View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Ghent University Academic Bibliography

African Language Technology: The Data-Driven Perspective

Guy De Pauw and Gilles-Maurice de Schryver

In this paper we outline our recent research efforts, which introduce data-driven methods in the development of language technology components and applications for African languages. Rather than hard-coding the solution to a particular linguistic problem in a set of hand-crafted rules, data-driven methods try to extract the required linguistic classification properties from annotated corpora of the language in question. We describe our efforts to collect and annotate corpora for African languages and show how one can maximise the usability of the (often limited) data with which we are presented. The case studies presented in this paper illustrate the typical advantages of using data-driven methods in the context of natural language processing, namely language independence, development speed, robustness and empiricism.

1. Introduction

Most research efforts in the field of natural language processing (NLP) for African languages are still firmly rooted in the rule-based paradigm. Language technology components in this sense are usually straight implementations of insights derived from grammarians. While the rule-based approach definitely has its merits (particularly in terms of design transparency) it has the distinct disadvantage of being highly language-dependent and costly to develop, as it typically involves a lot of expert manual effort.

Furthermore, many of these systems are decidedly *competence*-based. The systems are often tweaked and tuned towards a small set of ideal sample words or sentences, ignoring the fact that real-world language technology applications have to be principally able to handle the *performance* aspect of language. Many researchers in the field of African language technology are quite rightly growing weary of publications that ignore quantitative evaluation on real-world data or that report unreasonably high accuracy scores, excused by the erroneously perceived regularity of African languages.

In a linguistically diverse and increasingly computerised continent such as Africa, the need for a more economical approach to language technology is high. In this paper we outline our recent research efforts, which introduce data-driven methods in the development of language technology components and applications for African languages. Rather than hard-coding the solution to a particular NLP problem in a set of hand-crafted rules, these data-driven methods try to automatically extract the required linguistic classification properties from large, annotated corpora of natural language.

We describe our efforts to collect and annotate these corpora and show how one can maximise the usability of the (often limited) data with which we are presented. We focus on the following aspects of using data-driven approaches to NLP for African languages, and illustrate them on the basis of a few cases studies:

- **Language independence:** we show how the same technique can be used to perform diacritic restoration for a wide variety of resource-scarce African languages (Ciluba, Gikuyu, Kikamba, Maa, Northern Sotho, Venda and Yoruba).
- **Development Speed:** we illustrate how a small, annotated corpus can be used to develop a robust and accurate part-of-speech tagger for Northern Sotho.
- **Robustness:** our case study of Swahili memory-based lemmatisation shows that a data-driven technique can rival a rule-based approach not only in terms of development speed, but also in terms of classification accuracy.
- **Empiricism:** all three case studies show how language technology components can be simultaneously developed and evaluated using real-world data, offering a more realistic estimation of their usability in a practical setting.

2. Corpus Collection and Normalisation: A Language-Independent Approach to Automatic Diacritic Correction

Early work in computational linguistics was burdened by the practical limitations of computational power and storage, preventing the use of large, annotated corpora. This all changed in the late 1980s when researchers started unearthing the full use of the language corpus, using statistical approaches and machine-learning techniques. In a matter of years, rule-based approaches had fallen out of favour in the research community and the new language-independent *performance* models had taken over most of the publications in the field.

2. 1 Corpus collection

While the corpus-based approaches were readily applicable to the world's most commercially interesting languages, resource-scarce languages were left behind. By definition, these languages are low on linguistic resources, with very few digital corpora available to them, let alone annotated data. For a long time, this forced researchers working on such languages to stick to the empirically less demanding rule-based paradigm, further alienating them from the main scientific current in NLP. This is even the case for a language like Swahili: despite being spoken by more than fifty million people, it is still a lesser-used language from a language technological point of view.

The proliferation of the Internet in the urban areas of Africa, however, meant that more and more vernacular language data became available in a digital format. This not only increases the visibility of African languages in the world, but now also enables the collection of large corpora, through web crawling the available content on the Internet (de Schryver 2002).

2. 2 Corpus normalisation

Unfortunately this type of user-generated corpus material comes at a cost, since its consistency and cleanliness cannot be guaranteed. This poses a particular problem for languages that have diacritically marked characters in their orthography. Despite an increasing awareness of encoding issues and the development of specialised fonts and computer keyboards (ANLoc 2009), many digital language resources do not use the proper orthography of the language in question, with accented characters represented by their unmarked equivalents. While language users can often perform real-time disambiguation of unmarked text while reading, a lot of phonological, morphological and lexical information is lost in this way–information that could be useful in the context of language technology.

Most automatic diacritic restoration methods tackle both the actual task of retrieving diacritics of unmarked text and the related tasks of part-of-speech tagging and word-sense disambiguation (e.g. Yarowski 1994). Although complete diacritic restoration ideally involves a large amount of syntactic and semantic disambiguation, this type of analysis can typically not be done for resource-scarce languages. Moreover, these methods rely heavily on lookup procedures in large lexicons, which are usually not available for such languages.

2. 3 Grapheme-based diacritic correction

One of the first applications of machine-learning techniques to an African language technology problem was presented in (Wagacha et al. 2006) for Gikuyu and expanded in (De Pauw et al. 2007) for a wider range of African languages. The basic method, adapted from (Mihalcea 2002), uses an alternative approach to diacritic restoration: it uses a machine-learning technique operating on the level of the grapheme. The general idea of the approach is that local orthographic context encodes enough information to solve the disambiguation problem. By backing off the problem from the word level to the grapheme level, it opens up the possibility of diacritic restoration for languages that have no electronic word lists available.

The training material for our approach is a word list for the language in question that contains all the proper diacritics. This word list can be the result of selecting properly encoded documents from a web crawling session. We then identify for each language the *confusables:* those characters that can occur with or without diacritics.

The diacritic correction task is identified as a classic machine-learning task, where we associate a number of features with a given class. This is illustrated in Table 1 for the Gikuyu word *mbũri*. We first strip the word of all its diacritics. Then, for each character in the word (F), we identify a window of five characters to the left (L) and five characters to the right (R). Finally, these features are associated with a class (C), which features the correct character. Instance 3 in Table 1, for example, describes the confusable *u*, which in Gikuyu orthography can be either *u* or *ũ*. In this case, the correct class is ũ. Similarly in Instance 5, the confusable *i*, should be represented as *i* instead of ũ.

	L1	L2	L3	L4	L5	F	R1	R2	R3	R4	R5	C
1	-	-	-	-	-	**m**	b	u	r	i	-	**m**
2	-	-	-	-	m	**b**	u	r	i	-	-	**b**
3	-	-	-	m	b	**u**	r	i	-	-	-	**ũ**
4	-	-	m	b	u	**r**	i	-	-	-	-	**r**
5	-	m	b	u	r	**i**	-	-	-	-	-	**i**

Table 1: Instances for Gikuyu diacritic restoration task

Instances are extracted for each character in each word in the word list and presented to the memory-based learner TiMBL (Daelemans et al. 2004) as training material. This data is stored in memory. Diacritics can now be restored for previously unseen words by deconstructing the word in the same vein. The second confusable in the word *umbũre* for example, is represented in Table 2. Its class is unknown, but it shares nine features with Instance 3 in Table 1 (namely L1, L2, L4, L5, F, R1, R3, R4 and R5). If Instance 3 turns out to be the most similar entry in memory, its class is extrapolated and suggested as the class for the instance in Table 2.

L1	L2	L3	L4	L5	F	R1	R2	R3	R4	R5	C
-	-	u	m	b	**u**	r	e	-	-	-	**???**

Table 2: Classification of new Gikuyu word

We compiled data for a wide range of African languages that have diacritically marked characters in their orthography: the Bantu languages Ciluba (Congo), Gikuyu, Kikamba (Kenya), Northern Sotho, Venda (South Africa), the Nilotic language Maa (Kenya) and the Defoid language Yoruba (Nigeria). We applied the exact same machine-learning technique to all of the languages to perform diacritic restoration.

The experimental results are displayed in Table 3. We evaluate the performance of our system (**MBL**) on a portion of the corpus that was not used in the training of the system. We compare our results to that of a lexicon lookup approach (**LLU**), which retrieves the diacritically marked variant of a word from the lexicon induced from the training set. Whereas the LLU approach by definition fails on previously unseen words, the memory-based approach working on the grapheme level is always equipped to make a calculated guess.

Language	Types	LLU	MBL
Ciluba	20.0k	77.0	85.3
Gikuyu	9.1k	77.3	92.4
Kikamba	9.7k	79.4	91.6
Maa	22.2k	66.7	75.5
Northern Sotho	157.8k	97.6	99.2
Tshivenda	9.6k	97.7	99.4
Yoruba	4.2k	67.8	76.8

Table 3: Diacritic restoration results

The results indeed show that the memory-based approach significantly outperforms a lexicon lookup method for all of the languages, sometimes with as few as 10,000 words in the training data. This is not surprising, given the morphological richness of these languages and consequently the high number of previously unseen words in the test data. While for some languages (e.g. Northern Sotho) diacritic restoration is close to a solved problem, the restoration of tonal diacritics appeared to be more problematic on the basis of graphemic data for others (e.g. Yoruba and Maa). Nevertheless, this research showed that the same, relatively simple set of preprocessing scripts and the same machine-learning technique, can be employed to a wide range of languages on the African continent, even when relatively little data is available.[1]

3. Corpus Annotation: Rapid Development of a Robust Part-of-Speech Tagger for Northern Sotho

Our work on diacritic restoration was one of the first published attempts at applying machine-learning techniques to African language technology. Previously we had described experiments with data-driven part-of-speech (POS) taggers for Swahili (De Pauw et al. 2006), trained and evaluated on the three million-word POS-tagged part of the Helsinki Corpus of Swahili (Hurskainen 2004a).

Many researchers assume that data-driven approaches to NLP require hundreds of thousands of annotated tokens. Inspired by the encouraging results that even smaller data sets had yielded, as seen in Table 3, we decided to build a small, manually POS-tagged corpus of Northern Sotho and develop a data-driven tagger on the basis of this data (de Schryver & De Pauw 2007).

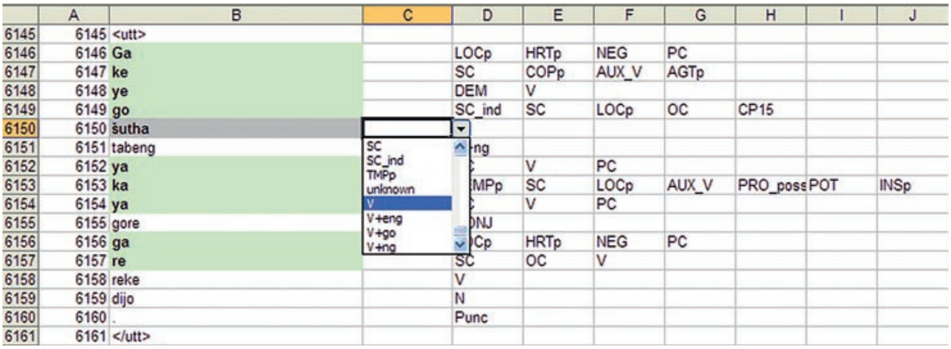
3. 1 Data annotation

The annotation was set up as an exercise in development speed. We pre-defined a list of around 50 PoS tags for Northern Sotho, but allowed annotators to refine the protocol during annotation. This on-the-fly approach enabled the organic construction of a consistent tag set, grounded in linguistic, corpus-based evidence.

Furthermore, despite the availability of dedicated annotation tools, we used Microsoft Excel as the annotation environment of choice. Installation is trivial and most computer-literate users are familiar with the Microsoft Office Suite, so that the learning curve for the annotators is favourable. While annotation is an unlikely use of a spreadsheet, Excel's cell-based approach can significantly speed up an annotation task such as PoS-tagging, also allowing on-line adjustments of the tagging protocol.

[1] A demonstration system for diacritic restoration of the languages in Table 3 can be found at http://aflat.org/?q=node/184

The annotation environment is illustrated in Figure 1: Column B contains the word to be tagged, while columns D, E, F, etc. provide the possible tags for the word, as retrieved from the TshwaneDJe HLT Northern Sotho lexical database. The correct tag for ambiguous words (highlighted in light-grey) is selected by the annotator. An additional drop-down box in Column C is available if the correct tag is not featured in columns D, E, F, etc. This is by definition the case for previously unseen words (highlighted in dark-grey). Should earlier annotations need to be adjusted for some reason, Column B can be sorted alphabetically, while the indices in Column A ensure the original order of the document can be restored.



	A	B	C	D	E	F	G	H	I	J
6145	6145	<utt>								
6146	6146	Ga		LOCp	HRTp	NEG	PC			
6147	6147	ke		SC	COPp	AUX_V	AGTp			
6148	6148	ye		DEM	V					
6149	6149	go		SC_ind	SC	LOCp	OC	CP15		
6150	6150	šutha								
6151	6151	tabeng		ng						
6152	6152	ya		C	V	PC				
6153	6153	ka		MPp	SC	LOCp	AUX_V	PRO_poss	POT	INSp
6154	6154	ya		C	V	PC				
6155	6155	gore		NJ						
6156	6156	ga		Cp	HRTp	NEG	PC			
6157	6157	re		SC	OC	V				
6158	6158	reke		V						
6159	6159	dijo		N						
6160	6160	.		Punc						
6161	6161	</utt>								

Figure 1: Excel sheet containing the initial POS-tagging material for the annotator

Restricted to a total annotation time of a mere 10 person-hours, the design of the annotation environment nevertheless maximised the amount of annotated data. After post-processing the data, we obtained a manually tagged corpus of more than 10,000 words, in a format ready to be used as training material for a data-driven tagger:

(1) Ke_SC a_PRES eletša_V ._Punc

While a 10,000-word-tagged corpus is indeed modest, compared to the million-word corpora available for English (Marcus et al. 1993), the experimental results (Section 3.3) show that even a small annotated data set can yield an accurate data-driven PoS-tagger.

3. 2 MaxTag

We used the annotated data to train and evaluate a POS-tagger based on the machine-learning technique of maximum entropy (Berger et al. 1996). Rather than the stock maximum entropy tagger, MXPOST (Ratnaparkhi 1996), we used a self-constructed POS-tagger, called MaxTag, which acts as a front-end to the general machine-learning package Maxent (Le 2004).

MaxTag takes as its input POS-tagged data (e.g. example (1)) and extracts for each word in the corpus a number of features that are possibly relevant to the disambiguation problem. Similar to the diacritic restoration approach, MaxTag uses a windowing approach to describe linguistic events. Instead of working on the character level however, MaxTag describes the problem on the word level, extracting for each word in the corpus an instance that contains both contextual and orthographic features. Each instance is then associated with the POS-tag for that word.

	Instance	**Tag**
1	[‚W-1=#‘,‚T-1=#‘, **‚FW=Ke‘**,‚FT=SC_COPp‘,‚W+1=a‘,‚T+1=SC_PRES_PC_DEM_OC_HRTp‘,‚P1=K‘,‚S1=e‘, ‚P2=Ke‘,‚S2=Ke‘,‚CAP‘]	**SC**
2	[‚W-1=Ke‘,‚T-1=SC‘, **‚FW=a‘**, FT=SC_PRES_PC_DEM_OC_HRTp‘,‚W+1=eletša‘,‚T+1=V‘,‚P1=a‘,‚S1=a‘]	**PRES**
3	[‚W-1=a‘,‚T-1=PRES‘, **‚FW=eletša‘**,‚FT=V‘,‚W+1=.‘,‚T+1=Punc‘,‚P1=e‘,‚S1=a‘,‚P2=el‘,‚S2=ša‘,‚P3=ele‘, ‚S3=tša‘]	**V**
4	[‚W-1=eletša‘,‚T-1=V‘, **‚FW=.‘**,‚FT=Punc‘,‚W+1=#‘,‚T+1=#‘,‚P1=.‘,‚S1=.‘]	**Punc**

Table 4: Four instances for Example (1)

Example instances are displayed in Table 4. The focus word (**FW=**) is associated with previous and subsequent words (**W±*n*=**) and tags (**T±*n*=**) and a list of possible tags for the word itself (**FT=**). MaxTag also allows for the inclusion of character clusters as morphological features towards disambiguation, which is valuable for processing morphologically rich languages.

On the basis of these instances, the maximum entropy machine learner constructs a statistical model that optimally relates features to classes. The advantage of using maximum entropy for this problem is that instances do not need to have a value for all features, making it more robust for sparse data sets.

3. 3 Experimental results

To evaluate the tagger, we performed 10-fold cross validation. The corpus is divided into 10 slices. In each experiment, one slice is used as the evaluation set, while the other nine are used as training data. We distinguish between known words (words in the evaluation set that are present in the training data) and unknown words (words in the evaluation set, not occurring in the training data). The latter category of words averages to about 8 % of the words in a typical evaluation set. The experimental results (Table 5) show that, despite the minimal amount of training data, the tagger is able to significantly outperform a baseline tagger (unigram probabilistic tagger). It achieves an overall tagging accuracy of 93.5 %. Nevertheless, the more modest score for unknown words indicates that more annotated data can still significantly improve the accuracy of the tagger.

	Known	Unknown	Total
Baseline	75.8	35.1	73.5
MaxTag	95.1	78.9	93.5

Table 5: Accuracy scores for the baseline method and MaxTag (all values in %)

This is corroborated by the learning curve experiments we conducted. In these experiments we started out with a POS-tagger trained on just 1/10 of the available training data (roughly 1000 words) and added 1/10 of the training data in each subsequent experiment. The result for this experiment can be found in the learning curve graph, displayed in Figure 2. The graph shows that the learning curve is still linear and that we can still gain quite a bit of tagging accuracy by collecting more annotated data. Future research will therefore concentrate on the semi-automatic development of new annotated data. The more-than-encouraging experimental results show that the Northern Sotho version of MaxTag can provide an invaluable tool in this endeavour.[2]

[2] A demonstration system of the Northern Sotho POS-tagger can be found at http://www.aflat.org/?q=node/177

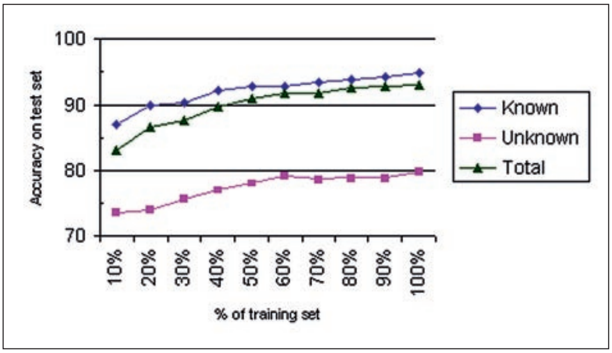


Figure 2: Graph for learning curve experiments

4. Corpus Extraction: Robust and Accurate Morphological Analysis of Swahili

The Helsinki Corpus of Swahili, HCS (Hurskainen 2004a) is the only large-scale annotated corpus of a Bantu language available to date. Annotation layers include POS-tagging and lemmatisation. It is important to note, however, that these have been generated by an automatic finite-state method, SALAMA (Hurskainen 2004b) and have not been manually cross-checked nor empirically evaluated. In this section, which draws on the work of De Pauw and de Schryver (2008), we present experiments that allow for a direct comparison between a meticulously designed rule-based approach to morphological analysis (SALAMA) and an alternative based on the machine-learning technique of memory-based learning (MBSMA).

4. 1 Data preparation

To construct a memory-based system for morphological analysis, we require morphologically annotated data. This is however not available for Swahili, but we can go a long way by extracting the necessary information from HCS, lemmatised using the SALAMA morphological analyser.

In HCS, every word is lemmatised, as illustrated in examples (2) and (3):

(2)	ulikanusha	kanusha
(3)	ulikonzia	anza

We can use this information to perform pattern-matching and match the lemma to the word form. Through this operation we can automatically induce a morphologically segmented surface and lexical representation of the word form, in which we distinguish a prefix-group ([P]), the root morpheme ([R]) and a suffix group ([S]). In some cases, this is straightforward, like for the entry in example (2) which can easily be transformed into example (4):

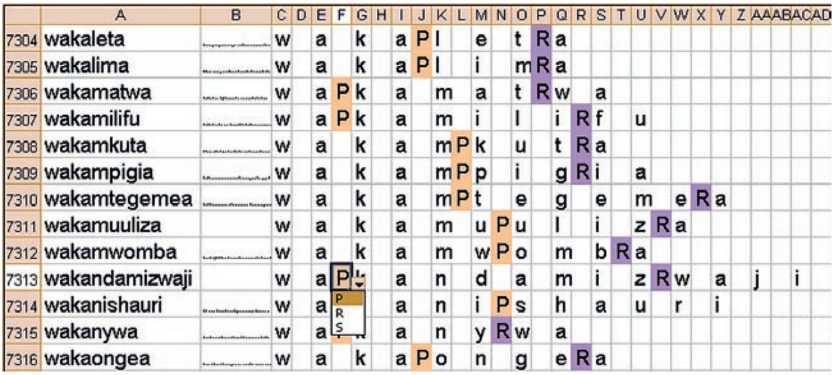
(4) ulikanusha kanusha → Surface: uli[P] + kanusha[R]
→ Lexical: uli[P] + kanusha[R]

For the entry in example (3), this leads to the creation of a bound root morpheme *anz-* in the surface representation, associated with the full lemma *anza* in the lexical representation:

(5) ulikonzia anza → Surface: uliko[P] + anz[R] + ia[S]
→ Lexical: uliko[P] + anza[R] + ia[S]

Using this method we automatically extracted a morphological database of 97,000 entries from HCS. Since HCS has been lemmatised using an automated method, quite a few erroneous and inconsistent lemmatisations can be observed in the data. We therefore randomly extracted 10 % of the data from the morphological database and had it manually annotated according to the prefix-root-suffix protocol illustrated in examples (4) and (5). The availability of this manually annotated, gold-standard evaluation set does not only allow us to cross-check the accuracy of our system on clean data, but also enables a post-hoc quantitative evaluation of the rule-based approach used to annotate HCS.

Similarly to the annotation approach described in Section 3.1, we again used Microsoft Excel as the annotation environment. The annotation sheet seen in Figure 3, lists each word on a separate row. The word form itself is listed in Column A. Column B contains a sentence extracted from HCS, illustrating that word form in context. The minimised sentence can be displayed in full by double-clicking on the cell. Columns C and onwards list the individual characters of the word form from Column A, separated by blank cells. Each blank cell has a drop-down box available with three options: P (end of prefix group), R (end of root group) and S (end of suffix group). The annotator can quickly move through the annotation process using only the keyboard or mouse clicks.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD
7304	wakaleta		w		a		k		a	P	l		e		t	R	a													
7305	wakalima		w		a		k		a	P	l		i		m	R	a													
7306	wakamatwa		w		a	P	k		a		m		a		t	R	w		a											
7307	wakamilifu		w		a	P	k		a		m		i		l		i	R	f		u									
7308	wakamkuta		w		a		k		a		m	P	k		u		t	R	a											
7309	wakampigia		w		a		k		a		m	P	p		i		g	R	i		a									
7310	wakamtegemea		w		a		k		a		m	P	t		e		g		e		m		e	R	a					
7311	wakamuuliza		w		a		k		a		m		u	P	u		l		i		z	R	a							
7312	wakamwomba		w		a		k		a		m		w	P	o		m		b	R	a									
7313	wakandamizwaji		w		a	P	k		a		n		d		a		m		i		z	R	w		a		j		i	
7314	wakanishauri		w		a	P R S			a		n		i	P	s		h		a		u		r		i					
7315	wakanywa		w		a				a		n		y	R	w		a													
7316	wakaongea		w		a		k		a	P	o		n		g		e	R	a											

Figure 3: Excel sheet containing the morphological material for the annotator

In this way, the surface representation of the morpheme boundaries is annotated. In a second annotation step, the lexical representations of the roots, thus the actual lemmas, are double-checked and corrected where necessary.

4. 2 Memory-based morphological analysis

The memory-based Swahili morphological analyser reuses and adapts the basic methodology coined in van den Bosch and Daelemans (1999), which has been successfully applied to morphologically rich(er) languages such as Dutch (De Pauw et al. 2004) and Arabic (van den Bosch et al. 2007).

We use the dataset described in Section 4.1 as our primary information source. Analogous to the method described in Sections 2.3 and 3.2, we use a windowing approach to represent the data. Instead of using characters (Section 2.3) or words and tags (Section 3.2), we describe this problem at the level of the syllable. This is a more appropriate level of description when dealing with Bantu morphology than the character level, originally used in van den Bosch and Daelemans (1999).

We describe each syllable in the word, associated with a context on the left-hand side and a context on the right-hand side. This is illustrated for the word *ulikoanzia* in Table 6. Here each syllable is linked to a class. The syllable *ko* in Instance 3 marks the end of the prefix-group, while syllable *a* in Instance 7 marks the end of the suffix group. The syllable *zi* (Instance 6) marks the end of the root group and receives an extra instruction that the root needs to be repaired to the full lemma *anza* by deleting the *i* (part of the suffix group) and adding an *a* to the bound morpheme *anz-*. As was the case for the diacritic restoration task, new instances are classified by comparing them to the ones in memory and extrapolating the class of the most similar instance.

	L	L	L	L	L	F	R	R	R	R	R	CLASS
1	–	–	–	–	–	**u**	li	ko	a	nzi	a	0
2	–	–	–	–	u	**li**	ko	a	n	zi	a	0
3	–	–	–	u	li	**ko**	a	n	zi	a	–	P
4	–	–	u	li	ko	**a**	n	zi	a	–	–	0
5	–	u	li	ko	a	**n**	zi	a	–	–	–	0
6	u	li	ko	a	n	**zi**						R+a
7	li	ko	a	n	zi	**a**	–	–	–	–	–	S

Table 6: Syllable-based instances extracted from the morphological database

4. 3 Experimental results

We are most interested in the accuracy of the morphological analyser on previously unseen words: how well is the system able to morphologically segment and lemmatise unknown word forms? To investigate this, we perform blind testing, withholding a 10 % partition of the data to evaluate the system. As the evaluation set, we naturally use the manually annotated gold standard evaluation set, described in Section 4.1.

The gold-standard evaluation set also allows us to quantify the accuracy of SALAMA, the rule-based approach used to lemmatise the Helsinki Corpus of Swahili. We will follow the standard approach of using word-error rate (WER) as our primary evaluation metric. It expresses the accuracy on the word-level, that is, how many words have not been completely correctly segmented and lemmatised. In other words, the lower the word-error rate (WER), the better the system.

The experimental results in Table 7 show that the memory-based approach (**MBSMA**) can be observed to outperform SALAMA, establishing a small reduction in WER on surface-level segmentation and a more substantial reduction for full lemmatisation (i.e. full restoration of the underlying lemma).

	Segmentation of the surface representation	Further lemmatisation
	WER	**WER**
SALAMA	11.7%	12.0%
MBSMA	11.6%	11.7%

Table 7: Accuracy scores for SALAMA and MBSMA on the manually annotated evaluation set

This result may be surprising: how can a data-driven approach outperform the system that was used to create its information source? The answer to this question lies in the generalisation capabilities of the machine-learning technique. As previously mentioned, and as further illustrated by the SALAMA results in Table 7, quite a few erroneous analyses can be found in the annotation of HCS. Rather than completely mimicking the properties of the data the machine-learning approach uses to train its model, it implicitly generalises over the data and filters out the noise. This eventually generates a more accurate lemmatiser for the data in question.

The biggest advantage is the robustness of the memory-based approach: it does not rely on any kind of underlying lexicon of root forms or lemmas. When presented with an unknown word form or even a word form for a previously unseen lemma, the memory-based approach will *degrade gracefully* and guess the lemma with a surprisingly high degree of accuracy.

To the best of our knowledge, the research results presented in this section describe the first attempt at building a data-driven morphological analyser for a Bantu language. We have demonstrated how this system can be properly and quantitatively evaluated with relatively little manual effort, and experimental results show that the method compares favourably to a meticulously designed rule-based technique, even when it is trained on the basis of its output. Defining the problem of data-driven morphological analysis on the level of the syllable, rather than on the character level, furthermore showed how techniques typically designed with Indo-European language processing in mind, can be adjusted to work for Bantu languages as well.[3]

5. Conclusion: An Empirical Approach to African Language Technology

In this paper we presented an overview of on-going work on applying data-driven techniques to natural language processing of African languages. We demonstrated the **language-independent** aspects of data-driven NLP by applying the same technique to the problem of diacritic correction of a varied array of African languages. The goal of such a system goes well beyond simple diacritic restoration: the orthography of most African languages is (morpho-)phonological in nature with a mostly unambiguous mapping between phoneme and grapheme. A good diacritic restoration method, in other words, basically amounts to a robust grapheme-to-phoneme conversion method that can be used as a front-end for text-to-speech systems.

[3] A demonstration system of the Swahili lemmatiser can be found at http://www.aflat.org/?q=node/241

We then demonstrated how data-driven techniques can result in the **rapid development** of a part-of-speech tagger for Northern Sotho. Rather than investing time in designing extensive tagging protocols and painstakingly implementing expert knowledge, we showed how a small, annotated data set, constructed in about 10 hours, can already yield an accurate part-of-speech tagger. This tagger can then serve as the basis of future annotation efforts, further unlocking the language technology potential of this resource-scarce language.

We finally showed how a **robust** memory-based lemmatiser can be constructed on the basis of automatically annotated data. This research showed how previous rule-based efforts can go hand in hand with a data-driven approach and help construct a more accurate lemmatiser that is inherently capable of analysing previously unseen word forms, even when the underlying lemma is unknown. The lemmatiser is currently being used as a preprocessing module in the context of machine translation for the language pair English—Swahili (De Pauw et al. 2009).

Possibly the most important aspect of our research from a methodological point of view is its inherent **empiricism**. Working in the data-driven paradigm automatically enables quantification of research results, something that up to now has all too often been ignored in research efforts in the field of computational linguistics for African languages. Our experimental results do not only serve to showcase the strength of our approach, but more importantly help to indicate those areas that need to be further developed. This has served to create a more competitive research field, with many recent publications adapting and improving on the approaches described in this paper (Faaß et al. 2009; Groenewald 2009).

Our research efforts constitute the first thorough exploration of data-driven methods for African language technology, and experimental results show that this is indeed the way forward if we are to re-introduce African languages on the scientific agenda of the NLP research community. Furthermore, in the context of Africa's linguistic diversity, as well as the resource-scarceness of the languages in question, we believe the data-driven paradigm, with its language independence, fast development phase and its focus on creating robust *performance* models of language, is the most appropriate approach to African language technology.

Acknowledgements

The research presented in this paper was partly funded by the SAWA BOF UA-2007 project. The first author is funded as a Postdoctoral Fellow of the Research Foundation - Flanders (FWO). We wish to thank Peter Waiganjo Wagacha for his continued scientific input, as well as the annotators without whom data-driven NLP would be impossible.

References

ANLoc (2009). Retrieved February 24, 2009, from http://www.africanlocalisation.net

Berger, A. L. / Della Pietra, S. / Della Pietra, V. J. (1996). "A Maximum Entropy Approach to Natural Language Processing", *Computational Linguistics*, 22(1), 39–71.

Daelemans, W. / Zavrel, J. / van den Bosch, A. / van der Sloot, K. (2004). *TiMBL: Tilburg Memory Based Learner, version 5.1, Reference Guide*. ILK Technical Report 04–02. Tilburg University.

De Pauw, G. / de Schryver, G.-M. (2008). "Improving the Computational Morphological Analysis of a Swahili Corpus for Lexicographic Purposes", *Lexikos* 18, 303–318.

De Pauw, G. / de Schryver, G.-M. / Wagacha P. W. (2006). "Data-driven part-of-speech tagging of Kiswahili" in *Proceedings of Text, Speech and Dialogue, 9th International Conference*. Berlin: Springer Verlag, 197–204.

De Pauw, G. / Laureys, T. / Daelemans, W. / Van Hamme, H. (2004). "A comparison of two different approaches to morphological analysis of Dutch" in *Proceedings of the Seventh Meeting of the ACL Special Interest Group in Computational Phonology.* Barcelona: ACL, 62–69.

De Pauw, G. / Wagacha, P. W. / de Schryver, G.-M. (2007). "Automatic diacritic restoration for resource-scarce languages" in *Proceedings of Text, Speech and Dialogue, 10th International Conference.* Berlin: Springer Verlag, 170–179.

De Pauw, G. / Wagacha, P. W. / de Schryver, G.-M. (2009). "The SAWA Corpus: a Parallel Corpus English—Swahili" in *Proceedings of the First Workshop on Language Technologies for African Languages*. Athens: ACL, 9–16.

de Schryver, G.-M. (2002). "Web for/as Corpus: A Perspective for the African Languages", *Nordic Journal of African Studies*, 11(2), 266–282.

de Schryver, G.-M. / De Pauw, G. (2007). "Dictionary Writing System (DWS) + Corpus Query Package (CQP): The case of TshwaneLex", *Lexikos*, 17, 226–246.

Faaß, G. / Heid, U. / Taljard, E. / Prinsloo, D. J. (2009). "Part-of-Speech Tagging of Northern Sotho: Disambiguating Polysemous Function Words" in *Proceedings of the First Workshop on Language Technologies for African Languages*. Athens: ACL, 38–45.

Groenewald, H. J. (2009). "Using Technology Transfer to Advance Automatic Lemmatisation for Setswana" in *Proceedings of the First Workshop on Language Technologies for African Languages*. Athens: ACL, 32–37.

Hurskainen, A. (2004a). *HCS 2004 - Helsinki Corpus of Swahili*. Compilers: Institute for Asian and African Studies. University of Helsinki and CSC.

Hurskainen, A. (2004b). "Swahili Language Manager: A Storehouse for Developing Multiple Computational Applications", *Nordic Journal of African Studies*, 13(3), 363–397.

Le, Z. (2004). *Maximum Entropy Modeling Toolkit for Python and C++*. Technical Report. Retrieved February 24, 2009, from http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html

Marcus, M. / Santorini, B. / Marcinkiewicz, B. (1993). "Building a Large Annotated Corpus of English: The Penn Treebank", *Computational Linguistics*, 19(2), 313–330.

Microsoft. (2009). Retrieved February 24, 2009, from http://www.microsoft.com/

Mihalcea, R. F. (2002). “Diacritics restoration: Learning from letters versus learning from words” in *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics.* Berlin: Springer Verlag, 339–348.

Ratnaparkhi, A. (1996). “A Maximum Entropy Model for Part-of-Speech Tagging” in *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* Somerset: EMNLP, 133–142.

TshwaneDJe HLT. (2009). Retrieved February 24, 2009, from http://tshwanedje.com/

van den Bosch, A. / Daelemans, W. (1999). “Memory-based morphological analysis” in *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. Maryland: ACL, 285–292.

van den Bosch, A. / Marsi, E. / Soudi, A. (2007). “Memory-based morphological analysis and part-of-speech tagging of Arabic” in *Arabic computational morphology: Knowledge-based and empirical methods*. Berlin: Springer Verlag, 203–219.

Yarowsky, D. (1994). “A comparison of corpus-based techniques for restoring accents in Spanish and French text” in *Proceedings of the Second Annual Workshop on Very Large Corpora*. Kyoto: COLING, 19–32.

Wagacha, P. / De Pauw, G. / Githinji, P. (2006) “A grapheme-based approach for accent restoration in Gĩkũyũ” in *Proceedings of the Fifth International Conference on Language Resources and Evaluation*. Genoa: ELRA, 1937–1940.