International Journal of Computational Linguistics and Applications, Vol. 6, No. 2, 2015, pp. 141–158
Received 25/01/2015, Accepted 27/02/2015, Final 20/12/2015.
ISSN 0976-0962, http://ijcla.bahripublications.com

A #hashtagtokenizer for Social Media Messages

VLÁDIA PINHEIRO
RAFAEL PONTES
VASCO FURTADO
Universidade de Fortaleza, Ceará, Brasil

ABSTRACT

In social media, mainly due to length constraints, users write succinct messages and use hashtags *to refer to entities, events, sentiments or ideas.* Hashtags *carry a lot of content that can help in many tasks and applications involving text processing such as sentiment analysis, named entity recognition and information extraction. However, identifying the individual words of a* hashtag *is not trivial because the traditional POS taggers typically consider it as a single token, despite the fact that it might contain multiple words, e.g.* #fergusondecision*,* #imcharliehebdo*. In this work, we propose a generic model for* hashtagtokenisation *that aims to split up one* hashtag *into several tokens corresponding to each individual word contained in it (e.g. "*#imcharliehebdo*" would become four tokens, "#", "*i*", "*am*" and "*Charlie Hebdo*"). Our* hashtagtokenizer *is based on a machine learning segmentation method for Chinese language and makes also use of Wikipedia as encyclopedic knowledge base. We have evaluated the inference power of our approach by comparing the tokens produced by our approach to those produced by human taggers. The results demonstrated the good accuracy and applicability of the proposed model for general-purpose applications.*

Keywords: Social media, information extraction, tokenization, hashtag

1. INTRODUCTION

Social medias are extremely popular and thus generate so much user-generated data. These data are interesting from a computational linguistic point of view to investigate the potential of extracting useful information from this data. In the field of natural language processing (NLP), a large number of tools rely on the availability of morphosyntactic or part-of-speech (POS) information about texts of microblogs and of social networks reviews [1]. In these social media, users post texts ina very specifically way that requires special handling. Mainly due to the restriction on the length of the messages, users write succinct messages and use *hashtags* to refer to another entities or events in order to facilitate the text retrieval and to express sentiments and ideas. The use of *hashtags*[1] is a popular way to give the context of a tweet or the core idea expressed in the tweet [2].For example, the *hashtag* #savethenhs reads as 'savethenational health service.'[2]

Hashtags carry a lot of content that can help in many tasks and applications involving NLP such as sentiment analysis [3,4], named entity recognition, information extraction and retrieval [5], prediction of the spread of ideas for marketing purposes [2,6]. Maynard and Greenwood [3] claim that much useful sentiment information is contained within *hashtags* and that we can make use of the information contained within them for sentiment detection. For example, we can recognize positive and negative words within a *hashtag*.

However, identifying the individual words of a *hashtag* is not trivial because the traditional POS taggers typically tokenize the *hashtags* as a single token, although they contain multiple words, e.g. #notreally, #fergusondecision, #imcharliehebdo. General purpose tokenizers need to be adapted to work correctly on social media, in order to handle specific tokens like URLs, *hashtags*,

[1] A *hashtag* is a sequence of non-whitespace characters preceded by the hash character "#". For example, #healthcarereform is a hashtag.

[2] Note that National Health Service (NHS) stands for the four publicly funded health care systems in the countries of the United Kingdom.

user mentions in microblogs, special abbreviations, and emoticons. Few works focus on decomposing the *hashtag* into tokens. Systems those somehow aim to discover the individual tokens of a *hashtag* or are incipient or intend to specific situations [2,3]. For example, Maynard and Greenwood [3] use gazetteers and a dictionary of common slang for detecting sarcasm within hashtags.

In this work, we propose a model for *hashtag*tokenisation that aims to split up a *hashtag* (commonly defined by traditional POS taggers) in several tokens corresponding to each individual word contained in the hashtag (e.g. “#imcharliehebdo” becomes four tokens, “#”, “i”, “am” and “Charlie Hebdo”). We argue that this model can be coupled in a traditional POS tagger to leverage the potential of *hashtag* analysis in NLP processors. Our #hashtagtokenizer is based on an unsupervised word segmentation approach, used for Chinese language [7], plus a linguistic and worldwideknowledge approach which uses a lexicon and anencyclopedic knowledge base. Specifically, we propose a segmentation algorithm which generates the possible segmentation options $s_j = w_1 \oplus \ldots \oplus w_n$ of a hashtag***h***, where w_i is a valid word hypothesis. In order to do so, we have searched in a lexicon or vocabulary of the language used and in Wikipedia. This strategy addresses the observation that several *hashtags* are composed of people’s names, places, brands, etc, and may be directly solved through world knowledge bases (e.g. #iphone, #androidgames). To solve the best segmentation options for a *hashtag**h***, our approach utilizes a word induction score inspired on the work of [7]. This score is calculated from a value of external boundaries, which captures the limits to the left and right of valid word hypothesis through a co-occurrence matrix. It also makes use of a value of internal boundary, which indicates the most likely separation point of a determined word combination.

We use the implemented #hashtagtokenizer in a list of the most frequent *hashtags* from a corpus collected from Twitter in December, 2014. We conducted experiments not only to analyze the accuracy of the proposed model but also the challenges to discover the components words of a *hashtag*. The results demonstrate the good accuracy and applicability of the proposed model for general-purpose applications.

2. Related Work

Gimpel et al. [8] address the problem of POS tagging for English data from Twitter. Starting from scratch, they developed an English Twitter-specific tag set. Using this tag set, they manually corrected English tweets that were annotated using Stanford POS tagger [9] and additionally developed features to build a machine learning classifier that tags unseen tweets. Avontuur et al. [1]propose a similar approach for Dutch tweets. TwitIE [5] is an open-source NLP pipeline customized to microblog text at every stage that comprises: language identification, tokenizer, normalizer, POS tagger and Named Entity Recognition. In this work, the authors recognize that general purpose tokenizers need to be adapted to work correctly on social media, in order to handle specific tokens like URLs, hashtags (e.g. #nlproc), user mentions in microblogs (e.g. @GateAcUk), special abbreviations (e.g. RT, ROFL), and emoticons. TwitIEtokenizer follows Ritter's tokenisation scheme [10] and treats *hashtags* and user mentions as two tokens (i.e., '#' and 'nlproc' in the above example) with a separate annotation HashTag covering both. All the aforementioned works focus on identifying one specific tag (e.g. /HASH) instead of a complete decomposition of a hashtag into several tokens.

In [2] the analysis of a hashtag content has been done for understanding meme propagation. The authors prepared a regression model using features aboutthe length of the hashtag(characters and words) and the lexical items. These lexical items are manually segmented. After that therole of each one of them is discovered through searches in datasets of names, celebrities, countries, holidays, etc. It is worth mentioning that the focus of this work is not automatically identifying the lexical items of a *hashtag*.

Maynard and Greenwood [3] have compiled a number of rules, which enable to improve the accuracy of sentiment analysis when sarcasm is known to be present. In particular, they considered the effect of sentiment and sarcasm contained in hashtags, and they have developed a hashtagtokeniser for GATE [11], so that sentiment and sarcasm found within hashtags can be detected more easily. First, they try to form a token match against GATE's gazetteers (vocabulary, locations, organizations etc.), and against an edited dictionary of common slang words from [5]. According to their experiments, the hashtagtokenization achieves 98% precision. Unfortunately this work cannot be used as a parameter of comparison because the data used is not available and the impact of the gazeteers and dictionaries could not be evaluated.

3. THE #HASHTAGTOKENIZER MODEL

In this work, we propose a model for hashtagtokenisation that aims to split up the single token of a *hashtag* in several tokens corresponding to each individual word contained in it (e.g. "#imcharliehebdo" becomes four tokens, "#", "i", "am" and "Charlie Hebdo"). We argue that this model can be coupled in a traditional POS tagger to leverage the potential of *hashtag* analysis in NLP processors. Our #hashtagtokenizer is based on two approaches: (1) a lexicon and a world knowledge approach to identify tokens that express valid words of a lexicon of a language, or names of people, organizations, brands, locations, etc., in a encyclopedic knowledge base such as Wikipedia; (2) an unsupervised word segmentation approach of proposed in [7], used for languages where there is no visual representation of word boundaries in a text (e.g. Chinese or Japanese languages).

3.1. *Model overview*

Figure 1 presents the generic pipeline of our proposed model for hashtag tokenization.

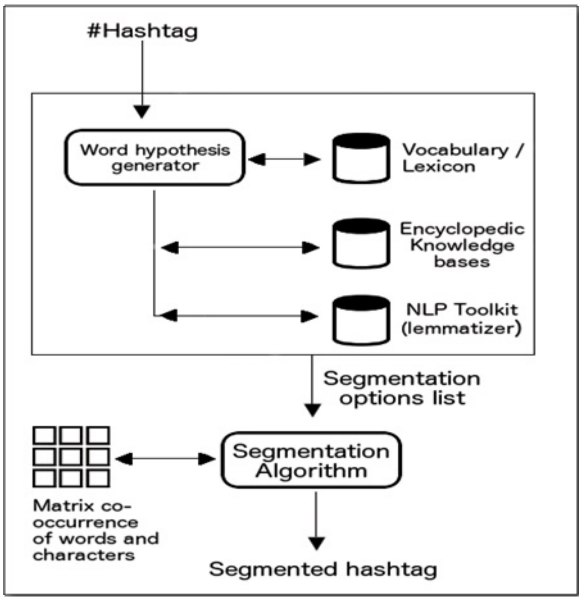
#Hashtag
Word hypothesis generator
Vocabulary / Lexicon
Encyclopedic Knowledge bases
NLP Toolkit (lemmatizer)
Segmentation options list
Segmentation Algorithm
Matrix co-occurrence of words and characters
Segmented hashtag

Figure 1. *The HashtagTokenizer Pipeline*

The **Word Hypothesis Generator** receives a *hashtag**h*** (typically a single token) and tries to decompose it in a sequence of valid word hypothesis. A word hypothesis is valid whether it exists in a vocabulary or knowledge base. The set of w_i valid words hypothesis are then sent to the ***Segmentation Algorithm*** that searches for the sequence that maximizes the *WordRank* score. Such a sequence is defined as the segmentation of ***h***. We are going to detail this process as following.

3.2. *Segmentation algorithm*

We developed a Viterbi-like algorithm in order to the search for the optimal segmentation of a hashtag***h*** (an utterance of continuous characters). Firstly, the segmentation algorithm is used to find the possible segmentationsj = $w_1 \oplus \ldots \oplus w_n$ of***h***, where w_i is a valid word hypothesis. Thereafter, it searches for the segmentation option that maximizes the following objective function (Formula 1):

$$s(h) = \text{argmax}_{w_i + \cdots + w_n} WR(\text{wi}) \quad (1)$$

where $WR(w)$ is the WordRank score of a valid word w. In other words, the resulting segmentation of hashtag***h*** is the segmentation option $w_1 \oplus \ldots \oplus w_n$ with the highest function value.

3.2.1. *Generating the segmentation options*

Given an unsegmented *hashtag**h***, we may retrieve “naive” word hypotheses by considering all the characters sequences to be word hypotheses. For example, for the hashtag “#fergusondecision”, we may retrieve the following naive segmentation options - “#”, “fer”, “gus”, “ond”, “eci”, “sion” - among many other possible combinations of character sequences. In order to reduce the number of segmentation options of ***h***, we use the following strategies:

1. Search in encyclopedic knowledge bases for the word hypothesis w, formed by the complete *hashtag* without the character hash “#”. In case of wbeing founded, it is generated as the resulting segmentation of ***h***. This strategy is a response to the observation that *hashtags* are formed by names of people, entities, places, brands, etc, which can be directly solved by world knowledge bases (e.g. #youtube, #iphone, #android). For instance, for the *hashtag* ‘#android’, the segmentation option s_1 = ‘android’ is generated and searched for in a database such as Wikipedia. Since this verbatim is found on Wikipedia, the segmentation of a***h*** is simply w_1 = ‘#’ $\oplus$ w_2 = ‘android’.
2. In an interactive algorithm, ***h***is divided in all possible sequences of word hypothesis w_i, and for every w_i, the process checks if:
   a. It is a valid word from a dictionary or a language lexicon;
   b. Its lemma is a valid word from a dictionary or a language lexicon

c. It is a verbatim from an encyclopedic knowledge base, e.g Wikipedia.

In case of any of the situations aforementioned being true to all word hypothesis w_i, they are considered <u>*valid word hypothesis*</u> and the corresponding combination $w_i \oplus \ldots \oplus w_n$ is generated as a possible option for segmentation of ***h***. For instance, for the hashtag***h*** = 'good', the possible sequences for word hypothesis are the following:

$s_1 =$ *'g' ⊕ 'o' ⊕ 'o' ⊕ 'd'*
$s_2 =$ *'g' ⊕ 'o' ⊕ 'od'*
$s_3 =$ *'g' ⊕ 'oo' ⊕ 'd'*
$s_4 =$ *'g' ⊕ 'ood'*
$s_5 =$ *'go' ⊕ 'o' ⊕ 'd'*
$s_6 =$ *'go' ⊕ 'od'*
$s_7 =$ *'goo' ⊕ 'd'*
$s_8 =$ *'good'*

In this example the segmentation option s_4 is rejected because all word hypothesis 'g' e '*ood*' have not matchany of the mentioned conditions (a), (b) or (c). Options 5 was also rejected because the word hypothesis '*od*' failed, although '*go*' is a valid word in English. Options 8 was accepted since its word hypothesis '*good*' is a valid English word.

3. Finally, a list of segmentation options in the form of $w_i \oplus \ldots \oplus w_n$ is generated. It is worth noting that the use of a reliable encyclopedic knowledge and with a currently updated database allows for a more robust and in-depthmodel. Furthermore, it enables for some word hypothesis to have their meaning clarified since they have been associated to a Wikipedia concept, thus facilitating the Word Sense Disambiguation process.

3.2.2. *Computing the WordRank (WR) score*

We propose a word induction criteria based on [7], which propose the WordRank. The intuition of their idea is that word boundaries between adjacent words indicate the correctness of each other, i.e., if a word hypothesis has a correct (or wrong) word boundary, we may infer that its neighbor would simultaneously have correct (or wrong) word boundary at its corresponding side. This idea is similar to the ideas of Firth [12] that "You shall know a word by its company" and the distributional hypothesis of Harris [13], that words will occur in similar contexts only if they have similar meanings.

All this motivated us to construct a matrix of co-occurrence of words which expresses how often a word co-occurs in a corpora to other words from a window of [-n,+n] words. Latent Semantic Models are also based on information about the context of use [14]. Among those, Hyperspace Analog to Language (HAL) [15] is a model that acquires representations of meaning by capitalizing on large-scale co-occurrence information inherent in the input language stream. The basis for the methodology to represent the meaning of HAL is to develop a matrix of word co-occurrence values for a given vocabulary, from a large text corpus, using a window size. The smallest useable window would be [-1,+1] words, corresponding to only the immediately adjacent words. By constructing this matrix it was able to express the frequency of occurrence of words adjacent to the left and right of a given word$w$. Table 1 presents an example of a matrix of co-occurrence of words for an input corpus "*City Employees Plan Walkout for Police Reform. City employees are mobilized*", with window [-1,+1], which values express the frequency of a given wordw_k to the left (and right) of a wordw_l. By using the matrix row as guidance, the word 'city' co-occurs 02 (twice) to the left of the word '*employee*'. By using the matrix column as guidance, the word '*employee*' co-occurs 02 (twice) to the right of the word 'city'.

The matrix of co-occurrence of words is similar to the link structures proposed in [7], considering that it also expresses the connections between words adjacent to the left and right of a given word w.

Table 2. *Matrix of co-occurrence of words for an input corpus "City Employees Plan Walkout for Police Reform. City employees are mobilized", with window [-1,+1]*

	city	employee	plan	walkout	for	police	reform	are	mobilize
city		2							
employee			1					1	
plan				1					
walkout					1				
for						1			
police							1		
reform									
are									1
mobilize									

Having constructed the matrix of co-occurrence of words, it was possible to calculate the Left-side Information (LI) and Right-side Information (RI) for a valid word hypothesis w, according to Formula (2) and (3), respectively.

$$LI(w) = \sum\nolimits^{\cdot}_{l \in s_j} RI(l) \quad (2)$$

$$RI(w) = \sum\nolimits^{\cdot}_{r \in s_j} LI(r) \quad (3)$$

where,

- ***RI'***(***l***)is the frequency of co-occurrence of all words ***l*** in s_j which are present to the left of w (or in which w co-occurs to the right of l). In the co-occurrence matrix, it is the value of the cell M_{ij}, where i is the line of word l and j is the column of the word$w$.
- ***LI'***(***r***)is the frequency of co-occurrence of all words ***r*** in s_j which are present to the right of w (or in which w co-occurs to the left of l). In the co-occurrence matrix, it is the value of the cell M_{ij}, where i is the line of word ***w*** and ***j*** is the column of the word ***r***.

Finally, an External Value (EV) for a valid word hypothesis w is calculated by Formula (4).

$$EV(w) = LI(w) * RI(w) \tag{4}$$

According to [7], it is not enough to represent the goodness of a word hypothesis using only information of external boundaries, which, in this paper, is based on the co-occurrence frequency with words adjacent to the left and right (RI and LI). The justification is that word combinations are prioritized, since they might have a high EV value as long as the internal limits of the word hypothesis are ignored. Considerthe following example where the word hypothesis w = '*thatdog*', which external limits to the right and left were well defined. However, it is formed by two words 'that' and 'dog'. Thus we need to find the internal boundary between 't' e 'd' considering that the set of letters 'td' occurs more rarely in an English word.

To solve this problem, the Internal Value (IV) for a valid word hypothesis w is calculated by Formula (5), based on Mutual Information (MI) that measures the combining degree of pairs of adjacent characters [16]. It has been reported that a high MI value indicates a good chance of two characters combining together, whereasa low MI value indicates a word internal boundary between the two characters.

$$IV(w) = \min_{i=1,L-1}(MI(c_i, c_{i+1})) \tag{5}$$

Where L is the length of a given word hypothesis w and c_i is the i_{th} character of w. The Internal Value (IV) assumes the lowest MI value among all character pairs of the word hypothesis was long as it is the most likely point of the internal boundary of w.

Finally, we calculate the WordRank (WR) score for a word hypothesis w, according to [7, p.869], as follows:

$$WR(w) = EV(w) * f(IV(w)) \tag{6}$$

where,

- *f(x)* is the auxiliary function for optimal performance. Chen, Xu and Chang [7] use the functions polynomial *($f(x)=x^{\alpha}$)* and exponential *($f(x)=\beta^{x}$)* with parameters α and β. For example, for English, some experiments indicate that $\alpha = 4.4$ and $\beta = 4.6$ are optimal values.

4. EXPERIMENTAL EVALUATIONS

We have chosen the social media Twitter as a source of *hashtags* to evaluate our model. Twitter is a popular microblogging platform and users post *hashtags* to give the context of a tweet, mainly due to the restricted size of messages of only 140 characters. A study of 1.1 million tweets established that 26% of English tweets have a URL, 16.6% – a *hashtag*, and 54.8% – a user name mention [17].

The goals of our evaluationwere to analyze the accuracy of the proposed model and identify which are the main challenges in accomplishing this task. Our hypotheses for investigation are: (1) that the use of encyclopedic knowledge improves the accuracy of the approach; (2) the use of a lexicon and language vocabulary reduces the number of segmentation optionsand optimizes the model performance.

4.1. *DataSet and gold standard*

Using the Twitter 4J API[3], we collected 93000 Twitter posts with *hashtags* sent during one week in December, 2014. Overall it is 122705 *hashtags*. Table 2 presents the hashtags frequency distribution. Note that402distinct hashtags were used more than 100 times in the corpus and 120264 were mentioned from 1 to 19 times. We have than selected as the core of our dataset the 402more frequent hashtags.

[3] http://twitter4j.org/en/index.html

Table 3. *Frequency distribution of* hashtags *in the Twitter corpus.*

Frequency Range	**Number of hashtags**
>=100	402
80-99	113
60-79	235
40-59	398
20-39	1293
1-19	120264

To be able to evaluate the output of the *hashtag*tokenizer, a Gold Collection of correctly segmented *hashtags* is required. This allows for a comparison of the output of the implemented #hashtagtokenizer against this Gold standard. As we are not aware of any Gold Standard for this task, we had to manually build it. Two computer science graduate students, English-speakers, affinity with Twitter and adopters of *hashtags*, created the Gold standard. They agreed in 97% of the tokens. A third student resolved the cases of disagreements.

4.2. *Experimental setup and results*

We developed #hashtagtokenizerembedding the segmentation algorithm and the computation of the score (see Section 3)as well as the following resources:

- English vocabulary used in [18] with 27000 valid words
- Lemmatizer - The Stanford CoreNLP Toolkit [19]
- Encyclopedic knowledge base – DBPedia 3.9
- English corpus for the matrix of co-occurrence – Stanford WebBase Project [20]

Three evaluation scenarios were created, which are presented below.

- **BASELINE** approach – hashtag segmentation based on capitalization of words. For instance, #FergusonDecision, generates two tokens based on the capital letters 'F'e 'D' which are the indicatives of boundaries to left and to right of each token;

- **VOCabulary** + **WR** – segmentation uses a vocabulary of valid English words [18], the lemmatizer of the Stanford CoreNLP toolkit [19], and computes the WordRank Score (without Wikipedia) .
- **VOCabulary** + **DBPedia** + **WR** – the same as the previous using also Wikipedia.

Table 3 presents the results in each scenarioin terms of accuracy (based on the number of hashtags that have been correctly tokenized when compared to the Gold Standard).

Table 4. *Accuracy of the #hashtagtokenizerin the three evaluation scenarios*

Evaluation Scenario	Accuracy
BASELINE	63.9%
VOC+WR	58.3%
VOC+DBP+WR	73.2%

When we analyze the results, we can see that scenario VOC+DBP+WR (Vocabulary+DBPedia+WordRank) shows a gain of 25.5% of accuracy compared to scenario VOC+WR (Vocabulary+WordRank – without DBPedia). This result fortifies our claim that the use of encyclopedic knowledge improves the accuracy of hashtags tokenization (our first work hypothesis). Indeed the use of Wikipedia as a knowledge base is more and more a consensus since its dynamism, large scope and reliability [21]. Our analyses have shown that the mention to people, entities, brands, places and events is frequent in *hashtags*. Moreover, these emerge as memes requiring knowledge bases with high capacity to treat with volatile terms. Another benefit of using Wikipedia is that the meaning of the tokens is already defined leveraging the word sense disambiguation task. We know that the BASELINE scenario (using capitalization as a boundary for word separation) is naïve and useful only as an initial reference. However we did not find available similar approaches for comparisons.

Another formulated hypothesis is that the use of a lexicon and of a vocabulary of the language narrow the number of segmentation options thus optimizing the model. In fact, when we remove the vocabulary of the process of tokenization, the number of word hypothesis increases substantially. Without this resource the method has generates 16,486,141 options instead of 284,506 for the case the vocabulary is used. For the some hashtags such as #sledgehammervideopremiere, #mentionpeopleyouarethankfulfor and #asklittlemixalittlequestion more than 3 million options were generated.

We have also done a qualitative analysis from the cases in which the method failed. Most of the problems occur with hashtags containing numbers such as #63notout, #16days, #iphone6, #500aday (almost 20% of the errors). The other major source of bad inferences are due to acronyms such as #superstarRK, #AFCvBOR, #SEAvsSF) (17% of the errors). These problems will guide our future investigations.

5. CONCLUSION

In this paper we propose a generic model for *hashtag* tokenization based on a lexicon and a world knowledge bases, and on an unsupervised word segmentation algorithmused for languages where there is no visual representation of word boundaries in a text (e.g. Chinese or Japanese languages). The *Hashtag* Tokenizerpipeline searches, firstly, to identify tokens that express valid words of a lexicon, or names of people, organizations, brands, locations, etc., in an encyclopedic knowledge base such as Wikipedia. Thereafter, it searches for the segmentation option that maximizes the WordRank score, which captures the limits to the left and right of valid word hypothesis through a co-occurrence matrix of co-occurrence of words (external boundaries), combined with a value of internal boundary, which indicates the most likely separation point of a determined word combination.

Our research hypotheses were that the use of encyclopedic knowledge improves the accuracy of the approach, and that the use of a lexicon and language vocabulary optimizes the model performance. We have evaluated the accuracy of the proposed model by comparing the tokens produced by our approach to a Gold Standard produced by human taggers. The best evaluation scenario, whichused an English vocabulary, the DBpedia as encyclopedic knowledge base, and the WordRank score, presented an accuracy of 73.2%, with a gain of 25.5% of accuracy compared to scenario without DBPedia. This experimental evaluation provided real scenarios of assessing the challenges to discover the components words of a *hashtag*. Almost 37% of bad inferences were due *hashtags* containing numbers and acronyms. These problems will guide our future investigations.

REFERENCES

1. Avontuur, T., Balemans, I., Elshof, L., van Noord, N. & van Zaanen, M. 2012. Developing a part-of-speech tagger for Dutch tweets. *Computational Linguistics in the Netherlands Journal*, 2, 34-51.
2. Tsur, O. & Rappoport, A. 2012. What's in a hashtag?: Content based prediction of the spread of ideas in microblogging communities. In proceedings of the *WSDM'12, The Fifth ACM International Conference on Web Search and Data Mining* (pp. 643-652).
3. Maynard, D. & Greenwood, M. A. 2014. Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis. Diana Maynard. *Proceedings of LREC 2014*, Reykjavik, Iceland.
4. Pak, A. & Paroubek, P. 2010. Twitter as a corpus for sentiment analysis and opinion mining. *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010)* (pp. 1320-1326), Valletta, Malta.
5. Bontcheva, K., Derczynski, L., Funk, A., Greenwood, M.A., Maynard, D., Aswani & N. TwitIE. 2013. An open-source information extraction pipeline for Microblog text. In proceedings of the *International Conference on Recent Advances in Natural Language Processing, ACL.*
6. Cunha, E., Magno, G., Comarela, G., Almeida, V., Gonçalves, M.A., Benevenuto, F. 2011. Analyzing the dynamic evolution of hashtags on Twitter: A language-based approach. *Proceedings of the Workshop on Languages in Social Media* (pp. 58-65), Jun 23-23, Portland, Oregon.

7. Chen, S., Xu, Y. & Chang, H. 2011. A simple and effective unsupervised word segmentation approach. *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, *AAAI 2011*, San Francisco, California, USA, Aug 7-11.
8. Gimpel, K., Schneider, N., O'Connor, B. et al. 2011. Part-of-speech tagging for twitter: Annotation, features and experiments. In proceedings of the *49th Annual Meeting of the Association for Computational Linguistics*: *Short Papers*; Portland, OR, USA, ACL New Brunswick, NJ, USA.
9. Toutanova, K., Klein, D., Manning, C. & Singer, Y. 2003. Feature-rich part- of-speech tagging with a cyclic dependency network. *Proceedings of the HLT-NAACL; Edmonton, Canada, North American Chapter of the Association for Computational Linguistics (NAACL)*, (pp. 252-259).
10. Ritter, A., Clark, S., Mausam & Etzioni, O. 2011. Named entity recognition in tweets: An experimental study. In *proc. Of Empirical Methods for Natural Language Processing (EMNLP)*, Edinburgh, UK.
11. Cunningham, H. et al. 2011. Text processing with GATE (Version 6). University of Sheffield Department of Computer Science. ISBN 0956599311.
12. Firth, J. R. 1968. A synopsis of linguistic theory, 1930-1955. In John R. Firth (Ed.), *Selected Papers of JR Firth*, 1952-59. Indiana University Press, Bloomington (pp. 168-205).
13. Harris, Z. 1968. *Mathematical Structures of Language*. New York, USA: Wiley.
14. Landauer, T. K., Dumais, S. T. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104/2, 211-240.
15. Burgess, C., Livesay, K. & Lund, K. 1998. Explorations in context space: Words, sentences, discourse. *Discourse Processes*, 25, 211-257.
16. Sun, M., Shen, D. & Tsou, B. K. 1998. Chinese word segmentation without using lexicon and handcrafted training data. In proceedings of the *17th International Conference on Computational Linguistics*, 2, 1265-1271.
17. Carter, S., Weerkamp, W. & Tsagkias, E. 2013. Microblog language identification: Overcoming the limitations of short, unedited and idiomatic text. *Language Resources and Evaluation Journal*.
18. Han, L., Kashyap, A. L., Finin, T., Mayfield, J. & Weese, J. 2013. UMBC_EBIQUITY-CORE: Semantic textual similarity systems.

In proceedings of the *Second Joint Conference on Lexical and Computational Semantics. Association for Computational Linguistics*.

19. Manning, C. D., Surdeanu, M., Bauer, J. et al. 2014. The stanford core NLP natural language processing toolkit. In proceedings of *52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.
20. Stanford WebBase project. http://bit.ly/WebBase. Accessed in January, 31, 2015.
21. Zang L. J., Cao, C. & Cao, Y. N. et al. 2013. A survey of commonsense knowledge acquisition. *Journal of Computer Science and Technology*, 28/4, 689-719. DOI 10.1007/s11390-013-1369-6.

VLÁDIA PINHEIRO
PROGRAMA DE PÓS-GRADUAÇÃOEM INFORMÁTICA APLICADA –
UNIVERSIDADE DE FORTALEZA
AV. WASHINGTON SOARES, 1321,
FORTALEZA, CEARÁ, BRASIL.
E-MAIL: <VLADIACELIA@UNIFOR.BR>

RAFAEL PONTES
PROGRAMA DE PÓS-GRADUAÇÃOEM INFORMÁTICA APLICADA –
UNIVERSIDADE DE FORTALEZA
AV. WASHINGTON SOARES, 1321,
FORTALEZA, CEARÁ, BRASIL.
E-MAIL: <RAFAELLPONTES@GMAIL.COM>

VASCO FURTADO
PROGRAMA DE PÓS-GRADUAÇÃOEM INFORMÁTICA APLICADA –
UNIVERSIDADE DE FORTALEZA
AV. WASHINGTON SOARES, 1321,
FORTALEZA, CEARÁ, BRASIL.
E-MAIL: <VASCO@UNIFOR.BR>