UNIVERSITY OF CAPE TOWN
DEPARTMENT OF COMPUTER SCIENCE

CS/IT Honours
Final Paper 2020

Title: Part of Speech Tagger Efficacy on South African English

Author: Alec Badenhorst

Project Abbreviation: CASETEXT

Supervisor(s): Maria Keet

Category	*Min*	*Max*	Chosen
Requirement Analysis and Design	*0*	*20*	0
Theoretical Analysis	*0*	*25*	5
Experiment Design and Execution	*0*	*20*	20
System Development and Implementation	*0*	*20*	0
Results, Findings and Conclusions	*10*	*20*	20
Aim Formulation and Background Work	*10*	*15*	15
Quality of Paper Writing and Presentation	*10*		10
Quality of Deliverables	*10*		10
Overall General Project Evaluation (*this section allowed only with motivation letter from supervisor*)	*0*	*10*	0
Total marks		**80**	

Part of Speech tagger efficacy on South African English

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

ABSTRACT

Part-of-speech tagging is a fundemental part of natural language processing that forms the basis of a lot of research and processing in the field. It is a mature sub-field of natural language processing that has reached near-perfect levels of accuracies for English Kupiec [12] and Jianchao [8]. The research in the field was focussed on American and/or British English. We wish to conduct this research with a focus on South Africa and the English that has developed there through decades of assimilating loan words from other language found in the southern regions of Africa. To this end our researched focussed on tagging articles, blogs and novels produced in South Africa to test how domains affected the accuracies of existing taggers. We found that taggers trained on British or American English are able to tag South African Enlish with nearly the same accuracy as the English variants they were trained on.

KEYWORDS

Part-of-speech, natural language processing

1 INTRODUCTION

Part-of-speech tagging is a field with much research behind it. Kupiec [12] and Jianchao [8] found accuracies of up to 97%. Research in natural language processing that is centered around South African English is uncommon, and among those few reference the accuracy of part-of-speech tagging. Schlünz, Dlamini, and Kruger [19] cites an accuracy of 96.58% for English extracted from the government domain using the HunPos part-of-speech tagger. Even though they reference part-of-speech tagging accuracy, we have no comprehensive study on different types of taggers and their accuracy with regards to South African English and how domain affects these accuracies. Because South African English has different usages for words that already exist (ont top of loan words), the accuracies acheived by them could be due to the fact that taggers are more accurate with regards to government domain texts, as opposed to more informal communications, such as Twitter, Facebook, WhatsAPP, or blogs.

We wish to investigate how accurate existing taggers are in this paper, and take a look at how domain may affect accuracy. Regions containing multiple languages exhibits the use of loan words, adoptions and assimilations of terms and possible also grammar adjustments. South Africa is a country with a rich diversity of languages and thus South African English is a prime language to undergo such changes.

Using four different taggers, we wish to investigate how well they perform in tagging South African English.

We will investigate several questions: How accurate are modern part-of-speech taggers when tagging South African English when compared to other English variants?

South African English has taken many loan words from lnaguage in South Africa and it's surrounding countries. Some of these language differ greatly from English and are no germanic languages in nature. So our first hypothesis is as follows:

1. *Part-of-speech taggers will not perform as well on South African English as on other English variants.*

How does the domain of text affect part-of-speech taggers' accuracies?

The domain of the text greatly affects the register used in the text, as well as the language used. Legal (and other formal) documents use more standard language, while a sports article might use less formal language and more slang. Our hypothesis here is that

2. *taggers will performance measurably better when the register of the text is more formal.*

Does the type of tagger used affect the accuracy of tagging? By this we mean that if the tagger is a rule-based tagger as opposed to a stochastic tagger, would that affect the results? If the tagger uses some sort of look-ahead to gain some contextual knowledge of words, would the accuracy be improved?

3. *Context sensitive taggers will be more accurate in tagging words as opposed to taggers that do not use the surrounding*

context.

For our testing we settled on four part-of-speech taggers. They were selected due to either being developed taggers or being easy to access and use. The four taggers we are using are HunPos [7], Stanford [22] Part of Speech Tagger, NLTK (Averaged Perceptron tagger) Project [17], and TreeTagger [23].

2 RELATED WORK

Not much work in part-of-speech tagging centric to South African English has been done before, so a broader overview of part of speech tagging will be given after the discussion of South African natural language processing works.

2.1 South African Centric Work

Schlünz, Dlamini, and Kruger [19] used part-of-speech tagging for the purposes of text-to-speech synthesis, though their application was more broad, focussing on all of South Africa's official languages, rather than English specifically. Knowing the part-of-speech of a word will help with the pronunciation of the word, for example 'lead' can be the soft grey metal, or it could be the verb to show someone the way somewhere, either of which is pronounced in dramatically different ways. For the purposes of tagging English, they used the HunPos [7] tagger. They quote an accuracy score of 96.58%.

Louis, De Waal, and Venter [13] studied the accuracy of named entity recognition, la sub-field of part-of-speech tagging related to recognizing (largely) proper nouns. They used various tricks to identify potential named entities, such as checking for capitalization, checking for posessive endings and checking against a gazetteers. They exluded some common South African names such as 'Precious' and 'Gift' from their gazetter, which is a list of common names, since it could confuse their tagger. Their named entity tagger achieved accuracies (measured as an F-score) of 0.42 and 0.67 with and without gazatteers respectively.

Eiselen [3] focussed on named entity recognition in government domain. Their focus was not specifically South African English, but rather South African Languages as a whole. They looked specifically at person names, organization names and locations. They achieved f-scores of roughly 0.75 for all the South African languages, except for isiZulu and SiSwati, both of which scored below 0.7.

2.2 Part-of-speech tagging

Being a very old field, part-of-speech tagging has been extensively studied, and even considered solved by some. The goal of part-of-speech tagging is to assign a role to each word in a sentence.

Tagging faces several challenged. The main challenge of part-of-speech tagging is figuring out what role a word plays. This is due to the fact that words may have several valid parts of speech [5]. Another challenge taggers face is dealing with unknown words. These could be words that the tagger did not encounter during it's training, loan words or neologisms [5].

Taggers may also make errors due to typographical errors in the text. Elworthy [4] suggests a technique to detect such errors using Hidden Markov Models. By comparing the observable values of the tagging process with a threshold, some proportion of tagged words may be traded for accuracy.

Testing the efficacy of a tagger requires some set of tagged words that we know to be correct. This requires a person (or people) to tag the text. Marcus, Santorini, and Marcinkiewicz [15] found that humans disaggree on 7.2% of tags assigned to words, leading to some ambiguity in testing, and thus giving some error. A solution to this is allowing words to have multiple tags and accepting each of those.

Tagging algorithms come in various different kinds such as

- Rule-based tagging [14, 1]
- Transformation-based tagging [2]
- Hidden Markov Models [12]
- Maximum Entropy Models [8, 16, 18]

Part-of-speech tagging is considered 'solved' because it has reached incredibly high accuracies of 96%[12], 96.6%[18], 95%[8], 97%[11] etc.

3 TESTING METHODOLOGY

3.1 The software

For this experiment we used four different part of speech taggers:

1. HunPos [7]
2. Stanford [22] Part of Speech Tagger
3. NLTK (Averaged Perceptron tagger) Project [17]
4. TreeTagger [23]

HunPos [7] is a free and open source reimplementation of TnT (Trigrams'n'Tags), a statistical part-of-speech tagger. Halácsy, Kornai, and Oravecz [6] tested this tagger and found an overall accuracy of 96.58%.

The Stanford [22] part of speech tagger is an open source tagger provided by the University of Stanford. Two papers were written on this tagger and they found accuracies up to 97% depending on whether the words were known or not [10, 11].

The TreeTagger [23] is a tagger developed by Helmut Schmid for the Institute for Computational Linguistics of the University of Stuttgart. They wrote two papers on the tagger and have achieved accuracy of up to 96% [20, 21].

The NLTK averaged perceptron tagger [17] is an implementation of a tagger based on the averages perceptron machine learning algorithm in the NLTK library.

All four taggers used the PENN Treebank tagset for easy comparison, and a simpler testing script.

All the data used in testing was from a corpus on South African English provided by a colleague, Umar Khan [9]. Of the data provided, three news outlets, and several blogs were used for testing. In total 34 articles/blogs were used to this experiment.

3.2 The methodology

Testing the accuracy of the tagger happened in stages. First the data from the corpus needed to be extracted into a usable format, in this case plain text.

The extracted data was then tagged by each of the four taggers.

The tagged data was then manually checked by us, and compared to each other for as much consistency as possible between the different taggers. In the cases where plasuible ambiguity was found, the tag was left as the repective tagger had tagged it. This was done for each tagger as some taggers would leave out some symbols and others would split words (such as possesive ending) onto a new line.

The checked data and the unchanged tagged data was then compared with each other.

We assume that the checked data is 100% with an error of roughly 7% based on [15]. That is to say that if we assume that 100% is as good as it gets, then a human tagger would be between 93% accurate and 100% accurate. This difference is our error estimation when calculating our error for the recall, where it is assumed that the article is taggers 100% correctly.

We then proceed to calculate an F-score for the accuracy of a specific article. We calculate a *precision score* using the number of words that the tagger correctly identified divided by the number of words identified (this mean leaving out words tagged as unknown or foreign word).

$$p = \frac{correct}{identified_{total}} \tag{1}$$

A *recall* score is then calculated as the number of words correctly identified divided by the total number of words in the article.

$$r = \frac{identified_{correct}}{words_{total}} \tag{2}$$

These two values are then combined into a harmonic mean to get the F-score.

$$f = \frac{2}{\frac{1}{r} + \frac{1}{p}} \tag{3}$$

Because we are dealing with a function of the form

$$f = c_1 A + c_1 B \tag{4}$$

our error calculation is as follows:

$$\sigma_f^2 = c_1^2 \sigma_A^2 + c_2^2 \sigma_B^2 \tag{5}$$

In our case, $c_1 = c_2 = 1$, and A and B are the f-scores for particular sub-corpora. Since the error on each article is the same 7%, our error for each score will be the same.

We now aggregate the scores of each tagger over all the articles it tagged to gain an idea of the overall accuracy of the tagger. This is again done using a harmonic mean.

$$f_{aggregate} = \sum_{i=1}^{j} \frac{1}{f_i} \tag{6}$$

$$f_{mean} = \frac{n_{total}}{f_{aggregate}} \tag{7}$$

Where j is the number of articles tagged, and n_{total} is the total number of words tagged by the tagger.

These scores are then compared to the accuracies mentioned by papers that tested the accuracies of these taggers, with a preference for the papers that were solely focussed on the single tagger in question.

4 RESULTS

Because individual results would be too verbose, results were compressed into their own sub-corpora, with over all accuracies over specific outlets, or the blogs. Interesting outliers will be highlighted.

The news outlets used for testing are *The Daily Voice*, *The Mail and Guardian*, abbreviated as **M&G**, *The South African Blue Sky Publications*, abbreviated as *Blue Sky*, and *news24*. The *Blogs* category is an assortment of various blogs written about South Africa by South Africans.

The Daily Voice is a Western Cape news outlet that uses more colloquial expressions in their writing compared to the other news outlets, which do national news.

	Stanford	**HunPos**	**NLTK**	**Tree**
Daily Voice	0.950	0.956	0.959	0.942
M&G	0.964	0.967	0.967	0.968
Blue Sky	0.95	0.957	0.952	0.960
news24	0.968	0.968	0.97	0.974
Blogs	0.962	0.938	0.934	0.957
Aggregate	0.949	0.955	0.953	0.96

Since we have the same error for every score calculated, we need only calculate the error score once:

$$\sigma_f = 0.163 \tag{8}$$

5 DISCUSSION

All results were obtained using python scripts that automatically iterate over several tagged articles and comparing the tag that the tagger asigned with what the tagger should have tagged the word as.

This leaves room for errors as human taggers are not infallible. Room for ambiguity is also left over as some words may be multiple valid tags in any given context, so different taggers may tag the same word differently in the exact same context. An example of this is the word 'to'. The PENN Treebank Tagset has a dedicated tag for it, *TO*, however the taggers would sometimes tag it as *IN*, a preposition or subordinating conjuction, which in some cases is also correct. In such instances, the tag was left unchanged.

Another hurdle the taggers faced were punctuation marks, most notably part of contractions and posessive markers such as ''s'. Neither the Stanford tagger, nor the HunPos tagger correctly identified and posessive markers, and mostly got contractions right per chance. This may very well be due to the formatting of the data, as the data was cleaned of any special characters, to contain only ANSI characters. NLTK managed to correctly identify few part of speech tags.

Additionally, the Stanford tagger, albeit the best at identifying foreign words in general, nearly always tagged the first word of a sentence as a proper noun, and once it got confused, it stayed confused for a few words before correcting itself. This means that a single foregin word tag may leave the rest of the sentence tagged as forgeign words, even if only one word in the sentence was actually a loan word.

The formatting of the data itself, as already mentioned is one of the factors for the efficacy of these the taggers. Sometimes words would not split correctly, or due to the format of the data, special characters were left out, causing confusion in the taggers.

Across all taggers, a couple of instances where words were not split correctly was noted, and the taggers very rarely reported forgein words. In the cases where foreign words were tagged, they were usually tagged correctly.

The Treetagger has an extra column of output when tagging, which puts the base form word next to the tag. Here, if the word wasn't identified, '<unknown>' was displayed next to it. The words tagged as unknown wasn't specifically loan words, but seemed to be rather inconsistent, as it would be unable to identify words like 'especially' at times. This allows us to get two different values from the Treetagger, one where we count all instances of unknown words as foreign, and another where we only treat unknown and incorrectly tagged words as foreign. The data table in Results section is when all unknown words are considered to be forgein. If we do not consider all unknown words to be foreign, the accuracy of the tagger becomes astounding:

	Tree
Dail Voice	0.96
M&G	0.98
Blue Sky	0.98
news24	0.98
Blogs	0.97
Aggregate	0.96

The most interesting set of data that was tagged is the blogs, as we have some special outliers in this section. Because they are blogs, we see much less formal writing, and the accuracy of the taggers reflect this.

A blog written about South African slang was among the lowest scores recorder for each of the taggers, even when loan words were used in some of the other blogs and news articles. The blog in question got the following scores:

Stanford	**HunPos**	**NLTK**	**Tree**
0.875	0.925	0.922	0.925

A reason for this may be because the words are introduced in isolation without any additional context.

Similarly, when social media language was being used, such as Twitter's handles, or hashtags, the taggers became unable to correctly identify the part of speech to which these entities belonged.

6 CONCLUSIONS

We have successfully tested the four different taggers against various contexts of South African English.

Firstly we found that our taggers managed to be as accurate tagging South African English as they are tagging other English variants, depending on the context and formality of the register used, as when we compare our scores to the scores seen in [6, 10, 11, 20, 21], we see that HunPos did reach 96% in some contexts, and the Stanford tagger also managed to reach 97% accuracy. The Treetagger performed as well as expected, reaching its claimed accuracy of up to 97%. Although we found that the taggers do reach these accuracies in certain domains, their overall performance fell short of the values quoted in the mentioned papers. Even if we are talking of 1-2%, when we are reaching accuracies of 97%, that 1-2% means a lot. If we were to tag an article of tens of thousands of words, having 1-2% fewer correctly tagged words could mean we have 100's of words tagged incorrectly. Even though the taggers fell short, they still beat out our expectations when it came to tagging South African English.

Secondly, we can clearly see a marked improvement in efficacy when the domain and register of the text being tags shifts. The Mail and Guardian, and news24 both write very formally, and our data reflects that these outlets' pieces are tagged more accurately than other outlets' such as the Daily Voice or blog posts. This is as we expected it to be.

Finally, three of our taggers use context to determine the tags of the words; the Stanford tagger uses a cyclic dependancy network [11], and both HunPos and the Treetagger are based on hidden markov models [21, 6]. The final tagger, NLTK, uses perceptron nodes, a machine learning technique to tag words. This does not include the ability to recognize context, yet it pdid not perform significantly worse than any of the other taggers.

7 FUTURE WORK

Certainly the domain of testing can be widened given less of a time constraint. This would provide more data and more context as to what eactly it is that the taggers fail at (if truly anything at all) when it comes to tagging South African English.

Additionally, even though existing taggers are practically as good as can get (based on our relatively small sample size) when it comes to tagging South African English, research into creating a tagger specifically for South African English may not go amiss.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

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

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

[6] Péter Halácsy, András Kornai, and Csaba Oravecz. "HunPos-an open source trigram tagger". In: (2007).

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

[8] 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.

[9] Badenhorst Alec Khan M. Umar. "Project Proposal: Building a South African English corpus and Assessing Part of Speech tagging accuracy." In: (2020).

[10] Christopher D Manning Kristina Toutanova. "Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger". In: *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*. 2000, pp. 63–70.

[11] et. al. Kristina Toutanova Dan Klein. "Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network". In: *Proceedings of HLT-NAACL 2003*. 2003, pp. 252–259.

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

[13] 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.

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

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

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

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

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

[19] 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).

[20] Helmut Schmid. "Improvements in Part-of-Speech Tagging with an Application to German". In: *Proceedings of the ACL SIGDAT-Workshop. Dublin, Ireland.* 1995.

[21] Helmut Schmid. "Probabilistic Part-of-Speech Tagging Using Decision Trees". In: *Proceedings of International Conference on New Methods in Language Processing, Manchester, UK.* 1994.

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

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