrating his/her birthday posts about this on a SMN. Typically, friends in the network interact with that user by sending congratulatory messages. Another example is when somebody comments that is going on a trip and a friend asks where and/or when the trip is, or posts 'have a nice trip' or something similar (see Figure 1 for a detailed illustration of a conversation). Briefly, conversations not only can help to identify events with higher precision (since more text is available for the tasks), but also can be a way to infer additional information that might not be in the original post.

This work has two main goals. First, we aim at developing a classifier for classifying conversations in a life event detection task, using conversation-specific feature sets, and compare its accuracy with that of a classifier for single posts. In this case, the idea is compare whether the conversation classifier can be better in practice or not. Second, we apply both classifiers on a large set of posts and compare their results in terms of finding life events posts. In this case, the goal is to observe if the number of life events detected will increase or decrease when a large dataset is used. Both evaluations consider Twitter posts in Portuguese, and we focus on detecting users posting about travelling.

2. Related Work

The possibility of mining content from Social Media Networks with the aim of detecting relevant events in people's lives has drawn a lot of attention in the last years [Atefeh and Khreich 2013]. Additionally, the opportunity of extracting related informa-

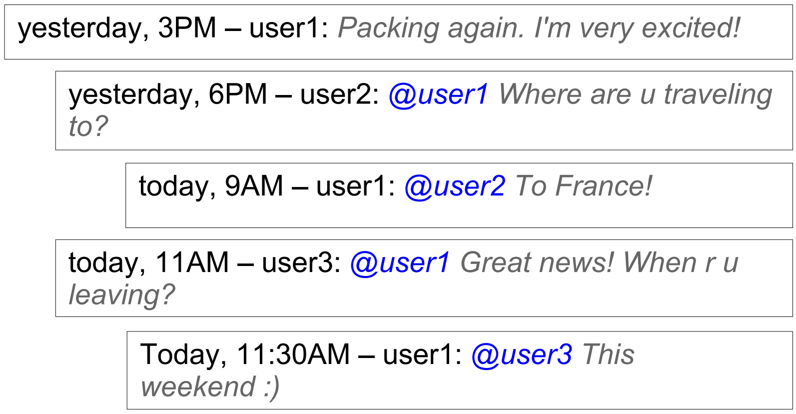

Figure 1. An illustration of a conversation. First, user1 posts about how excited he/she is about an up coming trip. Then, user2 gets curious and asks where user1 will travel to, and user1 answers that he/she is going to France. Next, user3 asks when user1 will travel, and user1 replies saying that it will be next weekend.

tion in order to better understand these events providing further context makes Social Media Networks an ever more interesting platform for analysis [Hernandez et al. 2013, Cavalin et al. 2014]. Different approaches have been proposed to detect life events, the majority of which consist in some combination of Natural Language Processing tools for extracting information from posts and Machine Learning techniques to classify posts in useful categories corresponding to the events of interest.

We mentioned earlier that there are a number of challenges in automatically processing Social Media for event detection, and specially microblogging platforms such as Twitter. For instance, the various degrees of informality in its use, the different types of contents, or the multiple styles of writing, all obscure user content and contribute to decrease classification performance. In the case of microblogging platforms, this problem is aggravated by the inherent short nature of posts, where lack of context invariably results in ambiguity. In works that use Machine Learning to classify life events, the most widely used strategy is to engineer a strong combination of features to improve performance. In a recent result [Eugenio et al. 2013], several classifiers are compared with different combinations of features (unigrams, Parts of Speech (PoS), history, among other methods). The authors find that most accurate classification approach (at least in the case of these class of life events) is also the simplest, i.e., by means of unigrams. More involved classification and information extraction procedures have been proposed, as is the case of [Li et al. 2014], where certain kinds of posts are considered as seeds leading to a dynamic crawling strategy which, combined with a number of classification methods, constitute a full-fledged life event detection system. An up-to-date survey, centered in the Twitter platform, can be found in [Atefeh and Khreich 2013] summarizing various event detection techniques (including supervised vs. unsupervised methods and retrospective vs. new detection tasks).

Other approaches exist in the literature to tackle the problem of life event detection, which focus on providing increased context (from other sources beyond text) to improve performance. One example of this is the use of information about social features characterizing the different users. This is the case of [Choudhury and Alani 2014], which makes use of some social features such as number of tweets, retweets, mentions, favorites, etc., combined with linguistic features. In a related approach, the authors in [De Choudhury et al. 2013] use a combination of emotion measures as well as linguistic measures to improve their classification methods. Another work includes context through history, but focusing on a different classification problem (sentiment analysis), is used in [Vanzo et al. 2014], where a stream of tweets is modeled through a Markovian Support Vector Machine (SVM). Temporal information in the form of conversations has been studied before (see [Honey and Herring 2009]), but in the context of understanding the nature of the exchange of information in the Twitter platform, and its potential as a collaboration tool.

Our work combines several of the different aspects discussed above. We focus on measuring the relative influence of considering entire conversations (i.e., we consider all related posts grouped by replies) as input for life event classification. Unlike some previous work, that have considered the use of conversations as an additional feature among several others [Li et al. 2014, Eugenio et al. 2013], here we compare in detail the effect of including history as opposed to considering only single posts. In addition, we focus on extending feature sets to take into account the particular characteristics of conversations instead of relying on more complex classification approaches such as that proposed in [Vanzo et al. 2014].

3. Methodology

In this section we provide a definition of conversations and describe the proposed approaches for classification and feature extraction.

3.1. Single Posts versus Conversations

We consider single posts as documents from a corpus which are treated independently. In other words, suppose $\mathbb{D}$ is a set of documents (the corpus), each document $d_i \in \mathbb{D}$ is treated as not having any relation to any other post in $\mathbb{D}$.

Conversations, on the other hand, consist of subsets of documents which are linked with each other by a common subject or some meta-data, for instance the 'in-reply-to' field in Twitter posts. In this case, there is a set $\mathbb{C} = \{c_j, \ldots, c_M\}$, where $\mathbb{C} \subset \mathbb{D}$. Each c_j contains two or more documents d_i.

3.2. Single Post Classifier

Since posts are individual documents, the classifier considered in this paper consists of a traditional machine learning-based approach using bag of n-grams as input features, which we describe in the following paragraphs.

After conducting basic pre-processings steps, for instance tokenization, case normalization and removal of stop words, features are extracted by computing the presence/absence of words and n-grams in a previously-computed vocabulary $\mathbb{V}$. Thus, during this process, each document d_i is associated to a binary vector v_i, where positions marked with 0 represent the absence of word $w_j \in \mathbb{V}$, while those marked with 1 represent its presence. Note that $|v_i| = |\mathbb{V}|$.

The entire process of creating and using this classifier involves two phases: 1) training, during which the machine learning classifier Λ is trained on a set of labeled documents, denoted $\mathbb{T}$; and 2) test (or operation), where Λ is used to predict to which class a given previously-unobserved document belongs to. It is worth mentioning that $\mathbb{V}$ is usually built with all words/n-grams in $\mathbb{T}$, except those with very low frequency. In addition, the labels during both phases must be associated with the pre-defined set of classes in Ω.

3.3. Conversation Classifier

In this work, we focus on investigating different feature sets for classifying conversations. In other words, the method is similar to that presented in Section 3.2, where a single feature vector v_j is extracted for each conversation c_j, and this vector is inputted to the classifier at both training and test phases.

Even though conversations can be presented in varied lengths, here we consider that all v_j have the same length, so a traditional machine learning classifier such as Logistic Regression or Support Vector Machine can be used. The feature extraction process, then, must take into account the structure of the conversations to take advantage of the interaction and the order of the messages.

In this work, we consider the following feature sets.

3.3.1. Extended N-Grams

This feature sets is basically an N-gram feature set, but considering all messages in a conversation as part of a single document.

In greater detail, consider that a conversation c_i consists of a set of documents, i.e. $c_j = \{d_1, \ldots, d_K\}$. For each document d_k in c_j, a feature vector v_k can be extracted as in Section 3.2. Then, the feature vector v'_j for c_j can be defined as the combination of all vectors $v'_j = \{v_1, \ldots, v_K\}$. Since v_k is a binary vector, $v'_j = \{v_1 \vee \ldots \vee v_K\}$.

3.3.2. Co-occurrence of N-Grams

In order to improve the structure of conversations and the interaction between the users and the messages, we propose a feature set that takes into account the co-occurrence of terms or phrases between two consecutive messages in the thread. Then, we compute the co-occurrence of n-grams between subsequent conversation documents. These features are encoded with a presence/absence binary feature vector, similar to bag of n-grams.

In other words, consider that the documents in c_j are organized in their chronological order or creation (posting). Then, for each d_k and d_{k+1} in c_j, we compute the presence/absence of each pair of n-gram (ng_1, ng_2), where ng_1 appears in d_k and ng_2 in d_{k+1}, and save this presence/absence vector considering all pairs of n-grams in, V_{ng}, the vocabulary of n-grams, in v'_j.

4. Experimental Evaluation

In experiments herein presented, the main goal is to compare the performance of classifying life events using conversations, compared with the classification of single posts. By doing so, we focus on posts and conversations related to travel, which is a life event that is very frequently commented by users on social networks.

The evaluation dataset have been collected from Twitter with the Twitter Search API. In this case, we search for posts in Portuguese that contained a few words or phrases of interest, such as *viagem* (trip), *viajar* (to travel), *vou viajar* (I'm going to travel). With this dataset, we have constructed two labeled datasets, both with 500 samples: one with single posts and another with conversations. The labeling of the first dataset has been straightforward, consisting only of marking if a posts represents a comment of the user about a personal trip (life event) or not (non life event). In the case of conversations, the labeling process was similar, but it required the crawling of all other messages involved in the conversations. We did that with the Twitter REST API. Through this API, we are only allowed to capture parent tweets but no subsequent replies.

The base classifier used in this work consists of Linear Support Vector Machines (SVM), provided by LibLinear[3]. The results provided in the remainder of this section represent the average of 5-fold cross validation, where the confusion matrices are composed of the sum of the confusion matrix of each of the five testing folds. The cost parameters of the classifier has been optimized for each fold.

The measures used for comparing the classifiers are Precision, Recall, Accuracy and F-Score, defined in equations 1, 2, 3, and 4, respectively. Here, TP stands for the total of positive (life events) samples classified correctly; TN represent the total of negative (non life events) samples classified correctly; FP corresponds to the total of negative samples classified as positive; and FN stands for the total of positive samples classified as negative.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3}$$

$$F - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

In Table 1 we present the results for single posts, considering bags of words and 2-grams. The total accuracy was of about 67%, while the precision and recall were of about 37% and 49%. The F-Score was of 0.42.

In Table 2 and Table 3 are presented the results of the conversation classifier, using as features Bag of Words, and Bag of Words and 2-grams, respectively. The accuracy, precision, recall and F-Score for the first feature case were approximately of 69%, 45%, 50%, and 0.47. Despite the small increase in accuracy compared with single posts, the

[3]http://www.csie.ntu.edu.tw/ cjlin/liblinear/

Table 1. Confusion matrix for single post classifier. Precision: 0.37, Recall: 0.49, Accuracy: 0.67, and F-Score: 0.42.

	P	**N**
P	62	102
N	63	273

F-Score is 0.05 higher given to a slight increase in recall and a more significant increase in precision. In the case of Bag of Words and 2-grams, it can be observed a slight increase in accuracy also, to nearly 70%, and an even higher increase in recall, to 54%. Nonetheless, in this case the precision is below of that of the single posts classifiers, with only 31%. For this reason, the F-Score of 0.39 is worse than that of single posts.

Table 2. Confusion matrix for conversation classifier with bag of words. Precision: 0.45, Recall: 0.5, Accuracy: 0.69, and F-Score: 0.47.

	P	**N**
P	69	84
N	69	277

Table 3. Confusion matrix for conversation classifier with bag of words and 2-grams. Precision: 0.31, Recall: 0.54, Accuracy: 0.70, and F-Score: 0.39.

	P	**N**
P	48	105
N	40	306

Finally, in Table 4 we present the results of the conversation classifier using co-occurrence of n-grams as features, where only words (1-gram) were considered. This classifier reaches the best results, with an accuracy of 73%, precision of 49%, recall of 57%, and F-Score of 0.53. This shows that a feature set specifically designed for conversations make conversations better to detect life events.

A summary of the results is presented in Table 5. The best results are achieved by the conversation classifier using co-occurrence of n-grams (C_{co}). Note that SP represents the classifier for single post, C_1 the conversation classifier with bag of words, and C_2 the conversation classifier with bags of words and 2-grams.

5. Conversations versus Single Posts to Find Life Events

To validate the proposed method to detect life events using conversations, we apply the system on a large scale dataset and compare the numbers generated by the single posts classifier. The main idea is, given a set of posts, to calculate how many life events are found by the system, and to compare how this number changes with the conversations classifier.

For doing so, first we used Twitter Streaming API to collect a set of 20,000 posts which were possibly related to travel. This was achieved by collecting posts that contained at least one of the following: *viajar*, *viagem*, *viajarei*, *viajando*, and *viajem*. Note that we care to consider misspellings to broaden the range of posts that could be collected.

Table 4. Confusion matrix for conversation classifier with co-occurrence of n-grams. Precision: 0.49, Recall: 0.57, Accuracy: 0.73, and F-Score: 0.53

	P	**N**
P	75	78
N	56	290

Table 5. Summary of results (in %).

	SP	C_1	C_2	C_{co}
Precision	37.0	45.0	31.0	**49.0**
Recall	49.0	50.0	54.0	**57.0**
Accuracy	67.0	69.0	70.0	**73.0**
F-Score	0.42	0.47	0.39	**0.53**

The first step of this study consisted in applying the single-post classifier. This resulted in obtaining 239 posts pointed out as life events, which represents 1.2% from the total of posts.

In the second step, we first completed the dataset with the corresponding conversation posts (similar to the dataset described in Section 4). From the 20,000 posts, 1,332 conversations have been crawled. By applying the classifier on the conversations, 71 life event conversations have been found, i.e. 5.3% of the conversations and 0.4% from the total of posts.

Finally, we combined both classifiers and applied them on the entire set of single posts and conversations, in which the classifier is chosen in accordance with the type of document, i.e. single post or conversation. This set consists of 17,836 documents, worth mentioning that the size of the original set decreased since many posts were fused into common conversations. In this set, a total of 310 life events have been detected, i.e. 1.73% of the total of documents.

A summary of these results in presented in Table 6.

Table 6. Summary of results from applying life events detection on a large set.

	# Documents	**# Life Events Detected**
Single Posts	20,000	239 (1.2%)
Conversations	1,332	71 (5.3%)
Mixed	17,836	310 (1.73%)

6. Conclusions

Life event detection in social media networks is a challenging classification problem, due to the many inherent characteristics of posts in SMNs that difficult classification tasks and weaken their performance. The systematic study of the different ways these obstacles can be overcome is central for many researchers in the field. In this work, we focus on including temporal information for expanding context and detection task improvement.

We compare the classification of life events in Twitter using single posts vs. conversations (both sets with 500 samples in Portuguese), using Linear Support Vector Machines. Additionally, we compare different sets of features (bag of words (BoW), BoW +

2-grams and co-occurrence of n-grams) in order to have a more robust understanding of the influence of more context via conversations.

Our results show that by using conversations instead of single posts, there is a significant improvement in precision, recall, accuracy and F-Score, for every combination tried (except for precision, when using BoW + 2-grams). The best combination is achieved by using co-occurrence of n-grams, leading to 49% precision, 57% recall, 73% accuracy and 0.53 F-Score. Additionally, we validate our classifier in a data set containing 20,000 posts, where we combine both approaches depending if it is an isolated post or part of a conversation, capturing 1.7% of posts are related to life event under study.

This work points to the fact that it is possible to use additional context contained in social media to improve performance in life event detection tasks. Future work in this line includes considering Part of Speech (PoS) and Named Entity Recognition (NER) information to expand context as well as the use of images as drivers of context information.

References

Atefeh, F. and Khreich, W. (2013). A survey of techniques for event detection in twitter. *Computational Intelligence*.

Cavalin, P., Gatti, M., and Pinhanez, C. (2014). Towards personalized offers by means of live event detection on social media and entity matching. In *1st International Workshop on Social Personalisation (SP 2014)*.

Choudhury, S. and Alani, H. (2014). Personal life event detection from social media. In *25th ACM Hypertext and Social Media Conference*. CEUR.

De Choudhury, M., Counts, S., and Horvitz, E. (2013). Major life changes and behavioral markers in social media: Case of childbirth. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, CSCW '13, pages 1431–1442, New York, NY, USA. ACM.

Ehrlich, K. and Shami, N. S. (2010). Microblogging inside and outside the workplace. In *ICWSM*.

Eugenio, B. D., Green, N., and Subba, R. (2013). Detecting life events in feeds from twitter. *2012 IEEE Sixth International Conference on Semantic Computing*, 0:274–277.

Hernandez, M., Hildrum, K., Jain, P., Wagle, R., Alexe, B., Krishnamurthy, R., Stanoi, I. R., and Venkatramani, C. (2013). Constructing consumer profiles from social media data. In *Big Data, 2013 IEEE International Conference on*, pages 710–716. IEEE.

Honey, C. and Herring, S. C. (2009). Beyond microblogging: Conversation and collaboration via twitter. In *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on*, pages 1–10. IEEE.

Khobarekar, J. (2013). *Detecting Life Events using Tweets*. PhD thesis, University of Illinois at Chicago.

Kwak, H., Lee, C., Park, H., and Moon, S. (2010). What is twitter, a social network or a news media? In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 591–600, New York, NY, USA. ACM.

Li, J., Ritter, A., Cardie, C., and Hovy, E. (2014). Major life event extraction from twitter based on congratulations/condolences speech acts. In *Proceedings of Empirical Methods in Natural Language Processing*.

Vanzo, A., Croce, D., and Basili, R. (2014). A context-based model for sentiment analysis in twitter. In *Proceedings of COLING*, pages 2345–2354.