Zipf's and Benford's laws in Twitter hashtags

José Alberto Pérez Melián
Univ. Politècnica de València
Camí de Vera s/n, 46022
València, Spain
jopeme@inf.upv.es

J. Alberto Conejero
IUMPA-UPV
Univ. Politècnica de València
Camí de Vera s/n, 46022
València, Spain
aconejero@upv.es

Cèsar Ferri
DSIC
Univ. Politècnica de València
Camí de Vera s/n, 46022
València, Spain
cferri@dsic.upv.es

Abstract

Social networks have transformed communication dramatically in recent years through the rise of new platforms and the development of a new language of communication. This landscape requires new forms to describe and predict the behaviour of users in networks. This paper presents an analysis of the frequency distribution of hashtag popularity in Twitter conversations. Our objective is to determine if these frequency distribution follow some well-known frequency distribution that many real-life sets of numerical data satisfy. In particular, we study the similarity of frequency distribution of hashtag popularity with respect to Zipf's law, an empirical law referring to the phenomenon that many types of data in social sciences can be approximated with a Zipfian distribution. Additionally, we also analyse Benford's law, is a special case of Zipf's law, a common pattern about the frequency distribution of leading digits. In order to compute correctly the frequency distribution of hashtag popularity, we need to correct many spelling errors that Twitter's users introduce. For this purpose we introduce a new filter to correct hashtag mistake based on string distances. The experiments obtained employing datasets of Twitter streams generated under controlled conditions show that Benford's law and Zipf's law can be used to model hashtag frequency distribution.

1 Introduction

Twitter is a microblogging social network launched in 2006 with 310 million active users per month and where 340 million tweets are daily generated[1]. By sending short messages called tweets of up to 140 characters, users can insert text, pictures, videos and links to interact with other users over the network. Twitter users can interact between them by using the @ symbol followed by the username they want to mention. They can also classify tweets in more than one category or theme by using *hashtags* (alphanumeric strings preceded by #). Hashtags are created by users. Some of them propagate and thrive while others are restricted to a few mentions and die. The most popular hashtags reach out what is called the trending topic list, who shows the most popular hashtags used at the moment. Popularity is considered either at a local level or worldwide. In this sense, the authors of (Ma et al., 2012) present a method to predict hashtag success. Hashtags are extremely popular in Twitter. Some studies have analysed how to extract hashtags from a microblogging environment (Efron, 2010). Other works apply Diffusion of Innovation (DoI) to model hashtag life cycle (Chang, 2010). However, to the best of our knowledge, there are not studies about the frequency distribution of hashtag popularity in Twitter conversations. In this work, our goal is to analyse Twitter datasets in order to discover if the the frequency of hashtags popularity follow some of the distribution laws that are very common in many types of data presented in the social sciences. Specifically, we study Benford's law and Zipf's law.

Benford's law (Benford, 1938), also known as the first-digit law, characterises the distribution of digits in large datasets. This law takes into account that in many natural occurring systems the frequency of number's first digits is not evenly dis-

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

Proceedings of the Student Research Workshop at the 15th Conference of the European Chapter of the Association for Computational Linguistics, pages 84–93, Valencia, Spain, April 3-7 2017.

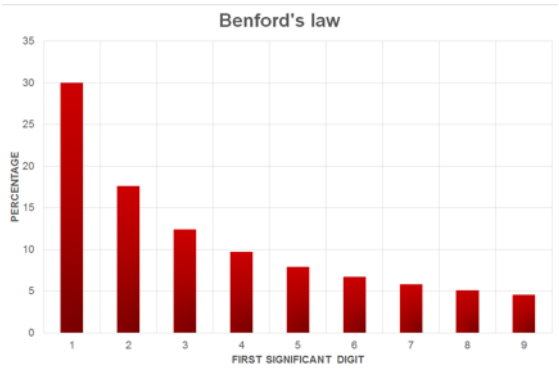

Figure 1: First Significant Digit probabilities calculated by Benford's law

tributed. Benford observed that numbers with 1 as first digit were observed far more often than those starting with 2, 3 and so on. The probability *P* of a number *d* having a particular non-zero first digit is given by formula 1.

$$P(d) = \log_{10}\left(1 + \frac{1}{d}\right) \quad (1)$$

For instance: if we have the number 81291, the First Significant Digit (FSD) is 8, the second is 1 and so on. Figure (1) shows the probabilities for the first significant digit distribution. The probability to find a 1 in the first position is about 30%, while the probability to find a 9 is around 4.6%. Some authors have applied Benford's law to forensic account (Durtschi et al., 2004), where an anomalous data distribution in the first significant digits can lead to detect fraud. It has been also applied to social networks by counting friends and followers distributions in Facebook, Twitter and many more networks (Golbeck, 2015). Other fields where Benford's law has been applied are: crime statistics (Hickman and Rice, 2010), electoral fraud (Bërdufi, 2013; Battersby, 2009), genome data (Friar et al., 2012) and macroeconomic data (Müller, 2011). For a recent account on other computer approaches for studying social networks, we refer the reader to (Kurka et al., 2016).

A related empirical law is Zipf's law. In fact, Benford can be seen as a special case of Zipf's law. Zipf confirmed that given a corpus with word frequencies of a language, the frequency of each word is inversely proportional to its position in the ranking of word's frequencies, see an updated reference in (Zipf, 1949). Both ranking and frequency distributions follow an inverse relationship who can be approximated by formula (2), where P_n represents the frequency of a word sorted in the *n*-th position with the exponent *a* very near to 1. Some applications of Zipf's law can be seen in (Powers, 1998; Popescu, 2003; Huang et al., 2008).

$$P_n \sim \frac{1}{n^a} \quad (2)$$

In this work we have considered, as corpus sets, hashtags appearing in some collection of tweets. The frequency in which they appear coincides with the number of times every tweet is mentioned. Therefore, in order to test Zipf's law on each dataset, we rank hashtags in the order from most to least relevant. For carrying out these analysis we have considered two different datasets that are described in Section 2. These datasets are processed in Section 3 in order to bring together hashtags with certain plausible typesetting mistakes or that were expected to refer to the same topic. Additionally, we also have optimised the process of joining similar hashtags in every dataset in order to drastically reduce computing times. Once the frequency of every hashtag is computed, in Section 4 we analyse the distribution of these frequencies in order to test whether Zipf's and Benford's law are satisfied. Conclusions are reported in Section 5.

2 Data Extraction

In this section, we summarise the process of collecting and extracting the datasets that is going to be employed in the experiments. Tweets of the datasets have been downloaded by means of the twitter API service [2]. This API provides programmatic access to Twitter data. Tweets are extracted in JSON format, and in every tweet we can find 26 different features [3]. In this work we only employ the field *["entities"]["hashtags"]* that contains the list of hashtags mentioned on the tweet and help us to count the total number of mentions of hashtags in a dataset.

The code for the use of Twitter API functions as well as for the data management has been developed in Python. This programming language provides a huge set of libraries for API connection and data management.

After we get the complete list of hashtags included in the dataset, we need to standardise and

[2] https://dev.twitter.com/overview/api

[3] https://dev.twitter.com/overview/api/tweets

	Users	Tweets	With #	Unique #
Argentina	650	635765	89643	44235
Chile	650	625739	63387	60262
Colombia	650	616046	144352	52248
Spain	650	623670	176167	76762
Mexico	650	624161	138631	66955
Peru	650	621325	144561	65156
Venezuela	650	610692	173906	59839
Total	4550	4357398	930647	425457

Table 1: Information about dataset *Hispatweets*. Number of users, number of tweets, number of tweets that contain hashtags and number of distinct hashtags.

normalise it in order to analyse correctly the hashtag distribution. The first step of this process consists in converting all the text in lower case characters. Given that the analysed tweets are in Spanish, we need to avoid the confusion that accents and some of the letters of the Spanish alphabet could produce[4]. Concretely, we remove accents and diaeresis from vowels, and the character *ñ* is converted into *n*.

In this work, we use two different datasets: *Hispatweets* and *Elecciones*. In the following points we summarise the information about these datasets.

2.1 Dataset *Hispatweets*

The dataset *Hispatweets* contains tweets from seven countries where different types of Spanish is spoken: Argentina, Chile, Colombia, Spain, Mexico, Peru and Venezuela. This dataset was generated in order to study the different features of the Spanish that is used in Twitter in each one of these countries. For that goal, 650 users of each country were selected and a set of tweets generated by these users were downloaded. Information about the creation of this dataset can be found in (Fabra-Boluda, 2016). The dataset is available in the following url: `https://s3.amazonaws.com/cosmos.datasets/hispatweets-populated.zip`.

In Table 1 we include some information about this dataset. In total, there are 4357398 tweets distributed almost uniformly among the seven countries. The presence of hashtags in the tweets is not uniform. Spain is the country where tweets contain more hashtags, since 21.36% of the tweets have at least one hashtag. The last column contains the number of different hashtags after the standardisation process.

[4]Some users tend to avoid the use of accents in Twitter hashtags.

Hashtag	Users
#PartidoPopular	Mariano Rajoy - @marianorajoy
	Soraya Saenz - @Sorayapp
#Ciudadanos	Albert Rivera - @Albert_Rivera
#PSOE	Pedro Sánchez - @sanchezcastejon
#Podemos	Pablo Iglesias - @Pablo_Iglesias_
#IzquierdaUnida	Alberto Garzón - @agarzon

Table 2: Hashtags and users employed in the dataset *Elecciones*.

2.2 Dataset *Elecciones*

The dataset *Elecciones* is formed by tweets collected during the 2015 Spanish General Election campaign on December 2015. Specifically, the tweets were stored during the period of the election campaign that started on 1/12/2015 and finished on 22/12/2015. For every day in this period, a Python script was executed every eight hours to download tweets referring some hashtags related to the main parties and tweets mentioning political leaders that were involved in the electoral process. Table 2 shows the exact terms that were explored for extracting the tweets. Summing up, this dataset is formed by 256293 tweets that contain 171650 hashtags (7950 distinct hashtags are distinguished).

3 Hashtag identification

After removing special characters from the hashtags, we observed that most of them had a low number of mentions, in many cases due to spelling errors on them. For instance: the hashtag *#7deldebatedecisivo* used for one of the debates for the 2015 Spanish General Election had a high number of mentions. Around them we find with hashtags like *#7ddebatedeviscisivo* or *#7deldevate* who had few mentions (both containing spelling errors).

For studying distributions of hashtags mentions in Twitter conversations, it is important if we are able to detect and correct in some way this kind of problems in hashtag identification. One possibility could be the use of automatic spell checkers in order to detect and correct spelling mistakes. Nevertheless, this solution is not feasible in this context for some reasons. Mainly because hashtags usually concatenate words, and strings without separators between the words are ambiguous and cannot be parsed correctly in many cases. This problem has been defined in NLP as compound

splitting (Srinivasan et al., ; Koehn and Knight, 2003). Additionally, in many cases hashtags contain acronyms, slang words or proper nouns, and these are not easily identified by compound splitting techniques and spell checkers.

Given these limitations, we have adopted a different approach based on the similarity of hashtags. We assume that in many cases if two hashtags are very similar (i.e, the similarity between the two terms is above a certain threshold α), they can be joined to be accounted as the same term. Therefore we need to measure similarities between terms. There is a plethora of different metrics that allow to estimate the distance between strings (Cohen et al., 2003). We have applied three string distances, *Levenshtein* distance, *Jaro* distance and *Jaro-Winkler* distance. These measures are implemented in the *python-Levenshtein*[5] library, written in Python. For a detailed description of these string distance metrics we refer to (Naumann and Herschel, 2010). A comparison between the differences in their application can be found in (Cohen et al., 2003). In this work we have used four levels for α: 0.95, 0.90, 0.85 and 0.80. Using smaller values can lead to group hashtags that are not very similar among them.

Table 3 shows an example of the measures of the string distances applied to some hashtags. Note that a measure of 1 indicates closeness similarity and 0 means no similarity at all.

Hashtag 1	Hashtag 2	Levenshtein	Jaro	Jaro-Winkler
#20delecciones	*#20democracia*	0.3846	0.6773	0.8064
#20delecciones	*#20dediciembre*	0.3846	0.7019	0.8211
#7deldebatedecisivo	*#7deldebateadecisivo*	0.9473	0.9824	1.0000
#7deldebatedecisivo	*#7deldebatedecsivo*	0.9445	0.9618	1.0000
#canarias	*#valencia*	0.2500	0.5834	0.5834
#marianorajoy	*#pedrosanchez*	0.0834	0.3889	0.3889

Table 3: String metrics between hashtags for different examples of dataset *Elecciones*

In order to unify similar hashtags the first approach could be to calculate distances between all hashtags of a dataset. However this process implies a quadratic complexity on the number of hashtags. Concretely, if we have n hashtags, we need to compute $\frac{n(n-1)}{2}$ pairwise distances. For instance, given the *Elecciones* dataset, with 7950 unique hashtags, we would need to compute 31597275 string distances. Due to its large complexity, this complete method is not feasible for medium size datasets. As a result, we propose in this paper a filter to group similar hashtags based on the alphabetical order:

1. We sort the n hashtags list in alphabetical order
2. We calculate the distance between one hashtag and the nearest k neighbours in the list.
3. Given a level of similarity α, starting from the beginning and in alphabetical order, we group hashtags with a similarity more or equal than α.

Note that using alphabetical order and computing distances between neighbours we only need n pairwise distance computations. This approximation has important limitations. For instance, if the spelling error is located in the first characters, the algorithm will not group properly this hashtag. We can also improve the performance of the filter using more than one neighbour (factor k) in the step 2 and 3, but this also could increase the time complexity of the filter. This k factor could be established depending on the size of the dataset. In this work we only consider the nearest neighbour, $k = 1$.

4 Experiments

After the correct identification of hashtags, in this section we study the distribution of hashtags for both datasets. In particular we analyse if the frequency distribution of hashtags follow Benford's and Zipf's law.

4.1 Zipf's law

First, we compare the frequency distribution of hashtags with respect to Zipf's law.

4.1.1 Dataset *Hispatweets*

If we analyse separately the frequency distribution of hashtags for each one of the countries of the dataset *Hispatweets*, we observe that all of them present a close distribution with respect to Zipf's law. Table 4 includes the regression line (considering a log-log scale) induced for the frequency distribution and the coefficient of determination R^2 computed with respect to Zipf's law distribution. Since all values are close to -1, we can see that the frequency distribution of hashtags follow approximately Zipf's law. Figure 2 shows an example of the line induced by regression with respect to the ideal Zipf's law.

[5]https://pypi.python.org/pypi/python-Levenshtein/0.12.0

Country	Regression line	R^2
Argentina	$-1.1011x + 4.4794$	-0.9549
Chile	$-0.9538x + 4.4206$	-0.9617
Colombia	$-0.9550x + 4.3778$	-0.9641
Spain	$-0.9496x + 4.5036$	-0.9628
Mexico	$-0.8612x + 4.0208$	-0.9527
Peru	$-0.8953x + 4.1562$	-0.9549
Venezuela	$-1.0394x + 4.8159$	-0.9617

Table 4: Regression lines induced from frequency of hashtags for each country of *Hispatweets* dataset. Coefficient of determination R^2 computed with respect to Zipf's law distribution

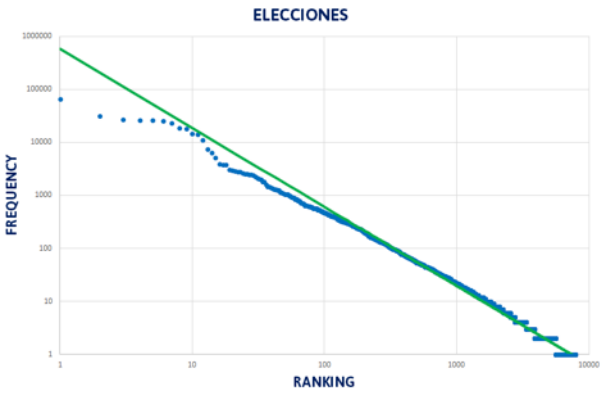

Figure 2: Regression lines induced from frequency of hashtags for Spain with respect to Zipf's law distribution (considering a log-log scale).

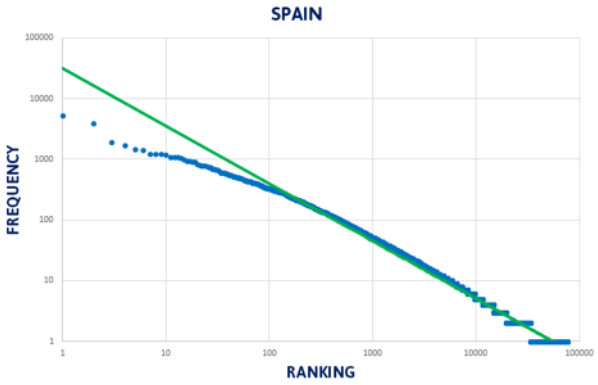

Figure 3: Regression lines induced from frequency of hashtags for dataset *Elecciones* with respect to Zipf's law distribution (considering a log-log scale).

4.1.2 Dataset *Elecciones*

For this dataset the distribution of the frequency of hashtags is again very close to Zipf's law distribution. Using a log-log scale, the distribution is approached by linear regression to a the following line: $-1.4909x + 5.7644$. Here, the Coefficient of determination $R^2 = -0.9879$ is extremely close to -1. Figure 3 includes the line induced by regression for this dataset with respect to the ideal Zipf's law.

4.2 Benford's law

After analysing the Zipf's law on the two datasets with succesful results, here we study if the distributions of the frequency of hashtags follow Benford's law.

4.2.1 Dataset *Hispatweets*

Table 5 shows the percentage of each FSD (*First Significant Digit*) for the seven countries of the dataset. We also include in the first row the theoretical percentage for each FSD according to the Benford's law. We can observe that, for all cases, there are important differences between the computed FSD values and the theoretical values expected by Benford's law. The disparity is specially great for the case FSD = 1, mainly because we have detected a gross number of hashtags that only appear once. In part, this is caused because sometimes Twitter users introduce unintended mistakes when writing hashtags, and then, they are accounted as different. In order to correct these wrong hashtags we try to unify some of them according to the procedure explained in

Section 3. We have tested three edition distances: Levenshtein, Jaro and Jaro-Winkler. In short, Levenshtein distance counts the number of editions (insertions, deletions, or substitutions) needed to convert one string into the other. Jaro gives a measure of characters in common, being no more than half the length of the longer string in distance, with consideration for transpositions. The modification included in Jaro-Winkler takes the idea that differences near the start of the string are more significant than differences near the end of the string, see for instance (Naumann and Herschel, 2010). All of them range from 0 to 1, with 1 representing the case of coincidence.

According to our results, this last distance is the most valid to unify similar hashtags. In Table 6 we include the values of the FSD for the case of Spain and different values of α. According to these results, $\alpha = 0.8$ is the value that obtains better results when we compare the distribution of FSD with respect to the FSD according of the Benford's law. Similar results have been obtained for the rest of countries.

4.2.2 Dataset *Elecciones*

We also have a similar result in the case of dataset *Elecciones*. Table 7 includes the computed distribution of FSD without filtering hashtags, and applying the filter based on Jaro-Winkler distance for different values of α. Again, we find a situation with a high number of hashtags with just one appearance. After applying the filter, we reduce this situation by joining hashtags that probably were different because of type-writing errors. As in the previous dataset, $\alpha = 0.8$ is the value that obtains more similar results to the theoretical estimates of FSD according to Benford's law.

4.3 Analysis of results

According to the results presented in the analysed datasets, we can observe that when we study a significant number of tweets, the distribution of the FSD approaches to Benford's law, specially if we apply a filter step that joins similar hashtags. In order to assess this conclusion, we introduce in this part some experiments where we measure the similarity between the computed distribution of FSDs with respect to the theoretical expected FSD distribution defined by Benford's law.

In Table 9 we include some measures for evaluating the similarity between the computed and theoretical distribution of FSDs. These are:

- **Pearson Correlation**: a measure for estimating the linear dependence between two variables. The estimated value is between +1 (total positive linear correlation) and -1 (total negative linear correlation). Correlation 0 indicates no linear correlation.

- χ^2: This metric is defined as the difference of the computed distribution with respect to the theoretical distribution:

$$\chi^2 = \sum_{d=m}^{9} \frac{(P_{obs}(d) - P_t(d))^2}{P_t(d)} \qquad (3)$$

  where:

  - $P_t(d)$ is the theoretical frequency and $P_{obs}(d)$ is the observed frequency
  - m refers to the analysed digit. Here we study the first digit, thus $m = 1$.

  Since χ^2 estimates the difference between distributions, lower values of the metric indicates distributions closer to Benford's law. According to (Nigrini, 2012), we can assume that a distribution does not follow Benford's law for the first digit (FSD) if $\chi^2 > 15.507$ (confidence 95%), and if $\chi^2 > 20.090$ (confidence 99%).

- **Mean absolute deviation (MAD)** : The average absolute deviation (or mean absolute deviation) is a summary statistic of dispersion. MAD estimates the average of the absolute deviations from a theoretical distribution. For Benford's law, it is computed in the following way:

$$MAD = \frac{1}{9} \sum_{d=1}^{9} |P_{obs}(d) - P_t(d)| \qquad (4)$$

  For making a hypothesis contrat, we consider as null hypothesis that a distribution follows Benford's law. Since χ^2 estimates the difference between distributions, lower values of the metric indicates distributions closer to Benford's law. According to (Nigrini, 2012), we use this metric to estimate different values of conformity of a distribution with respect to Benford's law. These ranges are presented in Table 8.

	1	2	3	4	5	6	7	8	9	Total
FSD Benford	30.01%	17.60%	12.40%	9.69%	7.91%	6.69%	5.79%	5.11%	4.57%	100%
FSD Argentina	62.54%	13.37%	6.69%	4.14%	2.41%	1.80%	1.18%	1.03%	0.83%	100%
FSD Chile	60.88%	19.19%	6.95%	4.51%	2.86%	2.11%	1.47%	1.16%	0.87%	100%
FSD Colombia	59.45%	19.41%	7.36%	4.71%	3.05%	2.30%	1.49%	1.28%	0.95%	100%
FSD Spain	60.17%	19.89%	7.00%	4.56%	2.74%	2.03%	1.52%	1.16%	0.92%	100%
FSD Mexico	63.53%	18.61%	6.41%	3.98%	2.52%	1.84%	1.33%	0.96%	0.82%	100%
FSD Peru	62.60%	19.15%	6.78%	4.08%	2.47%	1.84%	1.23%	1.05%	0.79%	100%
FSD Venezuela	58.71%	19.55%	7.48%	4.89%	3.01%	2.01%	1.61%	1.32%	1.12%	100%

Table 5: Percentage of each FSD *First Significant Digit* for the seven countries of the dataset *Hispatweets*. The first row contains the expected Percentage of each FSD according to Benford's law.

	Jaro-Winkler Distance								
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
FSD Benford	30.01%	17.60%	12.40%	9.69%	7.91%	6.69%	5.79%	5.11%	4.57%
FSD Spain	60.17%	19.89%	7.00%	4.56%	2.74%	2.03%	1.52%	1.16%	0.92%
$\alpha = 0.95$	54.57%	20.48%	8.46%	5.58%	3.60%	2.61%	1.96%	1.47%	1.27%
$\alpha = 0.90$	49.78%	20.70%	9.48%	6.55%	4.40%	3.13%	2.41%	1.92%	1.62%
$\alpha = 0.85$	44.74%	20.51%	10.62%	7.18%	5.25%	4.02%	2.95%	2.67%	2.06%
$\alpha = 0.80$	39.89%	20.56%	11.12%	8.33%	5.92%	4.72%	3.45%	3.49%	2.55%

Table 6: Percentage of each FSD for Spain in dataset *Hispatweets* applying a filter based on Jaro-Winkler distance and different values of α. The first row contains the expected percentage of each FSD according to Benford's law. The second row contains the percentage of each FSD without hashtag the union filter.

	Jaro-Winkler Distance								
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
FSD Benford	30.01%	17.60%	12.40%	9.69%	7.91%	6.69%	5.79%	5.11%	4.57%
FSD *Elecc.*	40.39%	25.99%	8.97%	9.23%	3.91%	4.50%	2.54%	2.69%	1.77%
$\alpha = 0.95$	39.00%	24.61%	9.85%	10.07%	4.45%	4.88%	2.51%	2.79%	1.83%
$\alpha = 0.90$	38.08%	23.24%	10.60%	9.79%	4.92%	5.09%	3.24%	3.07%	1.98%
$\alpha = 0.85$	36.35%	21.88%	11.55%	9.80%	5.42%	5.51%	3.61%	3.49%	2.40%
$\alpha = 0.80$	33.8%	20.49%	12.56%	9.70%	6.61%	5.81%	4.13%	4.27%	2.63%

Table 7: Percentage of each FSD for dataset *Elecciones* applying a filter based on Jaro-Winkler distance and different values of α. The first row contains the expected percentage of each FSD according to Benford's law. The second row contains the percentage of each FSD without the hashtag union filter.

Range	Conformity Level
0.000 to 0.006	High
0.006 to 0.012	Good
0.012 to 0.015	Medium
0.015 or more	Low

Table 8: Range of critical values and the corresponding conformity level for Mean absolute deviation and Benford's law on the first significant digit.

Distribution	Correlation	χ^2	MAD
Spain	0.9699	51.41	0.071
Spain + J-W $\alpha = 0.8$	0.9966	7.48	0.028
Elecciones	0.9835	23.78	0.038
***Elecc.* + J-W** $\alpha = 0.8$	0.9979	2.72	0.014

Table 9: Pearson Correlation, χ^2 statistics and Mean absolute deviation (MAD) between observed distribution of FSD and theoretical distribution of FSD according to Benford's law. We include original datasets and datasets after applying the Jaro-Winkler Distance filter.

If we analyse the results of Table 9, we can observe that for all cases correlation obtain high values (greater than 0.92). We can see that corrected versions for both datasets increase the correlation with respect to Benford's law.

A similar behaviour is observed in χ^2 statistics. The Jaro-Winkler Distance filter is able to unify numerous hashtags and then the similarity with respect to to Benford's law is drastically increased. If we consider the test proposed by (Nigrini, 2012), and the corrected version of the dataset, the hypothesis that distributions does not follow Benford's law cannot be rejected.

Finally, considering Mean absolute deviation (MAD), we find the same pattern. Jaro-Winkler Distance filter reduces the distance between distributions. In this case, the test proposed by (Nigrini, 2012) determines that *Spain* dataset has a low similarity with respect to Benford's law, and the *Elecciones* dataset (corrected version) has a medium similarity. These results are in some cases contradictory with respect to the conclusions observed with χ^2 statistics, and indicate that MAD test seems to be more strict that χ^2 test.

5 Conclusions

Benford's Law is useful to estimate the probabilities of highly likely or highly unlikely frequencies of numbers in datasets. Those who are not aware of this experimental law and intentionally manipulate numbers are susceptible to be discovered by the comparison with respect to Benford's Law. We find examples of this use in electoral processes, accounting fraud detection, scientific fraud detection...

In this paper, Benford's and Zipf's laws have been testing against hashtag frequency on datasets of tweets. A similar analysis has been recently checked for the case of followers distributions in Facebook, Twitter (Golbeck, 2015). We confirm that the distribution of hashtag frequency follows a power law, as Zipf's law expects. That is, few hashtags achieve a high number of mentions, and most of them lack of impact with few repetitions . The source of this dispersion is probably the lack of control of Twitter on the use of hashtags. The social network permits that hashtags can be created without any restriction, and it also lacks of a recommender system for the generation of hashtags. In fact, we detected an irregular number of hashtags with just one mention. Many of these hashtags are spelling mistakes of Twitter users. In order to mitigate this dispersion, we defined a union filter based on string distances that is able to group filters based on their similarity. We use alphabetical order of hashtags in order to reduce time complexity of the cluster algorithm. The comparison of three string distances *Levenshtein*, *Jaro* and *Jaro-Winkler* indicates that the last one, *Jaro-Winkler*, obtains the better performance in correcting hashtags.

We also analyse the distribution of the first significant digit of the hashtag frequencies with respect to Benford's law. Experiments on the datasets of tweets considering three different metrics: Pearson Correlation, χ^2 and Mean absolute deviation, reveal that this law is approximately followed by the distribution of the first significant digit of the hashtag frequencies, specially when we apply a group filter based on the *Jaro-Winkler* distance in order to correct spelling errors in hashtags. In order to give statistical significance to our research, we apply some of the tests provided by (Nigrini, 2012) that allow to verify the level of conformity of a frequency distribution with respect to Benford's law. According to the results,

χ^2 test returns high level of conformity, while considering Mean absolute deviation (MAD), we get medium and low level of conformity. These two tests are in some way contradictory and show that MAD test seems to be more strict that χ^2 test.

As future work, we propose the improvement of the hashtag unification filter by improving the mechanism for detecting similarities between hashtags. We will also study the applicability of the experimental laws on bigger tweet datasets, where, likely, the levels of conformity will be greater.

Acknowledgments

We thank Francisco Almenar Pedrós, José Francisco García Cantos, and Mirella Oreto Martínez Murillo for providing us the dataset *Elecciones*. We also thank Raül Fabra Boluda, Paolo Rosso, and Francisco Manuel Rangel Pardo for providing us the dataset *Hispatweets*. This work has been partially supported by the EU (FEDER) and Spanish MINECO grant TIN2015-69175-C4-1-R, LOBASS and the REFRAME project, granted by the European Coordinated Research on Long-term Challenges in Information and Communication Sciences Technologies ERA-Net (CHIST-ERA), and funded by MINECO in Spain (PCIN-2013-037) and by Generalitat Valenciana PROMETEOII/2015/013.

References

Stephen Battersby. 2009. Statistics hint at fraud in iranian election. *New Scientist*, 24.

Frank Benford. 1938. The law of anomalous numbers. *Proceedings of the American Philosophical Society*, pages 551–572.

Dorina Bërdufi. 2013. Statistical detection of vote count fraud: 2009 albanian parliamentary election and Benford's law. *Academic Journal of Interdisciplinary Studies*, 2(8):379.

Hsia-Ching Chang. 2010. A new perspective on twitter hashtag use: Diffusion of innovation theory. *Proceedings of the American Society for Information Science and Technology*, 47(1):1–4.

William W. Cohen, Pradeep Ravikumar, and Stephen E. Fienberg. 2003. A comparison of string distance metrics for name-matching tasks. In *International Joint Conference on Artificial Intelligence (IJCAI) 18, Workshop on Information Integration on the Web*.

Cindy Durtschi, William Hillison, and Carl Pacini. 2004. The effective use of Benford's Law to assist in detecting fraud in accounting data. *Journal of forensic accounting*, 5(1):17–34.

Miles Efron. 2010. Hashtag retrieval in a microblogging environment. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 787–788. ACM.

Raül Fabra-Boluda. 2016. Identificación de la variedad del lenguaje para la mejora del geoposicionamiento en social media. Master's thesis, Universitat Politècnica de València. `http://users.dsic.upv.es/~prosso/resources/FabraMSc.pdf`.

James L. Friar, Terrance Goldman, and Juan Pérez-Mercader. 2012. Genome sizes and the Benford distribution. *PLoS One*, 7(5):e36624.

Jennifer Golbeck. 2015. Benford's Law Applies to Online Social Networks. *PLOS ONE*, 10(8):e0135169.

Matthew J. Hickman and Stephen K. Rice. 2010. Digital analysis of crime statistics: Does crime conform to Benford's law? *Journal of Quantitative Criminology*, 26(3):333–349.

Shi-Ming Huang, David C. Yen, Luen-Wei Yang, and Jing-Shiuan Hua. 2008. An investigation of Zipf's law for fraud detection. *Decision Support Systems*, 46(1):70–83.

Philipp Koehn and Kevin Knight. 2003. Empirical methods for compound splitting. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*, pages 187–193. Association for Computational Linguistics.

David Burth Kurka, Alan Godoy, and Fernando J. Von Zuben. 2016. Online social network analysis: A survey of research applications in Computer Science. *Preprint*, page arXiv:1504.05655.

Zongyang Ma, Aixin Sun, and Gao Cong. 2012. Will this # hashtag be popular tomorrow? In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 1173–1174. ACM.

Hans Christian Müller. 2011. Greece was lying about its budget numbers. *Forbes*, 12.

Felix Naumann and Melanie Herschel. 2010. An introduction to duplicate detection. *Synthesis Lectures on Data Management*, 2(1):1–87.

Mark Nigrini. 2012. *Benford's Law: Applications for forensic accounting, auditing, and fraud detection*, volume 586. John Wiley & Sons, Hoboken, NJ.

Ioan-Iovitz Popescu. 2003. On a Zipf's law extension to impact factors. *Glottometrics*, 6:83–93.

David M. W. Powers. 1998. Applications and explanations of Zipf's law. In *Proceedings of the joint conferences on new methods in language processing and computational natural language learning*, pages 151–160. Association for Computational Linguistics.

Sriram Srinivasan, Sourangshu Bhattacharya, and Rudrasis Chakraborty. Segmenting web-domains and hashtags using length specific models. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*.

George Kingsley Zipf. 1949. *Human behavior and the principle of least effort: An introduction to human ecology*. Ravenio Books, Cambridge, MA.