Part-of-speech Tagging of Code-Mixed Social Media Text

Souvick Ghosh
Jadavpur University
Kolkata, WB 700032
India

Satanu Ghosh
MAKAUT
Kolkata, WB 700064
India

Dipankar Das
Jadavpur University
Kolkata, WB 700032
India

Abstract

A common step in the processing of any text is the part-of-speech tagging of the input text. In this paper, we present an approach to tackle code-mixed text from three different languages Bengali, Hindi, and Tamil - apart from English. Our system uses Conditional Random Field, a sequence learning method, which is useful to capture patterns of sequences containing code switching to tag each word with accurate part-of-speech information. We have used various pre-processing and post-processing modules to improve the performance of our system. The results were satisfactory, with a highest of 75.22% accuracy in Bengali-English mixed data. The methodology that we employed in the task can be used for any resource poor language. We adapted standard learning approaches that work well with scarce data. We have also ensured that the system is portable to different platforms and languages and can be deployed for real-time analysis.

1 Introduction

Part-of-Speech (POS) tagging a syntactic analysis usually done after language identification - is one of the key tasks in any language processing applications. It is the process of assigning the appropriate part of speech or lexical category to each word in a sentence. Apart from assigning grammatical categories to words in a text, POS tagging also helps in automatic analysis of any text.

To develop an accurate tagger, it is essential to develop various rules based on the language or large annotated corpus which could be used for discovering the rules and training the model. Accurate annotation of a corpus requires the expertise of linguists which is expensive and time consuming. Also it is not extendable from one language to another. Use of automatic machine learning approaches is inexpensive, fairly accurate and can be extended between languages.

The increasing popularity of social media platforms blogs, micro-posts (e.g. Twitter[1]) and chats (Facebook [2]) - has ensured availability of large amount of code-mixed data. But, texts obtained from various online platforms differ from traditional writings. These texts are predominantly unstructured. Also, many variations can be observed in terms of writing style and vocabulary. Such texts are mostly informal and have multiple languages in a single sentence, or even in a single word. This code-mixed nature of text, coupled with the fact that they are written using Roman script (instead of native script), makes it extremely challenging for linguists and data analysts to process such data. This has given a new dimension to the traditional problems of language identification and POS tagging.

In this paper, we address the problem of part-of-speech tagging in mixed social media data. India is a land of many languages with Hindi and English recognized as the more popular ones. From the Indian perspective, it is generally observed that one of the languages used in social media conversations are either English or Hindi. In this work, all the three mixed scripts contain English as one of the lan-

[1] twitter.com
[2] www.facebook.com

Proceedings of the Second Workshop on Computational Approaches to Code Switching, pages 90–97,
Austin, TX, November 1, 2016.

guages. The Indian languages present are Bengali, Hindi and Tamil.

To tag the words with their corresponding part-of-speech tags, we have used Stanford part-of-speech tagger as our baseline and developed the final system using Conditional Random Field (CRF). We have obtained results for three language pairs, namely Hindi-English (Hi-En), Bengali-English (Bn-En) and Tamil-English (Ta-En). In this paper, we concentrate on building our POS tagger system with minimal external resources. Both our models do not use any language resource in addition to the dataset. While the Stanford POS Tagger uses no additional resource, the CRF model uses only a list of smileys.

The rest of the paper is organized as follows. We present an account of the previous works done in the part-of-speech tagging in Section 2. In Section 3, we discuss the dataset. The system has been described in Section 4. The results and observations have been presented in Section 5 and the conclusion in Section 6.

2 Related Work

Part-of-Speech tagging has been a centre of many researches for the past few decades. Since it started in the middle sixties and early seventies (Greene and Rubin, 1971), a lot of new concepts have been introduced to improve the efficiency of the tagger and to construct the POS taggers for several languages.

Rule based POS tagger was introduced in the nineties (Karlsson et al., 1995) and gave better accuracy than its predecessors. One of the most successful rule based English tagger (Samuelsson and Voutilainen, 1997) had a recall of 99.5% with a precision of around 97%. The rule based taggers consists of complex but accurate constraints which makes them very efficient for disambiguation. Statistical model based tagger (DeRose, 1988; Cutting et al., 1992; Dermatas and Kokkinakis, 1995; Meteer et al., 1991; Merialdo, 1994) are widely used because of the simplicity and the independence of the language models. Most commonly used statistical models are bi-gram, tri-gram and Hidden Markov Model (HMM). The only problem with statistical models is that these kinds of taggers require a large annotated corpus. Machine learning algorithms are statistical in nature but the models are more complicated than simple n-gram. Models for acquiring disambiguation rules and transformation rules from the dataset were constructed in late 80's and early 90's (Hindle, 1989; Brill, 1992; Brill, 1995a; Brill, 1995b). Neural networks have also been used for POS tagging (Nakamura et al., 1990; Schütze, 1993; Ma and Isahara, 1998; Eineborg and Gambäck, 1994). POS taggers were also developed using Support Vector Machine (SVM) (Nakagawa et al., 2001). These taggers were more simple and efficient than the previous taggers. The successor of this tagger was developed by Giménez and Marquez (2004) and the approach they used for POS tagging was considerably faster than its predecessor. A more recent development was the use of Conditional Random Field (CRF) for POS tagging (Sha and Pereira, 2003; Lafferty et al., 2001; Shrivastav et al., 2006). These taggers are better for disambiguation as they find global maximum likelihood estimation.

2.1 POS Taggers for Indian Languages

Recently, a large number of researchers are trying to expand the scope of automatic POS taggers so that they can work on complex non European languages. India is a country with rich linguistics so POS taggers for Indian languages are one of the most explored topics. The first effort was to develop a Hindi POS tagger dated back in the nineties (Bharati et al., 1995). This tagger was based on a morphological analyzer. The analyzer would provide the root word with its morphological features and generalized POS category. Shrivastav et al. (2006) slightly modified this approach by using a decision tree based classifier and achieved an accuracy of 93.45%. Instead of using a full morphological analyzer Shrivastava and Bhattacharyya (2008) used a stemmer to generate suffixes which was in turn used to generate POS tags. Conditional Random Field was also used along with morphological analyzer in a couple of works (Agarwal and Mani, 2006; PVS and Karthik, 2007).

One of the earliest works on Bengali POS tagger was conducted by Seddiqui et al. (2003) and Chowdhury et al. (2004). (Chowdhury et al., 2004) implemented a rule based tagger which hand written rules formulated by expert linguists. In more recent work, Hasan et al. (2007) developed a supervised POS tagger. This method was less effective due to lack of tagged training corpus. In later years, we

have seen many works on Bengali POS tagger. One of the most successful taggers was developed by using HMM and Maximum Entropy models (Dandapat and Sarkar, 2006; Dandapat, 2007). They also used a morphological analyzer to compensate for the lack of annotated training corpus. These two models were used to implement a supervised tagger and a semi-supervised tagger. The accuracy achieved was around 88% for both models. Ekbal et al. (2007) carried out further research on the tagger. They annotated a news corpus and created two taggers - one SVM based tagger and another CRF based tagger - which reported an accuracy of 86.84% and 90.3% respectively.

In Tamil, Selvam and Natarajan (2009) proposed a rule based morphological analyzer to annotate the corpora and used it to train the POS tagger. They used the Tamil version of Bible for the tagged corpus and achieved an accuracy of 85.56%. Dhanalakshmi et al. (2009) developed a SVM based tagger using linear programming and a new tagset for Tamil with 32 tags. They used this tagset for building a training corpus and reported an accuracy of 95.63%. Another SVM based POS tagger (Dhanalakshmi et al., 2008) was proposed by them in a different work. They extracted linguistic information using machine learning techniques which was then used to train the tagger. This tagger achieved an accuracy of 95.64%.

Even after decades of research on monolingual POS taggers for Indian languages(mostly Hindi), there are just a few taggers with accuracy over 90%. A new challenge has developed over the past few years in the form of code mixed social media text. This field of research is at a nascent stage. The basic challenges and complexities of social media text are spelling variations and word sense disambiguation. As traditional POS taggers were not efficient for social media text, new taggers targeting social media text were constructed. However, these taggers are mostly monolingual and not suitable for code-mixed text. The first was developed by Gimpel et al. (2011) for tagging English tweets. They developed a new POS tagset and tagged 1827 tweets for training corpus for a CRF tagger with arbitrary local features in log-linear model adaptation. Owoputi et al. (2013) improved the original Twitter POS tagger as they introduced lexical and unsupervised word clustering features. This increased the accuracy from 90% to 93%.

One of the first POS taggers for code-mixed text was developed by Solorio and Liu (2008). They constructed a POS tagger of English-Spanish text by using existing monolingual POS taggers for both the languages. They combined the POS tag information using heuristic procedures and achieved the maximum accuracy of 93.4%. However, this work was not on social media text and hence the difficulties were considerably less. Gella et al. (2013) developed a system to identify word level language and then chunk the individual languages and produce POS tags or every individual chunk. They used a CRF based Hindi POS tagger for Hindi and Twitter POS tagger for English and achieved maximum accuracy of 79%. Vyas et al. (2014) developed a English-Hindi POS tagger for code mixed social media text.

3 Dataset

A recent shared task was conducted by Twelfth International Conference on Natural Language Processing (ICON-2015)[3] , for part-of-speech tagging of transliterated social media text. Organizers released the code mixed train and test set for English-Hindi, English-Bengali and English-Tamil language pairs.

In Table 1, we provide a summary of the dataset in terms of the utterances. The number of utterances have been recorded for both the training and test data. In Table 2, we present a statistics of the number of sentences for each pair of languages in training as well as test data.

Language	**Sentences (Training)**	**Sentences (Test)**
Bengali-English	2837	1459
Hindi-English	729	377
Tamil-English	639	279

Table 2: Summary of Dataset (Sentences).

4 System Description

We have followed a supervised approach in this work. We have extracted various features that are pertinent to this task. The various steps involved in POS tagging are listed as follows:

[3] http://ltrc.iiit.ac.in/icon2015/contests.php

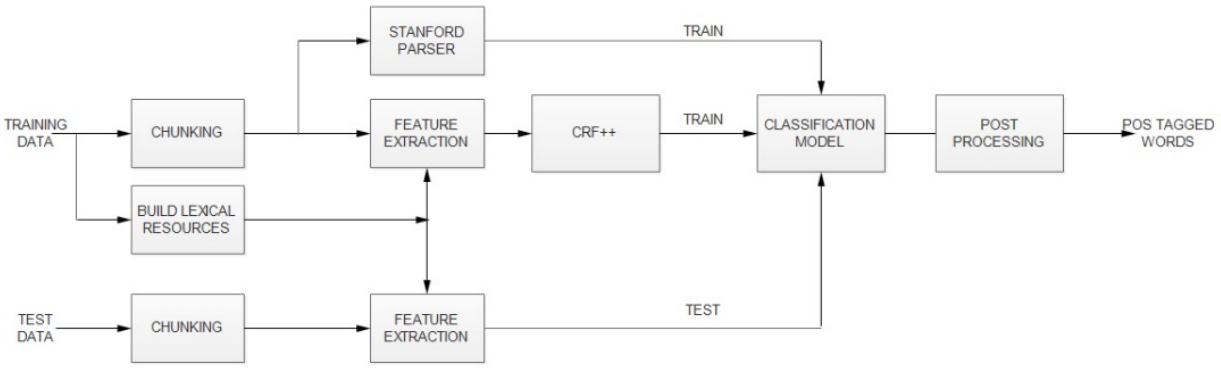

Figure 1: Overview of the System Architecture.

Language Tags	Utterances (Training)	Utterances (Test)
Hindi-English		
English (EN)	6178	8553
Hindi (HI)	5546	411
Others (O)	4231	2248
Total	15955	11212
Bengali-English		
English (EN)	9973	5459
Bengali (BN)	8330	4671
Others (O)	6335	3431
Total	24638	13561
Tamil-English		
English (EN)	1969	819
Tamil (TA)	1716	1155
Others (O)	630	281
Total	4315	2255

Table 1: Summary of Dataset (Utterances).

4.1 Chunking

Each of the three given corpora (Hindi-English, Bengali-English and Tamil-English) contains English as one of the dominant languages. The other dominant language is Bengali, Hindi and Tamil in each of the three texts. The various language tags used in the training data are en (English), hi (Hindi), bn (Bengali), ta (Tamil), ne (Named entities), acro (Acronyms), univ (Universal) and undef (Undefined). For each input file, we have performed chunking on the raw text to segment the words belonging to different language tag. We haves used the language ids to perform chunking. For each of the language tags, we have created a wordlist belonging to that particular language tag. We also maintain a table containing the file id, word id and position of every word. This table is useful for obtaining the output files from the chunked words.

4.2 Lexicons for Dominant Languages

English, Bengali, Hindi and Tamil were identified as the dominant languages. For each of these four languages, we have created a list of words which belong to any particular POS tag. These lists were constructed from the respective training files. We maintain lists for nouns, verbs and other parts-of-speech for each language. These lists are essential for extracting feature for training our CRF model.

4.3 POS Tagging

We have used two different approaches for POS Tagging of the test data. Both the models use training data for learning and model construction.

4.3.1 POS Tagging Using Stanford POS Tagger

For our baseline, we trained our system using Stanford POS Tagger (Toutanova et al., 2003). Using the training data, we trained the Stanford POS Tagger initially. The architecture (arch property of the tagger) that we used for training was: words(-1,1), unicodeshapes(-1,1), order(2), suffix(4). Four individual models were generated for English, Bengali, Hindi and Tamil. The test data was tagged using these generated models.

4.3.2 POS Tagging Using CRF++

In this work, Conditional Random Field (CRF) has been used to build the framework for word-level language identification classifier. We have used

CRF++ toolkit [4] which is a simple, customizable, and open source implementation of CRF.

The following features were used to train the CRF model:

- Length Of The Current Word

  The length of the current word has been used as one of the features. It is often noted that words belonging to a specific language and part-of-speech are often longer than others (Singh et al., 2008). We have used this feature to exploit word length in determining the part-of-speech of the word.

- Current Word

  For example, if the sentence is *I have been told of the place*, then each word is analyzed at a time. If the word currently being examined for part-of-speech tagging *is been*, then the word *been* is considered as one of the features.

- Previous Two Words

  For example, if the sentence is *I have been told of the place* and current word is *been*, then the previous two words are *I* and *have*.

- Next Two Words

  Using the previous example, if the sentence is *I have been told of the place* and the current word is *been*, then the next two words are *told* and *of*.

- Suffix

  This feature considers of the suffix of every word. If length of a word is more than 3 then suffix of length 3 and 2 are taken. e.g.: *een* and *en* are the suffixes for *been*.

- Prefix

  This feature considers of the prefix of every word. If length of a word more than 3 then prefix of length 3 and 2 are taken. e.g.: *bee* and *be* are the suffixes for *been*.

- If Word Contains Any Symbol

  This feature is boolean in nature and represents if the current word contains any symbol. Presence of symbol in a word gives a possible hint about the part-of-speech of the word.

- If Word Contains Any Digit

  Similar to the previous feature, this boolean feature represents if the current word contains any digit. Presence of digit in a word gives a possible hint about the part-of-speech of the word. e.g.: *kheye6ilam*, *ki6u*, *ka6e*, *6ghanta*

- Is Noun

  This feature represents if the current word is a noun. During the training phase, we build up a list of nouns for every language. This list is used during test phase to evaluate this feature. e.g.: *match*, *love*, *khushi*, *kaam*, *meye*

- Is Adjective

  This feature represents if the current word is an adjective. During the training phase, we build up a list of adjectives for every language. This list is used during test phase to evaluate this feature. e.g.: *ekta*, *beshi*, *good*, *nice*

- Is Verb

  This feature represents if the current word is a verb. During the training phase, we build up a list of verbs for every language. This list is used during test phase to evaluate this feature. e.g.: *hoy*, *lage*, *be*, *will*

- Is Pronoun

  This feature represents if the current word is a pronoun. During the training phase, we build up a list of pronouns for every language. This list is used during test phase to evaluate this feature. e.g.: *tomar*, *tumi*, *you*, *I*

- Is Conjunction

  This feature represents if the current word is a conjunction. During the training phase, we build up a list of conjunctions for every language. This list is used during test phase to evaluate this feature. e.g.: *kintu*, *and*, *to*, *but*

[4] https://taku910.github.io/crfpp/\#download

- Is Adverb

  This feature represents if the current word is an adverb. During the training phase, we build up a list of adverbs for every language. This list is used during test phase to evaluate this feature. e.g.: *ekhon*, *takhon*, *just*, *very*

- Is Determiner

  This feature represents if the current word is a determiner. During the training phase, we build up a list of determiners for every language. This list is used during test phase to evaluate this feature. e.g.: *the*, *this*, *a*

- Is Dollar

  This feature represents if the word represent any numerical measure. e.g.: *1st*, *26th*, *one*, *two*

- Is Q

  This feature represents if the word represent any quantitative measure. e.g.: *enuf*, *more*, *many*, *khub*

- Is U

  This feature represents if the word is website link e.g.: *pdf2fb.net*

- Is X

  This feature represents if the word is a non-classified token or if it has no meaning. e.g.: *geetamroadpi*

During the training phase, we train the CRF model using all the above features. Four language models are built, corresponding to the four dominant languages English, Bengali, Hindi and Tamil. In the test phase, we use the generated models to tag the words with their appropriate part-of-speech tags.

4.3.3 Post-processing

All the words belonging to the four dominant languages were tagged by the CRF model. The acronyms, named entities and the universal words were tagged by consulting the lists built during training. All the words which could not be tagged by our model were subjected to a post-processing module. For every language tag (acro, univ, ne), we found out the most frequent part-of-speech tag. Also, we used some logical reasoning to tag the words which were not tagged by our tagger models. For example, any untagged word which contains www, http or .com in it is allotted the U tag. Similarly, we use a smiley list to tag the smileys as E. Punctuations and hash-tags were tagged likewise. Finally, we combine all the words (which were chunked in initially) to obtain the output files.

5 Results and Observations

We evaluated the POS-tagging done by our baseline model (Stanford Parser) and the CRF model. The results are presented in Table 3.

	Accuracy in %	
Language Pair	**Baseline (Stanford Model)**	**CRF Model**
Bengali-English	60.05	75.22
Hindi-English	50.87	73.2
Tamil-English	61.02	64.83

Table 3: Accuracy of the system.

The results of Tamil-English are less than that of Bengali-English and Hindi-English. The primary reason for lower accuracy is the variation in tag used in gold standard files of Tamil-English.

6 Conclusion

In this paper, we have addressed the POS tagging of mixed script social media text. The texts contained two or three languages, with English being one of the three languages. The other languages were Hindi, Bengali and Tamil. We have trained Stanford POS Tagger to build a baseline model. Our final model used Conditional Random Field for part-of-speech tagging. Our results are encouraging and the performance deterioration of Tamil-English mixed text can be attributed to the mismatch of POS-tags.

Currently, there is a lack of quality training data. In the absence of sufficient training data, performance deteriorates using neural network based models or deep learning methods. In future, we would love to explore the effectiveness of Deep learning based features. Word2vec models can also be used to find out words which are semantically similar. We

would also like to use of ensemble learning by using various models and combining their results to arrive at the final result. A step in that direction would be to collect more mixed script data from social media and building gold standards using that data. Building an efficient normalization system and disambiguating between similar tags should also improve the accuracy of the system.

Acknowledgment

We are thankful to the organizers of the Twelfth International Conference on Natural Language Processing (ICON-2015)[5] for the dataset which we have used in our work.

References

Himanshu Agarwal and Anirudh Mani. 2006. Part of speech tagging and chunking with conditional random fields. In *the Proceedings of NWAI workshop*.

Akshar Bharati, Vineet Chaitanya, Rajeev Sangal, and KV Ramakrishnamacharyulu. 1995. *Natural language processing: a Paninian perspective*. Prentice-Hall of India New Delhi.

Eric Brill. 1992. A simple rule-based part of speech tagger. In *Proceedings of the workshop on Speech and Natural Language*, pages 112–116. Association for Computational Linguistics.

Eric Brill. 1995a. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational linguistics*, 21(4):543–565.

Eric Brill. 1995b. Unsupervised learning of disambiguation rules for part of speech tagging. In *Proceedings of the third workshop on very large corpora*, volume 30, pages 1–13. Somerset, New Jersey: Association for Computational Linguistics.

Md Shahnur Azad Chowdhury, NM Minhaz Uddin, Mohammad Imran, Mohammad Mahadi Hassan, and Md Emdadul Haque. 2004. Parts of speech tagging of bangla sentence. In *Proceeding of the 7th International Conference on Computer and Information Technology (ICCIT), Bangladesh*.

Doug Cutting, Julian Kupiec, Jan Pedersen, and Penelope Sibun. 1992. A practical part-of-speech tagger. In *Proceedings of the third conference on Applied natural language processing*, pages 133–140. Association for Computational Linguistics.

[5] ltrc.iiit.ac.in/icon2015/

Sandipan Dandapat and Sudeshna Sarkar. 2006. Part of speech tagging for bengali with hidden markov model. *Proceeding of the NLPAI Machine Learning Competition*.

Sandipan Dandapat. 2007. Part of specch tagging and chunking with maximum entropy model. In *Proceedings of the IJCAI Workshop on Shallow Parsing for South Asian Languages*, pages 29–32.

Evangelos Dermatas and George Kokkinakis. 1995. Automatic stochastic tagging of natural language texts. *Computational Linguistics*, 21(2):137–163.

Steven J DeRose. 1988. Grammatical category disambiguation by statistical optimization. *Computational linguistics*, 14(1):31–39.

V Dhanalakshmi, M Anandkumar, MS Vijaya, R Loganathan, KP Soman, and S Rajendran. 2008. Tamil part-of-speech tagger based on svmtool. In *Proceedings of the COLIPS International Conference on natural language processing (IALP), Chiang Mai, Thailand*.

V Dhanalakshmi, G Shivapratap, and Rajendran S Soman Kp. 2009. Tamil pos tagging using linear programming.

Martin Eineborg and Björn Gambäck. 1994. Tagging experiment using neural networks. *Eklund (ed.)*, pages 71–81.

Asif Ekbal, S Mondal, and Sivaji Bandyopadhyay. 2007. Pos tagging using hmm and rule-based chunking. *The Proceedings of SPSAL*, 8(1):25–28.

Spandana Gella, Jatin Sharma, and Kalika Bali. 2013. Query word labeling and back transliteration for indian languages: Shared task system description. *FIRE Working Notes*, 3.

Jesús Giménez and Lluis Marquez. 2004. Fast and accurate part-of-speech tagging: The svm approach revisited. *Recent Advances in Natural Language Processing III*, pages 153–162.

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 the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics.

Barbara B Greene and Gerald M Rubin. 1971. *Automatic grammatical tagging of English*. Department of Linguistics, Brown University.

Muhammad Fahim Hasan, Naushad UzZaman, and Mumit Khan. 2007. Comparison of unigram, bigram, hmm and brill's pos tagging approaches for some south asian languages.

Donald Hindle. 1989. Acquiring disambiguation rules from text. In *Proceedings of the 27th annual meeting on Association for Computational Linguistics*, pages 118–125. Association for Computational Linguistics.

Fred Karlsson, Atro Voutilainen, Juha Heikkilae, and Arto Anttila. 1995. *Constraint Grammar: a language-independent system for parsing unrestricted text*, volume 4. Walter de Gruyter.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1, pages 282–289.

Qing Ma and Hitoshi Isahara. 1998. A multi-neuro tagger using variable lengths of contexts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2*, pages 802–806. Association for Computational Linguistics.

Bernard Merialdo. 1994. Tagging english text with a probabilistic model. *Computational linguistics*, 20(2):155–171.

Marie Meteer, Richard Schwartz, and Ralph Weischedel. 1991. Studies in part of speech labelling. In *Proceedings of the workshop on Speech and Natural Language*, pages 331–336. Association for Computational Linguistics.

Tetsuji Nakagawa, Taku Kudo, and Yuji Matsumoto. 2001. Unknown word guessing and part-of-speech tagging using support vector machines. In *NLPRS*, pages 325–331. Citeseer.

Masami Nakamura, Katsuteru Maruyama, Takeshi Kawabata, and Kiyohiro Shikano. 1990. Neural network approach to word category prediction for english texts. In *Proceedings of the 13th conference on Computational linguistics-Volume 3*, pages 213–218. Association for Computational Linguistics.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. Association for Computational Linguistics.

Avinesh PVS and G Karthik. 2007. Part-of-speech tagging and chunking using conditional random fields and transformation based learning. *Shallow Parsing for South Asian Languages*, 21.

Christer Samuelsson and Atro Voutilainen. 1997. Comparing a linguistic and a stochastic tagger. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 246–253. Association for Computational Linguistics.

Hinrich Schütze. 1993. Part-of-speech induction from scratch. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*, pages 251–258. Association for Computational Linguistics.

Md Hanif Seddiqui, AKMS Rana, Abdullah Al Mahmud, and Taufique Sayeed. 2003. Parts of speech tagging using morphological analysis in bangla. In *Proceeding of the 6th International Conference on Computer and Information Technology (ICCIT), Bangladesh*.

M Selvam and AM Natarajan. 2009. Improvement of rule based morphological analysis and pos tagging in tamil language via projection and induction techniques. *International journal of computers*, 3(4):357–367.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 134–141. Association for Computational Linguistics.

M Shrivastav, R Melz, Smriti Singh, K Gupta, and P Bhattacharyya. 2006. Conditional random field based pos tagger for hindi. *Proceedings of the MSPIL*, pages 63–68.

Manish Shrivastava and Pushpak Bhattacharyya. 2008. Hindi pos tagger using naive stemming: Harnessing morphological information without extensive linguistic knowledge. In *International Conference on NLP (ICON08), Pune, India*.

Thoudam D Singh, Asif Ekbal, and Sivaji Bandyopadhyay. 2008. Manipuri pos tagging using crf and svm: A language independent approach. In *proceeding of 6th International conference on Natural Language Processing (ICON-2008)*, pages 240–245.

Thamar Solorio and Yang Liu. 2008. Learning to predict code-switching points. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 973–981. Association for Computational Linguistics.

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 the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics.

Yogarshi Vyas, Spandana Gella, Jatin Sharma, Kalika Bali, and Monojit Choudhury. 2014. Pos tagging of english-hindi code-mixed social media content. In *EMNLP*, volume 14, pages 974–979.