POS Tagging for Arabic Tweets

Fahad Albogamy
School of Computer Science,
University of Manchester,
Manchester, M13 9PL, UK
albogamf@cs.man.ac.uk

Allan Ramsay
School of Computer Science,
University of Manchester,
Manchester, M13 9PL, UK
allan.ramsay@cs.man.ac.uk

Abstract

Part-of-Speech (POS) tagging is a key step in many NLP algorithms. However, tweets are difficult to POS tag because there are many phenomena that frequently appear in Twitter that are not as common, or are entirely absent, in other domains: tweets are short, are not always written maintaining formal grammar and proper spelling, and abbreviations are often used to overcome their restricted lengths. Arabic tweets also show a further range of linguistic phenomena such as usage of different dialects, romanised Arabic and borrowing foreign words. In this paper, we present an evaluation and a detailed error analysis of state-of-the-art POS taggers for Arabic when applied to Arabic tweets. The accuracy of standard Arabic taggers is typically excellent (96-97%) on Modern Standard Arabic (MSA) text; however, their accuracy declines to 49-65% on Arabic tweets. Further, we present our initial approach to improve the taggers' performance. By doing some improvements based on observed errors, we are able to reach 79% tagging accuracy.

1 Introduction

The last few years have seen an enormous growth in the use of social networking platforms such as Twitter in the Arab World. A study prepared and published by Semiocast in 2012 has revealed that Arabic was the fastest growing language on Twitter in 2011. People post about their lives, share opinions on a variety of topics and discuss current issues. There are millions of tweets daily, yielding a corpus which is noisy and informal, but which is sometimes informative. As a result, Twitter has become one of the most important social information mutual platforms. The nature of the text content of microblogs differs from traditional blogs. In Twitter, for example, a tweet is short and contains a maximum of 140 characters. Tweets also are not always written maintaining formal grammar and proper spelling. They are ambiguous and rich in acronyms. Slang and abbreviations are often used to overcome their restricted lengths (Java et al., 2007).

POS tagging is an essential processing step in a wide range of high level text processing applications such as information extraction, machine translation and sentiment analysis (Barbosa and Feng, 2010). However, people working on Arabic tweets have tended to concentrate on low level lexical relations which were used for shallow parsing and sentiment analysis such as (Mourad and Darwish, 2013; El-Fishawy et al., 2014). They do not use the standard linguistic pipeline tools such as POS tagging which might enable a richer linguistic analysis (Gimpel et al., 2011). The properties listed above of the microblogging domain make POS tagging on Twitter very different from their counterparts in more formal texts. It is an open question how well the features and techniques of NLP used on more well-formed data (e.g. in newswire domain) will transfer to Twitter in order to understand and exploit tweets. Therefore, we experimentally evaluate the performance of state-of-the-art POS taggers for MSA on Arabic tweets. POS tagging accuracy drops from about 97% on MSA to 49-65% on Arabic tweets. We also analyse their limitations and errors they made. Finally, we propose an approach to boost their performance and we are able to reach 79% tagging accuracy.

Our contributions in this paper are as follows:

1. Evaluating how robust state-of-the-art POS taggers for MSA are on Arabic tweets.
2. Identifying problem areas in tagging Arabic tweets and what caused the majority of er-

Proceedings of Recent Advances in Natural Language Processing, pages 1–8,
Hissar, Bulgaria, Sep 7–9 2015.

rors.

3. Boosting the taggers' performance on Arabic tweets by using pre- and post-processing techniques to address Arabic tweets' noisiness.

2 Related Work

POS tagging is a well-studied problem in computational linguistics and NLP over the past decades. This can be inferred from high accuracy of state-of-the-art POS tagging not only for English, but also most other languages such as Arabic, which reaches 97% for Arabic and English being at 97.32% (Gadde et al., 2011). However, the performance of standard POS taggers for English is severely degraded on Tweets due to their noisiness and sparseness (Ritter et al., 2011). Therefore, POS taggers for English tweets have been developed such as ARK, T-Pos and GATE TwitIE which reaches 92.8%, 88.4% and 89.37% accuracy respectively (Derczynski et al., 2013).

People working on Arabic tweets have tended to concentrate on lexical relations because a tagger that can actually work on this domain with an acceptance degree of accuracy, is yet to be developed (Elsahar and El-Beltagy, 2014). There has been relatively little work on building POS tools for Arabic tweets or similar text styles. (Al-Sabbagh and Girju, 2012; Abdul-Mageed et al., 2012) are strictly supervised approaches for tagging Arabic social media and they have assumed labelled training data. Their weakness is that they need a high quantity and quality of training data and this labelled data quickly becomes unrepresentative of what people post on Twitter. They also have been built specifically for dialectal Arabic and subjectivity and sentiment analysis.

Our work is, to best of our knowledge, the first step towards developing a POS tagger for Arabic tweets which can benefit a wide range of downstream NLP applications such as information extraction and machine translation. We evaluate the existing state-of-the-art POS tagging tools on Arabic tweets, with an intention of developing a POS tagger for Arabic tweets by utilising the existing standard POS taggers for MSA instead of building a separate tagger. We use pre- and post-processing modules to improve their accuracy. Then, we will use agreement-based bootstrapping on unlabelled data to create a sufficient amount of labelled training tweets that we can retrain our augmented version of Stanford on it.

3 Data Collection

There is a growing interest within the NLP community to build Arabic social media corpora by harvesting the web such as (Refaee and Rieser, 2014; Abdul-Mageed et al., 2012). However, none of these resources are publicly available yet. They also do not contain all phenomena of tweets as they appear in their original forms in Twitter and they have been built to be used mainly in sentiment analysis. Hence, we built our own corpus which preserves all phenomena of Arabic tweets. We used Twitter Stream API to crawl Twitter by setting a query to retrieve tweets from the Arabian Peninsula and Egypt by using latitude and longitude coordinates of these regions since Arabic dialects in these regions share similar characteristics and they are the closest Arabic dialects to MSA. We did not restrict tweets language to "Arabic" in the query since users may use other character sets such as English to write their Arabic tweets (Romanisation) or they may mix Arabic script with another language in the same tweets. Next, we excluded all tweets which were written completely in English. Then, we sampled 390 tweets (5454 words) from the collected set to be used in our experiments (similar studies for English tweets use a few hundred of tweets e.g. (Gimpel et al., 2011)).

4 Evaluating Existing POS Taggers

We evaluate three state-of-the-art publicly available POS taggers for Arabic, namely AMIRA (Diab, 2009), MADA (Habash et al., 2009) and Stanford Log-linear (Toutanova et al., 2003).

4.1 Gold Standard

A set of correctly annotated tweets (gold standard) is required in order to be able to appraise the outputs of POS taggers. Once we have this, we can compare the outputs of the POS taggers with this gold standard. Since there is no publicly available annotated corpus for Arabic tweets, we have created POS tags for Twitter phenomena (i.e. REP, MEN, HASH, LINK, USERN and RET for replies, mentions, hashtags, links, usernames and retweets respectively) and we manually annotated our dataset. To speed up manual annotation, we tagged tweets by using the taggers, and then we corrected the output of the taggers to construct a gold standard.

4.2 POS Tagging Performance Comparison

We compare three taggers on 390 tweets (5454 words) from our corpus. The performance of these taggers are computed by comparing the output of each tagger against the manually corrected gold standard. We use standard precision, recall and F-score as evaluation measures. The results for the AMIRA, MADA and Stanford which were trained on newswire text present poor success rates, for example, the precision (P) for AMIRA, MADA and Stanford on Arabic tweets are 60.2%, 65.8% and 49.0% respectively (see Table 1). These figures are far below the performance of the same taggers on well-formed genres such as PATB, where accuracy is around 96% for AMIRA and Stanford whereas MADA achieves over 97% accuracy. This huge drop in the accuracy of these taggers when applied to Arabic tweets warrants some analysis of the problem and of mistagged cases.

Tagger	Newswire	Arabic Tweets
AMIRA	96.0%	**60.2%**
MADA	97.0%	**65.8%**
Stanford	96.5%	**49.0%**

Table 1: POS tagging performance comparison

4.3 Error Analysis

We noticed that most of the mistagged tokens are unknown words. In this case, the taggers rely on contextual clues such as the word's morphology and its sentential context to assign them the most appropriate POS tags (Foster et al., 2011). We identified the unknown words that were mistagged and classified them into two groups: Arabic words and non-Arabic tokens (see Table 2 for more details).

Arabic words These are words which are written in Arabic, but which were assigned incorrect POS tags by the taggers. This category represents 73.5%, 68.1% and 79.2% of the total of mistagged items by AMIRA, MADA and Stanford respectively. We observed that words in this category have different characteristics and most of them are twitter phenomena. So, we classify them into subcategories as follows:

MSA words These are proper words which are used in well-formed text and part of MSA vocabulary, but which were assigned incorrect POS tags by the taggers. We observed that the accuracy of MSA words which are not noisy dropped from 96% for AMIRA, 96.5% for Stanford and 97% for MADA on newswire domain to 71.8%, 55% and 79.3% respectively on Arabic tweets. There are three possible reasons for that: 1) the context of MSA words being noisy, 2) text structure has been changed, for example, many function words are omitted in tweets and 3) the domain change between the Arabic Treebank corpus on which they were trained and tested and the Arabic tweets. For example, the word "عصينا" (disobey) was tagged *NN* by AMIRA,*noun* by MADA and *NNP* by Stanford but, in fact, it is a verb.

Concatenation In this classification, two or more words were connected to each other to form one token. So, the taggers struggled to label them. Users may connect words deliberately to overcome tweets restricted length or accidentally. In this experiment, the taggers mistagged all connected words in the subset. For example, the word "تأكدأن" was labelled *NN* by AMIRA, labelled *noun* by MADA and tagged *NNP* by Stanford. But, in fact, it is two words "تأكد" and "أن" connected together which are a verb and a conjunction respectively.

Repeated letters Words in this classification have one or more letters repeated. Users repeat letters deliberately to express subjectivity and sentiment. For example, the word "واااااااااقفيييييييين" (standing) was labelled *NNS* by AMIRA and Stanford and *noun* by MADA but , in fact, it is an adjective.

Named entities All of these words should be labelled proper noun by the taggers because they refer to person, place or organization, but they mistagged them since these words were not part of their training data. For example, the proper noun "مسلم" was tagged *NN* by AMIRA and Stanford and labelled *noun* by MADA.

Spelling mistakes It is not easy to know the intent of the user, but some words seem likely to have been accidentally misspelled. Most words belonging to this category were mistagged by the taggers. For example, the word "كثرة" was misspelled and it should be written as "كثرت" (abounded). AMIRA and Stanford tagged it *NN* and MADA labelled it *noun* but , in fact, it is a verb.

Slang It is one of Twitter phenomena. The words in this category are regarded as informal and are typically restricted to a particular context or group of people. They are often mistagged by

Tagger	Types of mistagged items	Arabic Words								Non-Arabic Tokens				
		MSA words	Concatenation	Repeated letters	Named Entities	Spelling mistakes	Slang	Characters deletion	Transliteration	Romanisation	Emoticons	Emoji	Foreign words	Twitter specific
AMIRA	% of Errors	53.3%	1.8%	0.8%	8.7%	0.6%	6.2%	0.9%	1.2%	1.0%	0.5%	2.8%	2.6%	19.6%
	Accuracy	71.8%	0.0%	40.0%	49.2%	35.0%	30.4%	16.7%	61.8%	21.4%	0.0%	0.0%	35.6%	0.0%
MADA	% of Errors	45.5%	2.1%	0.8%	8.5%	0.6%	7.1%	1.0%	2.4%	1.4%	0.5%	3.3%	3.9%	22.8%
	Accuracy	79.3%	0.0%	50.0%	57.0%	40.0%	32.0%	20.8%	35.3%	7.1%	0.0%	0.0%	17.2%	0.0%
Stanford	% of Errors	65.5%	1.4%	0.9%	3.2%	0.6%	6.4%	0.5%	0.8%	0.7%	0.4%	2.2%	2.4%	15.1%
	Accuracy	55.0%	0.0%	20.0%	75.7%	20.0%	7.2%	45.8%	67.6%	25.0%	0.0%	0.0%	21.8%	0.0%

Table 2: Errors percentage of each mistagged class and its accuracy

the taggers. For example, the slang word "شوف" is the counterpart of MSA word "انظر" which means *look!*.

Characters deletion Arabic users delete letters from words deliberately to overcome tweets restricted length or because they do not have enough time to write complete words. For example, the word "في" (at) was shorten to only one letter "ف". This word was tagged *PUNC* by AMIRA, *conj* by MADA and *CC* by Stanford but , in fact, it is a preposition.

Transliteration Arabic users borrow some words and multiwords abbreviations from English. They use their Arabic transliteration in Arabic tweets. For example, LOL in English (Laugh Out Loud) is written in Arabic as "لول" and "mix" in English is written in Arabic as "مكس" . AMIRA and Stanford tagged the translated form of mix as *NN* whereas MADA labelled them all as *noun* but, in fact, it is a verb.

Twitter-specific They are elements that are unique to Twitter such as reply, mention, retweet, hashtag and url. They represent 19.6%, 22.8% and 15.1% of the total of mistagged items by AMIRA, MADA and Stanford respectively. In fact, taggers mistagged all Twitter-specific elements in the experiment and they tokenised them in different ways. AMIRA uses punctuation as an indicator for a new token so replies, mentions, retweets and hashtags in tweets are broken into the indicator part (@ for replies, mentions and retweets and # for hashtags) and the remainder of them. Moreover, if the remainder part contains punctuation marks, AMIRA will split it further into parts. AMIRA also breaks urls into parts since they contain punctuation marks. In contrast, MADA and Stanford do not break all Twitter-specific elements into parts since they use the space as an indicator for a new token. MADA has one exception to this rule. If a hashtag started with an Arabic letter, then MADA breaks it into parts when punctuation is found. We notice that MADA always labels unsplitted Twitter-specific elements as nouns *noun* (see Table 3).

	AMIRA		**MADA/Stanford**	
Twitter element	Token	Tag	Token	Tag
@Moh_Ali	@	PUNC	@Moh_Ali	noun
	Moh	NN		
	_	PUNC		
	Ali	NN		

Table 3: Twitter element tokenised and tagged by taggers

Non-Arabic tokens This group contains the remaining twitter phenomena which are appear in Arabic tweets, but which are not written by using the Arabic alphabet. They represent 6.9%, 9.1% and 5.7% of the total of mistagged items by AMIRA, MADA and Stanford respectively. We classify them into subcategories based on their shared characteristics as follows:

Romanisation Arabic users tend to use Latin letters and Arabic numerals to write Arabic tweets because the actual Arabic alphabet is unavailable for technical reasons, difficult to use or they speak Arabic but they cannot write Arabic script. For example, the word 3ala which is the Romanised form of the Arabic word "على" was tagged *NN* by AMIRA, labelled *noun* by MADA and *CD* by Stanford but, in fact, it is a preposition.

Emoticons They are constructed by using traditional alphabetics or punctuation, usually a face expression. They are used by users to express their feelings or emotions in tweets. AMIRA and MADA break emoticons into parts during tokenisation processes and they deal with each part as punctuation so all emoticons lost their meaning.

For example, the emoticon (= was broken into two parts: ”(” (labelled *PUNC*) and ”=” (labelled *PUNC*). In contrast, Stanford does not break them into parts but it mistagged all of them.

Untagged emoji Emoji means symbols provided in software as small pictures in line with the text which are used by users to express their feelings or emotions in tweets. AMIRA and MADA omitted these symbols in the tokenisation stage and they did not tag them. For example, the heart symbol ♡ was omitted when tweets were tokenised by the taggers. In contrast, Stanford does not omit them but it mistagged all of them.

Foreign words Some Arabic tweets contain foreign words especially from English. These words may refer to events, locations, English hashtags or retweet of English tweets with comments written in Arabic. ”I’m at Arab Bank البنك العربي” this tweet is an example of this category. AMIRA and Stanford tagged foreign words in this tweet as ’I’m’ is a *VBD*, ’at’ is a *PUNC*, ’Arab’ is a *NN* and ’Bank’ as *NN* whereas MADA labelled them all as *noun*.

5 Improving POS Tagging Performance

Our experiments show that the taggers present poor success rates since they were trained on newswire text and designed to deal with MSA text. They fail to deal with Twitter phenomena. As a result, their outcomes are not useful to be used in linguistics downstream processing applications such as information extraction and machine translation in microblogging domain. Therefore, there is a need for a POS tagger which should take into consideration the characteristics of Arabic tweets and yield acceptable results.

Our goal is not to build a new POS tagger for Arabic tweets. The goal is to make existing POS taggers for MSA robust towards noise. There are two ways to do so, one is to retrain POS taggers on Arabic tweets and alter their implementation if needed, the other is to overcome noise through pre- and post-processing to the tagging. Our approach is based on both approaches. We combine normalisation and external knowledge to boost the taggers’ performance. Then, we will retrain Stanford tagger on Arabic tweets since its speed is ideal for tweets domain and it is only the retrainable tagger. However, we do not have suitable labelled training data to do so. Therefore, we will use bootstrapping on unlabelled data to create a sufficient amount of labelled training tweets.

5.1 Pre- and Post-processing

As seen in error analysis, unknown words (out-of-vocabulary tokens or OOV) represent a large proportion of mistagged tokens. We argue that normalisation and external knowledge will reduce this proportion which will improve the performance of the proposed tagger. Normalisation is the process of providing in-vocabulary (IV) versions of OOV words (Han and Baldwin, 2011). We create a mapping from OOV tokens to their IV equivalents by using suitable dictionaries and the original token is replaced with its equivalent IV token. External sources of knowledge such as regular expression rules, gazetteer lists and an output of English tagger are also used. The combination of normalisation and external knowledge is applied to text as pre- and post-processing steps.

Handling Concatenation Users may connect words deliberately to overcome tweets restricted length or accidentally. This forms tokens which all taggers struggle to tag them correctly. One approach to deal with these cases is to use a MSA dictionary. We constructed a MSA dictionary from 250k Arabic words which were extracted from news website[1]. We handle concatenation for a word in the corpus W as follows:

1. If the length of W is <= 5, then it is left as it is, since the average length of Arabic words is five letters (Mustafa, 2012).
2. Else, if W exists in the MSA dictionary, then it is left as it is, since it is a valid MSA word.
3. Else, if a part P of W exists in the MSA dictionary, then W is split into two parts P and the remainder and the same steps are applied to the remainder.

We apply the above algorithm on ”تأكدأن”. The length of this token is six characters, it is larger than the average length of Arabic words, so we check if it exists in the MSA dictionary, but it does not exist in the dictionary. Then we check if any part of it exists in the dictionary, we find ”تأكد” in the dictionary so we split the token into two parts ”تأكد” and the remaining characters and then we apply the algorithm on the second part. Because the length of the second part ”أن” is two characters, it is left as it is and the algorithm stops.

Handling Elongated Words We handle these

[1] http://sourceforge.net/projects/ar-text-mining/files/Arabic-Corpora/

cases by using the same MSA dictionary mentioned above. Given a word in the corpus W, we do the following steps:

1. If a word W exists in the MSA dictionary, then it is left as it is, even it contains repeated letters.
2. Else, a compressed form of it is constructed by removing any repetition in letters.

Handling Characters Deletion We have noticed that users tend to shorten closed-class lexical items more than other speech classes to overcome tweets restricted length since it is easy for recipients of tweets to recognise them. We handle these cases by detecting and replacing them by their IV equivalents.

Handling Slang We handle these cases by mapping slangs to their IV equivalents, but slang is an open class and it is difficult to detect all slangs in tweets domain. Therefore, we select the most frequent twenty slang words from 17k types in our corpus (10 million tokens) and map them to their IV equivalents.

Handling Twitter-specific Items We use regular expression rules to detect and tag Twitter-specific elements such as mentions, hashtags, urls and etc. by doing some pre-processing and then tagging and finally doing post-processing. Due to the space limit, we present the way we deal with hashtags: all the remaining Twitter elements are tagged in similar ways. First, we detected hashtags by using regular expression rules. Then, we removed the hashtag signs and underscores from raw tweets. Next, we tagged them by using AMIRA, MADA and Satnford. Finally, we inserted hashtag signs in their original place in tweets to indicate the beginning and the end of hashtags content as shown in Table 4.

Raw Tweet	حياتي فالبيت بقضيهاجنب الشاحن!! #جالاكسي _لا _تكلمني
MADA	... !,punc !,punc #,punc jAlAksy,noun _, noun **lA,verb**_,noun tklmny,verb
Preprocessing	حياتي فالبيت بقضيها جنب الشاحن !! جالاكسي لا تكلمني
MADA	... punc !,punc jAlAksy,noun **lA,part_neg** tklmny,verb
Postprocessing	... punc !,punc <**hash**> jAlAksy,noun lA,part_neg tklmny,verb </**hash**>

Table 4: Pre- and post-processing (tag hashtag's words)

In fact, the taggers not just mistagged Twitter elements, but they also mistagged some MSA words in the same tweets because the text is noisy and the taggers rely on contextual clues. By using the above approach, we are not just able to tag Twitter elements correctly but we also make the context less noisy so the taggers are more likely to tag MSA words correctly as "IA" word in Table 4.

Handling Named Entities These can be recognised by using gazetteer lists. We use ANERGazet[2] which a collection of three Gazetteers, (i) Locations: it contains names of continents, countries, cities, etc.; (ii) People: it has names of people recollected manually from different Arabic websites; and finally (iii) Organizations: it contains names of organizations like companies, football teams, etc..

Handling English Words Our focus is on Arabic tweets, but some of them contain English words. These words may refer to events, locations, English hashtags or retweet of English tweets with comments written in Arabic and they are part of the syntactic structure of Arabic tweets. So, they need to be tagged correctly. In this case, we use Stanford for English (Toutanova et al., 2003) to tag English words as a post-processing step.

5.2 Agreement-based Bootstrapping

Bootstrapping is used to create a labelled training data from large amounts of unlabelled data (Cucerzan and Yarowsky, 2002; Zavrel and Daelemans, 2000). There are different ways to select the labelled data from the taggers' outputs. We will follow (Clark et al., 2003) in using agreement-based training method. We will use the augmented versions of AMIRA, MADA and Stanford taggers to tag a large amount of Arabic tweets and add the tokens which they are agreed on to the training data. The taggers use different tagsets. Therefore, we will map these tagsets to a unified tagset consisting of main POS tags. Finally, we will retrain Stanford tagger on the selected labelled data.

Results for Pre- and Post-processing

In our experiments, the taggers were adapted to handle Twitter phenomena. The experiments were run using three off-the-shelf taggers trained on PATB and our augmented approach to address Arabic tweets noisiness as described in Section 5. Table 5 shows the overall performance of the augmented versions of the taggers compared with their baseline performance in Table 1. By combining normalisation and external knowledge,

[2]http://users.dsic.upv.es/grupos/nle/?file=kop4.php

we are able to reduce unknown tokens in each category which boosts the taggers' performance. The overall performance of the three taggers increases by absolute twelve percent accuracy for AMIRA, by absolute thirteen percent for MADA and by absolute sixteen percent for Stanford. This improvement in accuracy will reduce the propagation of POS tagging errors to downstream applications on Arabic tweets such as information extraction.

Tagger	Tweets	Processed Tweets
AMIRA	60.2%	**72.6%**
MADA	65.8%	**79.0%**
Stanford	49.0%	**65.2%**

Table 5: Impact of applying pre- and post-processing on POS tagging accuracy

6 Conclusion and Future Work

We have examined the consequences of applying MSA-trained POS tagging to Arabic tweets. The combination of normalisation and external knowledge was applied to text as pre- and post-processing steps. These steps go some of the way towards improving the taggers' accuracy over the MSA baseline. Our next step is to use bootstrapping and taggers agreement on unlabelled data to create a sufficient amount of labelled training tweets in order to retrain Stanford on it since it is only the retrainable tagger.

Acknowledgments

The authors would like to thank the anonymous reviewers for their encouraging feedback and insights. Fahad would also like to thank King Saud University for their financial support. Allan Ramsay's contribution to this work was partially supported by Qatar National Research Foundation (grant NPRP-7-1334-6 -039).

References

Muhammad Abdul-Mageed, Sandra Kübler, and Mona Diab. 2012. SAMAR: A system for subjectivity and sentiment analysis of Arabic social media. In *Proceedings of WASSA*.

Rania Al-Sabbagh and Roxana Girju. 2012. A supervised POS tagger for written Arabic social networking corpora. In *Proceedings of KONVENS*.

Fahad Albogamy and Allan Ramsay. 2015. Towards POS tagging for Arabic tweets. In *Proceedings of ACL Workshop on Noisy User-generated Text*.

Luciano Barbosa and Junlan Feng. 2010. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of ACL*.

Eric Brill. 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*.

Stephen Clark, James R. Curran, and Miles Osborne. 2003. Bootstrapping POS taggers using unlabelled data. In *Proceedings of NAACL*. ACL.

Silviu Cucerzan and David Yarowsky. 2002. Bootstrapping a multilingual part-of-speech tagger in one person-day. In *Proceedings of NLL*. ACL.

Leon Derczynski, Alan Ritter, Sam Clark, and Kalina Bontcheva. 2013. Twitter part-of-speech tagging for all: Overcoming sparse and noisy data. In *Proceedings of RANLP*.

Mona Diab. 2009. Second generation AMIRA tools for Arabic processing: Fast and robust tokenization, POS tagging, and base phrase chunking. In *2nd International Conference on Arabic Language Resources and Tools*.

Nawal El-Fishawy, Alaa Hamouda, Gamal M. Attiya, and Mohammed Atef. 2014. Arabic summarization in twitter social network. *Ain Shams Engineering Journal*.

Hady Elsahar and Samhaa R. El-Beltagy. 2014. A fully automated approach for Arabic slang lexicon extraction from microblogs. In *Proceedings of CICLing*.

Jennifer Foster, Özlem Çetinoglu, Joachim Wagner, Joseph Le Roux, Stephen Hogan, Joakim Nivre, Deirdre Hogan, Josef Van Genabith, et al. 2011. # hardtoparse: POS tagging and parsing the twitterverse. In *Proceedings of AAAI*.

Phani Gadde, L. V. Subramaniam, and Tanveer A. Faruquie. 2011. Adapting a WSJ trained part-of-speech tagger to noisy text: Preliminary results. In *Proceedings of MOCR*.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for Twitter: Annotation, features, and experiments. In *Proceedings of ACL: HLT*.

Nizar Habash, Owen Rambow, and Ryan Roth. 2009. Mada+ tokan: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a #twitter. In *Proceedings of ACL: HLT*.

Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. 2007. Why we Twitter: Understanding microblogging usage and communities. In *Proceedings of WebKDD*. ACM.

Ahmed Mourad and Kareem Darwish. 2013. Subjectivity and sentiment analysis of modern standard Arabic and Arabic microblogs. In *Proceedings of WASSA*. ACL.

Suleiman H Mustafa. 2012. Word stemming for Arabic information retrieval: The case for simple light stemming.

Eshrag Refaee and Verena Rieser. 2014. An Arabic Twitter corpus for subjectivity and sentiment analysis. In *Proceedings of LREC*.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of EMNLP*.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of NAACL*.

Jakub Zavrel and Walter Daelemans. 2000. Bootstrapping a tagged corpus through combination of existing heterogeneous taggers. In *Proceedings of LREC*.