Part of Speech tagging for South African English

Alec Badenhorst
bdnale004@myuct.ac.za
University of Cape Town
Rondebosch, South Africa

ABSTRACT

Part-of-speech tagging is generally considered solved by academics. These studies focused on American and British English specifically. South African English is an a unique position with 10 other languages from which loan words could be taken. Named entity recogntion attempts to tag entities which are considered named, such as times, dates, places, and proper nouns. Although generally behind regular part-of-speech tagging, named entity recogntion still managed to reach high accuracies of 90% for American English, but suffers greatly when put up against South African English. This paper seeks to create an overview of part-of-speech tagging, and named entity recognition with respect to South African English, and determine whether further research is required for either part-of-speech tagging, or named entity recognition for South African English

KEYWORDS

Part-of-speech, Named Entity Recognition

1 INTRODUCTION

South Africa, like many other countries, has developed its own brand of English, taking loan words from its 10 other official languages. Although part-of-speech tagging has been done (quite successfully) to other English variants, notably American and British English, there is very little work involving South African part-of-speech tagging for South African English as a whole.

One work that involved part-of-speech tagging (for the purpose of text-to-speech synthesis for South African languages in general) was done by Schlünz, Dlamini, and Kruger [28]. They used part-of-speech tagging for the purposes of creating more accurate speech synthesis, as the pronunciation of certain words may rely on its part-of-speech, such as "lead" as a noun versus "lead" as a verb. According to their results, the part-of-speech tagger[1] they used reached an accuracy of 96.58%, similar to that of other English variants.

Louis, De Waal, and Venter [17] applied named entity recognition to South African texts, using various tricks to determine if something was a named entity or not. These including checking for capitalization, a possessive "'s", and the use of gazetteers (a list of common names). They excluded some common South African names such as "Precious" or "Gift" as it would confuse their tagger. They achieved F scores (a measure of accuracy) between 0.42 and 0.67 without and with a gazetteer respectively.

Eiselen [6] used named entity recognition focused on the government domain. They were not specifically focused on South African English, but South African languages as a whole. Their study focused in on the other 10 official languages of the country, and achieved an F score of roughly 0.75 for most languages with the exceptions of SiSwati and isiZulu.

1.1 Natural Language Processing, Part-of-speech tagging, and Named entity recognition

Natural language processing (NLP) is the use of computers for analysing large amounts of human (natural) language. NLP applications include speech recognition, understanding natural language, natural language generation, and machine translation. Computer analysis of natural language is a field of active research and includes several subfields of interest. There are several subfields in natural language processing, such as syntactical analysis, semantic analysis, part-of-speech tagging, parsing, and word segmentation, etc.

The traditional method of natural language processing happens in several steps that pass their end product to the next step. The segmented nature of NLP allows researchers to hone in on a particular field, and do extensive research in the field without needing to worry about the others. Some researchers, such as Collobert et al. [5], believe that a different approach may yield better results. They approached the field as a whole, and trained an AI framework from the ground up, as opposed to focussing on one subfield such as part-of-speech tagging.

One of the early steps in NLP is part-of-speech tagging (POS tagging), which is concerned with indicating the syntactic role of words. There are several methods for doing this such as using Hidden Markov Models [15] and maximum entropy [11], [27]. POS faces difficulties when it comes to words that may have multiple valid parts of speech, but is a mature field that produce near-human levels of accuracy when tagging [21].

[1] 10.

One challenge in part-of-speech tagging is that of named entity recognition, or NER. Named entity recognition is the identification of proper nouns, such as the names of people, places, organizations, and numeric expressions, such as time, money, percentages, etc. NER is an application of POS tagging that faces some design challenges. NER systems tend to use POS tags to help them along, but one is left with 4 design choices accoring to Ratinov and Roth [26]. NER usually follows similar techniques as POS tagging, where the problem becomes a sequential prediction one. Some common techniques used in NER are Hidden Markov Models [25], conditional random fields [16], and perceptron algorithms [4].

2 ONLINE DICTIONARIES AND THEIR COMPUTATIONAL USE

An online dictionary is something most people have interacted with, but some exist with the additional (and probably primary) purpose of being used for computation. These dictionaries include different types of annotations (such as the part-of-speech). These online dictionaries can be used for various purposed including training AI with test (pretagged) data, or potentially be used as a look-up to help named entity recognition.

WordNet by Princeton University[2] is an open online (also available as an offline download) lexicon of English. A South African version of this dictionary also exists under Rhodes University, known as DSAE[3].

3 PART-OF-SPEECH TAGGING

Part-of-speech (POS) tagging assigns a role to each word in a sentence. The part-of-speech, or the role of a word depends on context. For example, the word "*lead*" may either be a noun (a soft grey metal), or a verb, to show someone the way to somewhere. Humans tell the difference between parts of speech through context. The first challenge with POS tagging comes from identifying the different set of suitable parts of speech. Pustet [24] categorized the parts of speech into three primary parts, nouns, verbs, and adjectives. These categories may of course be expanded to include adverbs, auxiliary verbs, etc. in separate categories at the discretion of researchers, or the creators of tagging tools.

Part-of-speech tagging first relies on the ability to tell different words apart. In English, words are seperated by a space. In other languages, such as Chinese, Japanese, and Korean, this is not the case, and some extra steps would have to be taken to seperate the text into words, before tagging can be done.

[2] 31.
[3] 32.

The next problem we have to deal with is ambiguous words. Some English words could be categorized into multiple categories [9].

We also need to deal with unknown words, loan words, or neologisms. Unknown words may be named entities, i.e. the names of people, places, numbers, etc., words that did not appear in the training corpa, or words which are not handled by the rules laid out by the tagger [9]. Unknown words may also be loan words from other languages in the region (South African English taking from Afrikaans, for example), or neologisms.

Some part-of-speech tagging methods reach accuracies of up to 97% [15], [11], but humans can sometimes not even agree on what part-of-speech some words belong to. Marcus, Santorini, and Marcinkiewicz [19] found that there were disagreements on about 7.2% of words tagged. A solution to this is to allow a word to have multiple tags. Another solution to ambiguity may be to expand the view of the tagger to look at the context in which the word is being used.

Güngör [9] uses the example sentence

We can can the can.

Here a human can instantly recognize the different roles that *can* undertakes, but for a machine, the differentiation may not be so clear, since the category of a word may change based on affixes, or neighbouring words [12].

Other errors by taggers may be made due to typographical errors in the text they parse, or due to insufficient training. Elworthy [7] proposed a technique for detecting errors made by Hidden Markov Model taggers. They compared the observable values of the tagging process with a threshold. By trading some efficiency (the proportion of tagged words), the accuracy may be improved.

3.1 Tagging methods

3.1.1 Rule-based tagging. The earliest POS tagging methods used a rule-based approach. This method involves handcrafting a large set of rules in order to determine the part-of-speech. This was the method initially employed to tag the Brown Corpus [18], also known as the Brown University Standard Corpus, an American English corpus compiled in the 1960s.

Brill [2] proposed a simple rule-based tagger that had a 5.1% error rate when tested on 5% of the Brown Corpus, which included sections for every genre. They argue that a rule-based approach has several advantages over more sophisticated stochastic approaches, due to portability and extensibility.

3.1.2 Transformation-based tagging. Brill [3] later went on to define transformation-based tagging. Instead of defining rules manually, an AI framework learns a set of correction

rules from mistakes. Initially, the AI might tag words randomly, or it might choose the part-of-speech that a word has been tagged with before, or even tagging everything as a particular part-of-speech. After the initial tagging, it starts learning. The AI takes a set of predetermined rules, and applies them to data from the corpus. It then identifies the rules that reduces the error the most, and adds the rule to the set of learned rules. The process is then iterated over a new corpus with the previously learned rules as an initialization step, and the process is repeated until none of the remaining rules reduce the error.

3.1.3 Hidden Markov Models. Kupiec [15] describes a part-of-speech tagging system that uses a Hidden Markov Model AI trained on a corpus of untagged text. They modeled the set of n states representing the part-of-speech categories as part of a finite state machine. They then defined C_r and C_{r-1} as random variables denoting the category of the r^{th} word and the $r - 1^{th}$ word. The transition probability then represents the probability of a word of any particular part-of-speech following another. This approach allows us the context of the previous word (and may be expanded to add one or two more words), but the context of the words are largely unknown.

Despite the limitation of not knowing the context of a word, using this approach, Kupiec [15] achieved an accuracy of 96%.

3.1.4 Maximum Entropy Models. Ratnaparkhi [27] used a maximum entropy model for part-of-speech tagging, and more recently Jianchao [11] tackled an optimization (of time) problem, improving on the work of Petrov, Das, and McDonald [22]. This method allows better flexibility than the Hidden Markov Models, as we can have a larger view on the context of a word.

The maximum entropy model allows the use of a set of rules, and the probability that any one of those rules are correct. Yager [34] explains that the negation of a probability distributions works as follows:

Take a rule such as "If V is *tall*, then U is *b*, and if V is *not tall*, then U is *d*". If the idea of *tall* is presented as fuzzy, then the process of obtaining *not tall* is known, taking the inverse. Now, representing *tall* as a probability distribution, then determining *not tall* becomes one of determining the negation of said probability distribution.

Thus, if the probability of a particular word being "not a noun" is low, then the probability of a word being a "noun", is high.

Ratnaparkhi [27] achieved an accuracy of 96.6% using this approach, and Jianchao [11] achieved accuracies close to 95%, gaining not only higher accuracies than Petrov, Das, and McDonald [22], but also cutting the time it took to tag the data set in half.

3.2 South African English

Part-of-speech tagging is one of the most researched fields of NLP, and is considered 'solved' by some. However, little research appears with respect to South African English in particular. Although general tagging methods using existing taggers may be reasonably accurate, many South African English-specific words exist, including usage differences.

South African English has taken words from many Bantu languages, as well as from Afrikaans. Some such words include "lekker", and "ubuntu". Different usages also exist for some words, such as "robot" being taken to mean "traffic light".

General parsers may achieve acceptable accuracy, but will almost certainly not reach the high 95% accuracies presented by previous research if applied directly to South African English.

Annotated dictionaries dedicated to South African English do exist, such as the Rhodes University Dictionary of South African English (DSAE)[4].

4 NAMED ENTITY RECOGNITION

The idea of named entities has its origin in the Named Entity Recognition and Classication tasks, which forms a part of information retrieval systems [1]. This comes from the Message Understanding Conference (MUC), a conference which started with one task, identify a class of events in a piece of text [8]. The MUC was initiated by the Defense Advanced Research Projects Agency to extract information. The idea behind named entity recognition is to enable machines to recognize the *entities* which humans talk about. These entities are things like people, places, dates, organizations, etc. Marrero et al. [20] took aggregate definitions of named entities, and categorized them using four criteria: *grammatical category*, *rigid designation*, *unique identification*, and *domain of application*.

Marrero et al. [20]'s grammatical category refers to proper nouns, or common names acting as proper nouns.

Rigid designatiors are names that do not change. Kripke [13] uses the example of *Richard Nixon* as opposed to *President of the United States of America*. Because the president of the USA changes every couple of years, it cannot be classified as a rigid entity. In contrast, *the automotive company created by Henry Ford in 1903* is referred to as *Ford*. This is an example of a rigid designator, as the entity being referred to does not change.

Unique identifiers require previous knowledge of the entity being referred to, but are unique to that entity. For example, a model number for an aircraft, or specific type of animal.

[4] 32.

The purpose and domain of application determine what named entities one should prioritize for tagging. For example, tagging a chemistry textbook, one would be looking for names of chemicals specifically, and would include them in the gazetteer. Given the origin of the MUCs, military application was a clear goal, and thus, certain events would take precedence over other, such as looking out for an agent, time, cause, and where an event took place.

Marrero et al. [20] provided some examples for more clarity.

One such example was the named entity "Water". It had rigid designations of "sparkling" or "still", with the example domain being "chemistry". In this example, the named entity refers specifically to "sparkling" or "still" as these are names given to types of water. Another example was the named entity "Airbus A310", a proper noun with the rigid designation being that of a specific plane, and the example domain being "Army/ Tech. Watch".

Because NER is a part of POS tagging, similar methods are employed to recognize these entities. Krupka and Hausman [14] describes a rule-based system. There are also models based on machine learning and statistical modelling, such as Rabiner [25]'s Hidden Markov Models, or Lafferty, McCallum, and Pereira [16]'s conditional random fields model.

Ratinov and Roth [26] suggest that external resources, such as Wikipedia, be used in tandem with regular tagging methods to increase accuracy. They suggest that one could use an external resource to look up the names of unknown entities to determine whether they are a *Named Entity*.

4.1 Identification Methods

4.1.1 Hidden Markov Models. The Hidden Markov Models used here work in much the same way as the ones used for regular part-of-speech tagging. Collins [4] describes a version of HMM relying on Viterbi decoding and perceptron training.

Viterbi decoding uses the viterbi algorithm, developed by Andrew J. Viterbi [33], to decode a bitstream encoded by convolutional code (error-correcting code that uses parity symbols).

Collins [4] highlights errors with the parameter estimation method for maximum entropy models, and thus suggests variants of the perceptron training algorithm for tagging problems.

4.1.2 Conditional Random Fields. Conditional random fields (CRFs) are a part of statistical modelling which is commonly used for pattern recognition, and thus are useful in natural language processing.

Lafferty, McCallum, and Pereira [16] recognized problems with parameter estimation, and thus described the alternative method of using conditional random fields to determine these parameters.

HMMs use the previously tagged part-of-speech as a way to determine what the next part-of-speech might be, but this has its limitations. In contrast, CRFs use variables X, which may range over the set of natural language sentences, and Y, which ranges over possible part of speech tags. We then define a model $p(X|Y)$, where X and Y are jointly distributed, and the model represents paired observation and label sequences.

To put it in another way, certain parts of speech appear in certain types of sentences. Given this, we can apply this to NER. Because certain parts of speech appear in certain places in certain types of sentences, we can use this model to predict whether something is in fact a named entity.

5 PERFORMANCE

5.1 Part-of-speech tagging

Part-of-speech tagging has several techniques that one can make use of. Researchers have a preference for the Hidden Markov Model due to its extensibility. The biggest drawback of the Hidden Markov Model is the time it takes to train the model to acceptable accuracy levels.

An improvement on HMMs is the use of maximum entropy models, which allow for the tagger to see the sentence as a whole and understand the semantic context of a word, increasing its accuracy.

5.2 Named Entity Recogntion

Being similar to POS tagging, similar issues are faced in terms of performance in named entity recogntion. Ratinov and Roth [26] suggest using outside sources in tandem with a tagger to improve accuracies.

6 MODERN TOOLS

There are some free tools available for general NLP, and some dedicated to part-of-speech tagging.

WordNet [31] is Princeton University's lexical database for English. It is usable online, or downloadable, and is one of the most complete databases of the English language.

DSAE [32] is Rhodes University's database of South African English, including many of the loan words taken from other South African languages.

NLTK [23] is a natural language processing suite written in Python. It is an open source project that can be modified for specific use cases if necessary. NLTK includes many tools including annotaters for parts of speech and named entities.

HunPos [10] is another open source project which focuses on part-of-speech tagging. HunPos uses Hidden Markov Models to tag parts of speech.

Stanford university [29] has a log-linear part-of-speech tagger available under the GNU-GPL lisence.

There is also the TreeTagger used in several research papers, available from their website [30].

7 CONCLUSIONS

Part-of-speech tagging is often considered a solved problem, and in terms of English, taggers are robust enough to be able to deal with neologisms and loan words well enough if given the context of the word. The accuracy of existing taggers may be able to tag South African English as well as any other English variant [28], but further research in this area is needed to be be able to definitively determine whether a tagger would need to be specifically adapted for the use of South African English.

Named entity recognition, a subfield of POS tagging uses similar techniques to part-of-speech tagging, but may be futher improved by using outside resources in tandem with a tagger to determine whether something is a named entity. The accuracy of determining named entities suffers severely when being used for South African English and other South African languages, reaching accuracies of only 60%, and up to 75% for domain-specific recognition [17], [6]. This leaves something to be desired when compared to general NER tagging, reaching up to 90% [26].

In the case of both, it is possible to design a tagger which takes a tiered approach to tagging. First, run a rule-based tagger. These rules should account for the obvious cases, such as a named entity in posessive form; "The library's book". After the initial tagging has been done, move on to the next method. This may result in poorer time performance, but may be able to futher improve accuracy. A similar approach has been used before, but the general approach appears to be Hidden Markov Models.

Two solvable problems for South African English remain.

(1) Testing whether there is a difference, if any, in the tagging accuracy of South African English compared to American or British English.
(2) Improving named entity recognition with respect to South African English.

REFERENCES

[1] Oriol Borrega, Mariona Taulé, and M Antø'nia Martı. "What do we mean when we speak about Named Entities". In: *Proceedings of Corpus Linguistics*. 2007.

[2] Eric Brill. "A simple rule-based part of speech tagger". In: *Proceedings of the third conference on Applied natural language processing*. Association for Computational Linguistics. 1992, pp. 152–155.

[3] Eric Brill. "Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging". In: *Computational linguistics* 21.4 (1995), pp. 543–565.

[4] Michael Collins. "Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms". In: *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics. 2002, pp. 1–8.

[5] Ronan Collobert et al. "Natural language processing (almost) from scratch". In: *Journal of machine learning research* 12.8 (2011), pp. 2493–2537.

[6] Roald Eiselen. "Government Domain Named Entity Recognition for South African Languages". In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. 2016, pp. 3344–3348.

[7] David Elworthy. "Automatic Error Detection in Part of Speech Tagging". eng. In: *arXiv.org* (1994). URL: http://search.proquest.com/docview/2090484627/.

[8] Ralph Grishman and Beth M Sundheim. "Message understanding conference-6: A brief history". In: *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*. 1996.

[9] Tunga Güngör. *Part-of-Speech Tagging*. 2010.

[10] HunPos. *HunPos*. URL: https://code.google.com/archive/p/hunpos/ (visited on 05/10/2020).

[11] Tao Jianchao. "An English Part of Speech Tagging Method Based on Maximum Entropy". eng. In: *2015 International Conference on Intelligent Transportation, Big Data and Smart City*. IEEE, 2015, pp. 76–80. ISBN: 9781509004645.

[12] Dan Jurafsky and James H Martin. *Speech and language processing. Vol. 3*. Pearson London London, 2014.

[13] Saul A Kripke. "Naming and necessity". In: *Semantics of natural language*. Springer, 1972, pp. 253–355.

[14] George Krupka and Kevin Hausman. "IsoQuest Inc.: Description of the NetOwl™ Extractor System as Used for MUC-7". In: *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*. 1998.

[15] Julian Kupiec. "Robust part-of-speech tagging using a hidden Markov model". In: *Computer speech & language* 6.3 (1992), pp. 225–242.

[16] John Lafferty, Andrew McCallum, and Fernando CN Pereira. "Conditional random fields: Probabilistic models for segmenting and labeling sequence data". In: (2001).

[17] Anita Louis, Alta De Waal, and Cobus Venter. "Named entity recognition in a South African context". In: *Proceedings of the 2006 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*. 2006, pp. 170–179.

[18] Christopher D Manning, Christopher D Manning, and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT press, 1999.

[19] Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. "Building a large annotated corpus of English: The Penn Treebank". In: (1993).

[20] Mónica Marrero et al. "Named entity recognition: fallacies, challenges and opportunities". In: *Computer Standards & Interfaces* 35.5 (2013), pp. 482–489.

[21] Elaine Marsh and Dennis Perzanowski. "MUC-7 evaluation of IE technology: Overview of results". In: *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*. 1998.

[22] Slav Petrov, Dipanjan Das, and Ryan McDonald. "A universal part-of-speech tagset". In: *arXiv preprint arXiv:1104.2086* (2011).

[23] NLTK Project. *NLTK*. URL: https://www.nltk.org/ (visited on 05/10/2020).

[24] Regina Pustet. *Copulas: Universals in the Categorization of the Lexicon*. OUP Oxford, 2003.

[25] Lawrence R Rabiner. "A tutorial on hidden Markov models and selected applications in speech recognition". In: *Proceedings of the IEEE* 77.2 (1989), pp. 257–286.

[26] Lev Ratinov and Dan Roth. "Design challenges and misconceptions in named entity recognition". In: *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*. 2009, pp. 147–155.

[27] Adwait Ratnaparkhi. "A maximum entropy model for part-of-speech tagging". In: *Conference on Empirical Methods in Natural Language Processing*. 1996.

[28] Georg I Schlünz, Nkosikhona Dlamini, and Rynhardt P Kruger. "Part-of-speech tagging and chunking in text-to-speech synthesis for South African languages". In: (2016).

[29] Stanford. *Standford Log-linear Part-of-speech Tagger*. URL: https://nlp.stanford.edu/software/tagger.shtml (visited on 05/11/2020).

[30] TreeTagger. *TreTagger*. URL: https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/ (visited on 05/11/2020).

[31] Princeton University. *WordNet*. URL: https://wordnet.princeton.edu/ (visited on 04/27/2020).

[32] Rhodes University. *Dictionary Unit for South African English*. URL: https://www.ru.ac.za/dsae/ (visited on 04/25/2020).

[33] A. Viterbi. "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm". In: *IEEE Transactions on Information Theory* 13.2 (1967), pp. 260–269.

[34] Ronald R Yager. "On the maximum entropy negation of a probability distribution". In: 23.5 (2014), pp. 1899–1902.