TWORPUS – An Easy-to-Use Tool for the Creation of Tailored Twitter Corpora

Alexander Bazo, Manuel Burghardt, and Christian Wolff

University of Regensburg, Media Informatics Group, 93040 Regensburg, Germany

Abstract. In this paper we present Tworpus, an easy-to-use tool for the creation of tailored Twitter corpora. Tworpus allows scholars to create corpora without having to know about the Twitter *Application Programming Interface* (API) and related technical aspects. At the same time our tool complies with Twitter's "rules of the road" on how to use tweet data. Corpora may be composed in various sizes and for specific scenarios, as the Tworpus interface provides controls for filtering and gathering customized collections of tweets, which may serve as the basis for subsequent analyses.

Keywords: Twitter API, web corpora, social media corpora, corpus tool, corpus creation

1 Introduction

Since the days of the linguist and stenographer Friedrich Wilhelm Kaeding, who with the help of hundreds of assistants manually created and analyzed a corpus of approximately 11 million words from 1891-1897 (Kaeding, 1898), *corpus linguistics* has gained a significant boost from the developments in computerization and data processing. The *World Wide Web* (WWW) plays an important role in this development, as it provides an abundance of machine-readable, freely and ubiquitously available texts (Fletcher, 2012). With the rise of social media platforms like *Facebook.com* and *Twitter.com* in the past decade we also have access to large amounts of user generated content, which allows insights into actual (computer-mediated) language samples to empirically analyze linguistic, political or social issues.

1.1 Web Corpora

Although the web provides large scale and easily accessible language data, it has been discussed whether such data can be used as a corpus without concern. Kilgarriff & Grefenstette (2003) conclude that the web can generally be used as a corpus, but that it depends on the context and type of research question whether the web is a good and suitable resource. They also note, that the requirement of *representativeness*[1] is a general problem of any kind of corpus, and that the web as a corpus is only representative of itself. A prominent example for corpora built from the web can be found in *WaCky (Web-as-Corpus kool initiative)*[2], a working group of researchers interested in using the web as a corpus for linguistic studies (Baroni et al., 2009). The authors also provide an extensive review of other web corpus projects in the related work section.

1.2 Social Media Corpora

Social media have become an important source for collecting current language usage data (Beißwenger & Storrer, 2008). As most social media services are still being operated via the World Wide Web, corpora drawn from them may be categorized as a special kind of web-based corpora.[3] At the same time they are an important driving force of language change, as computer-mediated communication typically differs from other communication channels (Androutsopoulos, 2004, Crystal, 2007, ch. 24, Squires, 2010).

For many social media platforms like *Facebook*, *YouTube* or *Twitter*, APIs are available that allow to draw large samples of textual data for corpus creation. Twitter is the most prominent and dominant type of microblogging services, which allows individuals to publish short messages (”tweets”) of up to 140 characters (”SMS of the Web”) that can be read by others subscribing to the respective Twitter channel. Among the people with the most followers are idols of popular culture like Justin Bieber or Lady Gaga, each of whom have more than 35 million followers on Twitter.[4] ”Hashtags” (e.g. #gscl2013) can be used as descriptors within the tweet message, allowing to search for thematically related tweets. Tweets may also be ”retweeted”, i. e. a user can republish an existing tweet from another user for his own set of followers. In 2012, Twitter had more than 500 million users.[5]

Analysis of tweets has quickly become an interdisciplinary field of research. For Twitter alone, the cross-disciplinary bibliographic database *Web of Knowledge* (WOK) lists 90 entries with a publication time range from 2009 to 2013.[6] Among the studies looking into Twitter data are as diverse research questions

[1] An extensive discussion on ”representativeness as the holy grail” in the context of web corpora can be found in Leech (2007).

[2] http://wacky.sslmit.unibo.it, accessed April 10, 2013

[3] This attribution may change as more dedicated social media apps are used on smartphones with no explicit connection to the web.

[4] Cf. the Twitter monitoring platform *twitaholic*, http://twitaholic.com/, accessed April 17, 2013.

[5] For more detailed information on Twitter, see the comprehensive English Wikipedia article on Twitter, which has been marked as a *good article* (cf. http://en.wikipedia.org/wiki/Twitter, accessed April 17, 2013).

[6] ISI WOK search ”Topic=(twitter) AND Topic=(microblog*)” (http://www.webofknowledge.com, accessed April 14, 2013)

as analyzing tweets as electronic words of mouth in an E-Commerce context (Jansen et al., 2009), sentiment detection in tweets (Bae & Lee, 2012) or using Twitter for analyzing dialect variations in American English (”Twitalectology”, Russ, 2012).

2 Corpus Creation on Twitter

Although millions of tweets are published every day, it can be challenging for scholars to get access to this data in a way that enables them to build corpora tailored for their specific research questions. Twitter's *Application Programming Interface* (API) for accessing the continuous stream of tweets and the corresponding *”rules of the road”*[7] present some major technical and legal hurdles.

In the following section we will discuss some common approaches for building Twitter corpora, and how Twitter's developer agreement affects them.

2.1 The Twitter APIs

Twitter offers two different APIs that allow the searching and streaming of tweet collections. Developers may retrieve tweets by querying Twitter's *REST API* with different parameters. While it is possible to search for certain hashtags or query terms, this API does not randomize the sample, but rather returns the first tweets that match the query. Also, it is limited in size and timespan.

The *Streaming API* provides direct access to a continuous stream of current tweets. Free of charge access to this stream is limited to a random sample of approximately one percent of all tweets, while access to all tweets is charged and exclusively granted to selected customers. It is also possible to filter the free streaming sample by using certain query parameters, for instance hashtags or user names. Both APIs require the user to implement *GET* or *POST* requests and to interpret the returning result, which can be received in *JSON* or *XML* format. In order to request a larger number of tweets via the API, the user is required to authenticate as a registered Twitter user by means of the *OAuth* mechanism. To integrate the Twitter APIs in existing software tools, developers can make use of a collection of third party bridges and libraries for different programming languages. Scholars not familiar with the described aspects of programming have to rely on existing corpora or given tools to create tailored tweet collections.

2.2 Twitter Corpora

While many linguists have become familiar with utilizing ready-to-use tools to process and query large amounts of language data, only few of them are able to

[7] These rules are also known as the Twitter *developer agreement*. They basically describe the terms and policies for using and redistributing data acquired by the Twitter API (cf. https://dev.twitter.com/terms/api-terms, accessed April 10, 2013).

cope with technically more demanding, rather abstract interfaces to such data, for instance Twitter's *REST API* or the *Streaming API*. Therefore it seems obvious that those who are capable of accessing Twitter data via the APIs should create corpora and share them with the rest of the research community. However, since Twitter changed their developer agreement in 2010 it is no longer allowed to redistribute any tweet messages outside the Twitter platform. Consequently, projects like the *Edinburgh Twitter Corpus* (Petrović et al., 2010) with approximately 100 million tweets are no longer available. Recent Twitter corpora[8] may only be distributed as a list of numerical identifiers that allow to reconstruct the tweets and their corresponding metadata (McCreadie et al., 2012). To retrieve the actual text data, researchers have to use existing crawler applications or create their own implementations of the Twitter API.

Besides the technical and legal obstacles that occur during the creation and distribution of Twitter corpora, researchers also have to accept the lack of customization and personalization of such corpora, as most existing corpora are limited to certain languages, time periods or topics. At the same time, filtering generic tweet collections in a way that makes the language data suitable for answering specific research questions, may result in samples that are too small to derive meaningful observations and interpretations.

2.3 Twitter Corpus Tools

Available web-based tools for the creation of Twitter corpora come with various restrictions and may not be tailored to the specific needs of a particular research project. Such tools allow to monitor current tweets that contain certain hashtags (e.g. *TweetTag*[9]) or that match certain words, phrases or queries (e.g. *Tweet-Archivist*[10]). These tools do not maintain an internal database, but rather rely on *Twitter's Search API* to fetch matching tweets on the fly. One major drawback of such tools is that published tweets can only be restored up to a limit of 2.000 tweets or for a time span of 6-7 days[11]. Although both tools support live monitoring of current and upcoming tweets (continuous searches for selected queries each hour), the total corpus size is limited to 50.000 items (TweetArchivist) or a running time of one day (TweetTag). We designed Tworpus to encounter these restrictions of existing tools.

3 Description of Tworpus

Tworpus provides an easy-to-use interface that allows scholars to build large, tailored Twitter corpora. Our tool does not require the user to query the Twitter

[8] For instance the *TREC Microblogging Corpus* for the years 2011 and 2012 (cf. http://trec.nist.gov/data/tweets/, accessed April 10, 2013).

[9] http://www.tweet-tag.com/index.php, accessed April 10, 2013

[10] http://www.tweetarchivist.com, accessed April 10, 2013

[11] https://dev.twitter.com/docs/using-search, accessed April 10, 2013

stream via an abstract API and at the same time meets the Twitter developer agreement.

Tworpus consists of three main parts (cf. Fig. 1), which are described in more detail in the following sections. (1) A server-based crawler component links into Twitter's free streaming API and stores identifiers and corresponding metadata (but not the tweets themselves) in a *MySQL* database. Users can access the database via (2) a web interface (*corpus creation GUI*) and build customized corpora in the fashion of a list of tweet identifiers (IDs). The tweet ID sets can be filtered by metadata parameters such as language or length, which are stored with each ID in the database. As the reconstruction of large corpora may take several hours we provide (3) a desktop tool (*corpus extraction tool*) that allows to import a list of tweet IDs and subsequently builds a full corpus by automatically downloading the tweets in TXT or XML-format.

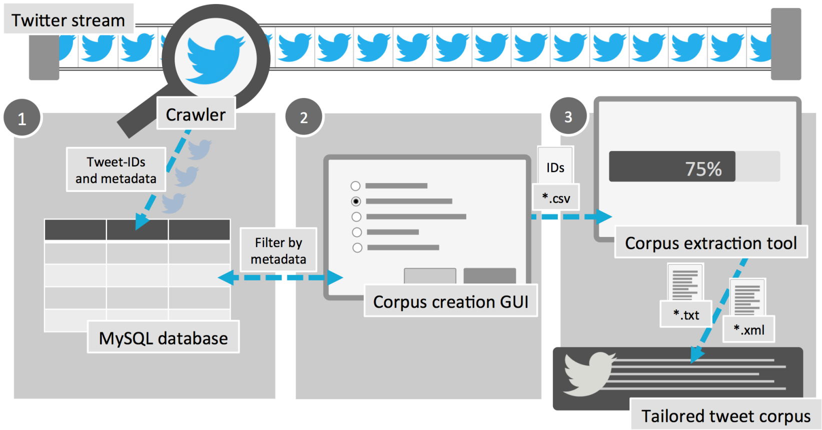


Fig. 1. Overview of the basic workflow and the three main components of Tworpus.

3.1 Dataset, Crawler and Language Detector

Dataset — In the current implementation all data is stored in a single MySQL database with several relations. Schema and engine are optimized for fast query processing in large tables. The database and the crawler are hosted on a state of the art desktop PC running the unix-based operating system *Debian Squeeze*. An *Apache* web server provides access to the web interface for corpus creation. Our current test dataset contains approximately eight million tweets. Based on the crawling speed, we expect the dataset to grow continuously by approximately 600.000 tweets a day. As the free streaming sample is limited to around 1% of *all* tweets, Twitter uses an algorithm to provide a randomized sample.

Unfortunately, Twitter does not provide any information on how the randomization algorithm works[12]. Given the huge amount of overall tweet production per day,[13] we believe that the sample provided will be large enough for many relevant research issues.

Crawler — The server-based crawler continuously fetches and processes Twitter messages. To connect to the stream we use *Twitter4J* (Yamamoto, 2007), a *Java* bridge to Twitter's APIs. We collect the tweets' actual message content as well as the metadata provided by Twitter (e.g. date and time). In addition, we count the number of *characters* and *words* for each tweet and store this information in the database. With each tweet being crawled in real-time from the stream after its very release, it is not possible to collect information about retweets and favorites, which obviously require the tweet to be published for a certain period of time. We will discuss solutions to dynamically populate these fields afterwards in section 3.3. Each tweet collected by the crawler is stored in the database with the following attributes:

- IDs for tweet and user,
- word and character count,
- date and time[14],
- location of origin,
- use of hashtags,
- and language.

Storing the unique tweet and user IDs which Twitter allocates to each tweet allows later reconstruction via different approaches.

Language detection — While most attributes are unproblematic and can be stored with clear-cut values, the language information available from Twitter unfortunately is rather ambiguous and unpredictable, as it is based on the settings in the user profile, where each author can define his preferred language. The *preferred language* may however differ from the language that is used in actual tweets, as users tend to write in different languages, or even mix up different languages in the same tweet. To address this problem, we integrated a *language detection library* (Nakatani, 2010) for Java, which uses n-gram frequency profiles to detect the actual language of a tweet. Even though a large number of language profiles is available in this library, several problems occur when using it on Twitter data: As tweets are by nature rather short text fragments, with a maximum

[12] https://dev.twitter.com/docs/faq#6861, accessed April 10, 2013

[13] *TechCrunch* gives the number of one billion tweets every 2.5 days as of June 2012 (cf. http://techcrunch.com/2012/07/30/analyst-twitter-passed-500m-users-in-june-2012-140m-of-them-in-us-jakarta-biggest-tweeting-city/, accessed April 17, 2013).

[14] The timestamp is saved together with a *UTC* offset to allow determination of the local creation time.

length of 140 characters, language detection is challenging, but nevertheless feasible (Gottron & Lipka, 2010). The length restriction for tweets entails heavy use of abbreviations and a generally more telegraphic and fragmentary style of writing. As a result, Bergsma et al. (2012) note that tweets are very heterogeneous in terms of style, and that they are often misspelled and ungrammatical. We have also learned that tweets encoded in non-Latin alphabets may cause additional troubles.

Against this background our recent implementation for detecting the actual language of a tweet can only be considered prototypical and remains an open problem that needs to be addressed in future work. As a result of a series of detection pretests, the current version of TWORPUS uses the language detection library to identify the following eight languages[15]: Dutch, English, French, German, Italian, Portuguese, Spanish and Turkish.

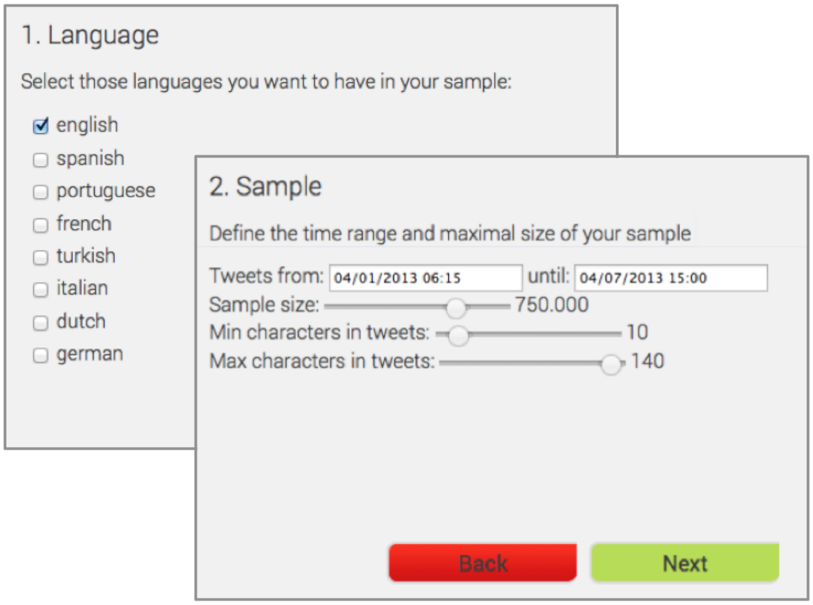
3.2 Designing a Tailored Twitter Corpus

The corpus creation web interface is realized by means of HTML5 and JavaScript. Users may build a tailored sub-corpus from our dataset (cf. Fig. 2), filtering tweets by the attributes that are stored in the TWORPUS database. The *geolocation* of a tweet or the origin of its author are not implemented as a valid filter criterion. Pretests have shown that the geolocation information that can be gathered from the user profiles is in many cases missing or obviously not realistic, as they do not refer to actual places and cannot be proved to be the actual origin of the tweet. This observation is backed up by Morstatter et al., who found that geolocation information is only available for approximately 3% of the streaming data.

As we aim to provide an easy-to-use tool for the creation of Twitter corpora, the user is guided through the process of corpus creation step by step. Each design step is explained in detail and context sensitive tool tips provide detailed information about the selected filter criteria and possible effects on the sample and its validity. Such information is important as some fields may be misinterpreted (e.g. "language").

Currently, the sample size may range from 10.000 to 1.000.000 tweets. In case the sample size is smaller than the number of available tweets that match the filter criteria, we randomize the sample by using an optimized implementation of SQL's *RANDOM()* function. Once the design parameters for the tweet corpus have been entered, the corpus can be downloaded for further investigation. Complying with Twitter's developer agreement, we only provide a *CSV* file with tweet IDs that allow to build a corpus of actual tweets. For use in later analyses this file also contains information from our database, including word and character counts as well as the detected language. Other metadata will be restored while (re-)building the corpus (cf . section 3.3).

[15] The reliability of the language detection is closely connected to the problems described above, including multilingualism and stylistic as well as orthographic aspects of language use in microblogging contexts.



1. Language

Select those languages you want to have in your sample:

- ☑ english
- □ spanish
- □ portuguese
- □ french
- □ turkish
- □ italian
- □ dutch
- □ german

2. Sample

Define the time range and maximal size of your sample

Tweets from: 04/01/2013 06:15 until: 04/07/2013 15:00

Sample size: 750.000

Min characters in tweets: 10

Max characters in tweets: 140

Fig. 2. Corpus creation GUI: Tailored corpora may be designed with regard to language, time and date, sample size and length of tweets.

3.3 Building the Corpus

To download the actual tweets, users need the corpus extraction tool, which can be obtained from the Tworpus website. It allows to import the previously created list of IDs and automatically fetches the corresponding tweets to build the actual corpus (cf. Fig. 3).

The tool generates a folder for the corpus, which contains all tweets in plain text and in XML format. While the plain text files only store the message text of each tweet, the XML file (cf. Fig. 4) contains the respective metadata that is stored in the database as well as up-to-date information fetched from Twitter while downloading. The plain text corpus can be analyzed using "distant reading" (cf. Moretti, 2007) tools such as *Voyant*[16], while the XML-encoded corpus may be investigated with existing query tools such as *XAIRA*[17] or *eXist*[18].

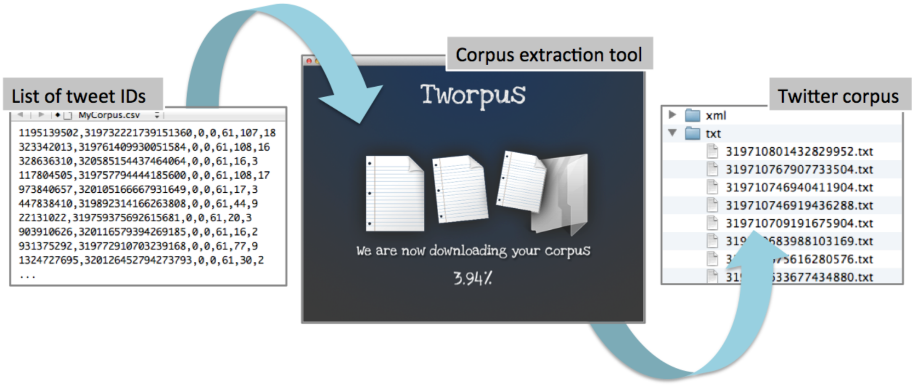
As downloading the tweets via Twitter's REST API would limit Tworpus to a maximum speed of 720 tweets per hour[19], we decided to take a different ap-

[16] http://voyant-tools.org, accessed April 10, 2013

[17] http://xaira.sourceforge.net/, accessed April 10, 2013

[18] http://exist-db.org/exist/apps/homepage/index.html, accessed April 10, 2013

[19] Twitter restricts downloading by GET request to 180 tweets in a rate limited window with a duration of 15 minutes (cf. https://dev.twitter.com/docs/rate-limiting/1.1/limits, accessed April 10, 2013).



```
1195139502,319732221739151360,0,0,61,107,18
323342013,319761409930051584,0,0,61,108,16
328636310,320585154437464064,0,0,61,16,3
117804505,319757794444185600,0,0,61,108,17
973840657,320105166667931649,0,0,61,17,3
447838410,319892314166263808,0,0,61,44,9
22131022,319759375692615681,0,0,61,20,3
903910626,320116579394269185,0,0,61,16,2
931375292,319772910703239168,0,0,61,77,9
1324727695,320126452794273793,0,0,61,30,2
...
```

Fig. 3. The corpus extraction tool.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
<tweet id=308877023269507072>
<user id=579178700>
<screenname>@NewsStocktonCA</screenname>
<fullname>Stockton News</fullname>
</user>
<date>1:49 AM 5 March 13</date>
<retweets>0</retweets>
<favoured>0</favoured>
<text chars=132 words=17 lang=en>Clean air grants offered by local
district: The Yolo-Solano Air Quality Management District Tuesday ...
http://q.gs/3aOXV #stockton</text>
</tweet>
```

Fig. 4. XML representation of downloaded tweet.

proach: The basic idea is that any published tweet is already available as HTML content on Twitter.com. Each HTML-tweet can be accessed via a unique URL, which contains IDs of the user and his tweet. McCreadie et al. (2012) outline a simple approach for parsing published tweets via the web[20]. By establishing multiple parallel connections on one machine, the speed for downloading the tweets can be improved significantly. In Tworpus we have implemented 30 parallel threads, which allow to download 10.000 tweets in less than 10 minutes using a broadband Internet connection. Obviously, larger corpora with sample sizes of one million tweets will still take some time to build. Therefore, we implemented a mechanism that automatically resumes a previously paused download process.

McCreadie et al. (2012) also note the dynamic nature of Twitter content as an important aspect that has to be considered: As users can delete tweets or hide them from public access after their release, our dataset may well contain identifiers for tweets that are not available for actual reconstruction via the extraction tool. A list of 10.000 identifiers may result in an actual corpus with a smaller size, as our tool will recognize if a tweet is not available for download and exclude it from the corpus. Future releases of Tworpus will implement a *back channel*, allowing data communication between the download client and the tweet database. When the download client detects unavailable tweets it can query the database for substitutions to accomplish the intended corpus size. The same back channel could be used to gather information about retweets and favorites.

It is important to note, that a corpus extracted from the same list of identifiers at a later point in time, might contain slightly different or modified tweets than a corpus from an earlier extraction, because Twitter users can change or delete their tweets at any time[21]. A corpus extraction in Tworpus therefore should be treated as a snap-shot of the dynamic and ever-changing *twittersphere*.

4 Outlook

Although in the current implementation Tworpus is still in its beta testing phase, the web interface for our dataset may be accessed via the corresponding website[22]. The corpus creation tool is also available on the website, and may be downloaded for different operating systems. We strongly encourage other scholars to test and use Tworpus and are happy to receive feedback on the tool and the dataset. At the same time we intend to work on known limitations and problems of Tworpus in order to substantially contribute to research with social media corpora.

[20] As described by McCreadie et al., this technique is also used for downloading the *TREC Microblogging Corpus*.

[21] This issue is also described in the TREC microblog track guidelines (cf. https://sites.google.com/site/microblogtrack/2012-guidelines, accessed June 5, 2013).

[22] http://tools.mi.ur.de/tworpus

In the long term, Tworpus aims at supporting linguistic research. Most of the current literature on tweet analysis focuses on social or political issues of Twitter usage or on *pragmatic* aspects of Twitter language like opinion and sentiment analysis. Little has been published on more traditional linguistic aspects such as lexical, morpho-syntactic or orthographic aspects of Twitter usage. As Twitter does not explicitly mark the actual language used in tweets (which is a vital criterion for linguistic studies), we have implemented a language detection on our own. However, our current language detector still needs to be improved for meeting the special characteristics of grammar, style and spelling in tweets. We are currently planning a crowdsourced study to manually label language, spelling errors, non grammatical terms and other unique characteristics to build language profiles that facilitate language detection on Twitter.

In addition, we are planning to integrate hashtags into the corpus building process. This would enable researchers to generate corpora that do not only match certain language or time span criteria, but also to aggregate tweets for a specific topic.

References

Androutsopoulos, J. K.: Online-Gemeinschaften und Sprachvariation. Soziolinguistische Perspektiven auf Sprache im Internet. Zeitschrift für germanistische Linguistik, 31(2), 173-197 (2004)

Bae, Y., & Lee, H. (2012). Sentiment analysis of twitter audiences: Measuring the positive or negative influence of popular twitterers. Journal of the American Society for Information Science and Technology, 63(12), 2521-2535.

Beißwenger, M., & Storrer, A.: Corpora of Computer-Mediated Communication. In: A. Lüdeling & M. Kytö (Eds.), Corpus Linguistics. An International Handbook (pp. 292-308). Berlin / New York: Mouton de Gruyter (2008)

Baroni, M. , Bernardini, S. , Ferraresi, A. & Zanchetta, E.: The WaCky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora. Language Resources and Evaluation 43 (3), 209-226 (2009)

Bergsma, S., McNamee, P., Bagdouri, M., Fink, C. & Wilson, T.: Language identification for creating language-specific Twitter collections. Proceedings of the Second Workshop on Language in Social Media, LSM 12, Montreal, Canada, Association for Computational Linguistics, 65-74 (2012)

Crystal, D.: How Language Works. London, Penguin (2007).

Fletcher, W. H.: Corpus analysis of the world wide web. In C. A. Chapelle, ed., Encyclopedia of Applied Linguistics. Wiley-Blackwell (2012)

Gottron, T. & Lipka, N.: A Comparison of Language Identification Approaches on Short, Query-Style Texts. Advances in information retrieval: 32nd European Conference on IR Research, Berlin, Springer 611-614 (2010)

Jansen, B. J., Zhang, M., Sobel, K., & Chowdury, A.: Twitter power: Tweets as electronic word of mouth. Journal of the American Society for Information Science and Technology, 60(11), 2169-2188 (2009).

Kaeding, F. W.: Häufigkeitswörterbuch der deutschen Sprache. Steglitz near Berlin, self published (1898)

Kilgarriff, A. & Grefenstette, G.: Introduction to the Special Issue on the Web as Corpus. Computational Linguistics, 29, 333-347 (2003)

Leech, G.: New resources, or just better old ones? The Holy Grail of representativeness. In: Hundt, M., Nesselhauf, N. & Biewer, C. (Eds.): Corpus Linguistics and the Web, Amsterdam et al., Editons Rodopi B.V., 133-149 (2007)

McCreadie, R., Soboroff, I., Lin, J., Macdonald, C., Ounis, I. & McCullough, D.: On building a reusable Twitter corpus. Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, SIGIR 12, New York, NY, USA, ACM, 1113-1114 (2012)

Nakatani, S.: Language Detection Library for Java (website), http://code.google.com/p/language-detection, accessed, April 10, 2013

Moretti, F.: Graphs, Maps, Trees: Abstract Models for a Literary History. London, Verso (2007).

Morstatter, F., Pfeffer, J., Liu, H. & Carley, K. M.: Is the Sample Good Enough? Comparing Data from Twitter's Streaming API with Twitter's Firehose. ICWSM 2013, to be published, online. http://www.public.asu.edu/ fmorstat/paperpdfs/icwsm2013.pdf, accessed, June 3, 2013

Petrović, S., Osborne, M. & Lavrenko, V.: The Edinburgh Twitter corpus. Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media, WSA 10, Los Angeles, California, Association for Computational Linguistics, 25-26, (2010)

Russ, B.: Examining Large-Scale Regional Variation Through Online Geotagged Corpora. Presentation, 2012 Annual Meeting of the American Dialect Society, online. http://www.briceruss.com/ADStalk.pdf, accessed, April 17, 2013

Squires, L.: Enregistering internet language. Language in Society 39 (2010), 457-492, URL: http://dx.doi.org/10.1017/S0047404510000412, accessed June 6, 2013

Yamamoto, Y.: Twitter4J. Java library for the Twitter API (website), http://twitter4j.org, accessed, April 10, 2013