Applying computing innovations to bilingual corpus analysis

Diana Carter,[1,2] Mirjam Broersma[3,4] and Kevin Donnelly[2]
[1]University of British Columbia / [2]Centre for Research on Bilingualism, Bangor University / [3]Centre for Language Studies, Radboud University / [4]Max Planck Institute for Psycholinguistics

With current innovations in corpus analysis, it is now possible to extract and analyze large amounts of monolingual and bilingual data in minutes, as opposed to the numerous hours previously needed to manually analyze a much smaller quantum of data. In this chapter, we review innovative techniques in bilingual corpus building and analysis, which include the use of automated glossing to allow the extraction of data that can then be statistically analyzed using mixed-effects models. We discuss the application of these techniques, among others, and provide examples from three bilingual corpora. We end by suggesting how researchers may benefit from the increasingly powerful computing capability that is now available.

Keywords: codeswitching, bilingual corpus, autoglossing, automated clause-splitting

1. Introduction

This chapter explores how modern, open-source tools can be combined to create powerful ways of extracting data from corpora in response to research questions. We do this by presenting in detail the methods we have used in an ongoing study (Carter, Broersma, Donnelly, & Konopka, 2015) to analyze data from bilingual Spanish-English, Spanish-Welsh, and Welsh-English corpora. Note that we will not present any detailed research findings here, but we use this study to illustrate how new techniques can facilitate the analysis of large (bilingual) corpora. In Section 2, we first explain the motivation behind the study, defining some terms where necessary, and the research questions we intended to answer. We then describe the corpus itself (Section 3), going on to look at how it was glossed

DOI 10.1075/sibil.51.11car

automatically using a new suite of tagging tools designed specifically for multilingual corpora (Section 4). Next we show how the raw data was used to generate further data relevant to the analysis (Section 5). We then describe the data analysis itself, concentrating particularly on the statistical technique used (mixed-effects modeling, also known as multilevel or hierarchical modeling). We also include a brief summary of the findings we generated by using these techniques (Section 6). In Section 7 we provide some tips and tricks for bilingual corpus analysis. Finally, we draw conclusions based on our experience regarding the ways that computing innovations may be used to assist in corpus analysis (Section 8).

2. Triggered codeswitching

As an illustration of the novel techniques that we propose, we describe our ongoing study into bilingual speech in three large corpora. The main motivation behind that study and the development of our corpus tools is the triggering hypothesis advanced by Clyne (1967, 2003), which suggests that cognates facilitate codeswitching. In our study we define cognates (or *trigger words*) as words that overlap in form and meaning in both languages of the bilingual. Codeswitching occurs when a speaker switches between two or more languages, either within a clause or between clauses (Myers-Scotton, 2002). The triggering effect of cognates on codeswitching is the result of the selection of the cognate from the mental lexicon (Broersma & de Bot, 2006). The mental lexicon of a bilingual is organized into language subsets (Paradis, 2004), and the occurrence of a cognate (which is part of the subsets of both languages) can cause the activation of the other language subset at the lexical level (Broersma & de Bot, 2006). This increases the chance that a word from the other language will be unconsciously selected as long as the speaker is in a situation where they feel comfortable to codeswitch (i.e. a bilingual mode). Therefore, the selection of a cognate increases the chance of codeswitching.

Our study was intended to look at this phenomenon in more detail, focusing on a series of research questions such as the following:

1. What characteristics of cognates affect the extent to which they can facilitate codeswitching?
2. How do trigger words belonging to different word classes affect codeswitching?
3. How does the number of trigger words affect codeswitching?

For the purposes of the current chapter, these research questions and the results of the study are not as relevant as the method and tools we used for data processing

and analysis; however, it is important to outline the original motivation behind the development of the methodology.[1]

In order to answer our research questions, we needed to analyze large corpora of spontaneous bilingual speech. Three bilingual corpora are included: the Spanish-Welsh Corpus from Patagonia, Argentina; the Spanish-English Corpus from Miami, Florida; and the Welsh-English *Siarad*[2] Corpus from Wales (c.f. bangortalk.org.uk to access the corpora). So far, we have mostly focused on the *Siarad* corpus (Carter et al. 2015) and we will further assess the data from the Spanish-Welsh and Spanish-English corpora in future research. In this chapter we will draw on examples from all three corpora to demonstrate that the methods and tools we have developed may be applied to a variety of corpora and language-pairs.

Previous studies that involved the analysis of data from bilingual corpora (e.g. Broersma, 2009; Broersma & de Bot, 2006; Broersma, Insurin, Bultena, & de Bot, 2009; Carter, Deuchar, Davies, & Parafita Couto, 2011; Davies & Deuchar, 2010; Duran Eppler, 2010; Fernández Fuertes, Liceras, Pérez-Tattam, Martínez, Alba de la Fuente, & Carter, 2006; Herring, Deuchar, Parafita Couto, & Moro Quintanilla, 2010) used manual analysis, which is extremely time-consuming. In the Broersma (2009) study, for example, the analysis of triggered codeswitching in a 2800-word Dutch-English corpus required 250 hours and a team of five people. Clearly, in order to examine a 450,000-word corpus, we needed to find more efficient, preferably automated, methods of analysis.

3. The Miami, Patagonia, and *Siarad* corpora

In the following section, we describe the method that was used to recruit participants and transcribe the recorded bilingual speech for the Spanish-English Miami corpus, the Spanish-Welsh Patagonia corpus, and for the Welsh-English *Siarad* Corpus (Deuchar, Davies & Donnelly, 2016). A similar method was followed for all three corpora.

The bilingual speech data from the *Siarad* Corpus was collected over a period of two years in North Wales by a team of bilingual researchers at the ESRC Centre for Research on Bilingualism at Bangor University (Deuchar, Davies, & Donnelly, 2016; Deuchar, Davies, Herring, Parafita Couto, & Carter, 2014). The corpus consists of almost 450,000 words and contains speech from 151 speakers in 69 conversations totaling 40 hours. The Miami corpus contains over 250

1. For a detailed account of the research questions, variables and results from the data analysis, please see Carter et al. 2015.

2. The word *siarad* means "speak, talk" in Welsh.

000 words from 84 speakers and was recorded over a period of three months by a small team of bilingual researchers from the ESRC Centre. There are 21 hours of recorded speech from 56 conversations. The Patagonia corpus has nearly 200 000 words from 94 speakers in 56 conversations. This corpus was collected by two multilingual researchers over two months. All three corpora are available under an open license from the BangorTalk website and have been utilized in several codeswitching studies (see Davies & Deuchar, 2010; Carter et al., 2011; Herring et al., 2010; among others).[3]

3.1 Participants

For all three corpora the speakers were recruited through a variety of means, including newspaper advertisements and the 'friend of a friend' approach (Milroy, 1987). They were recorded in groups of two or three using digital recorders. The participants could choose their own conversation partner and were free to discuss any topics of their choice. The investigator was not present during the recordings in order to minimize the Observer's Paradox (Labov, 1972). Recordings lasted between 19 and 64 minutes, with the mean length being 35 minutes. Participants were also required to complete a background information questionnaire that consisted of questions about their age, gender, nationality, education, profession, social networks, age of language acquisition for all languages of study (Welsh and either Spanish or English), and general attitudes about codeswitching.

3.2 Transcription

The digital audio recordings were transcribed using the CHAT system in the Computerized Language Analysis (CLAN[4]) program (MacWhinney, 2000). In each transcript, the data was organized into different tiers of information: transcribed audio per speaker with a link to the corresponding segment in the audio file, a morpheme by morpheme gloss (c.f. section 4), and a translation of all Welsh or Spanish words into English. Words in the most frequent language in the transcript were left untagged while words in the least frequent language were tagged in accordance with the 3-letter abbreviations of ISO-639-3[5] : @s:eng for English, @s:cym for Welsh, and @s:spa for Spanish. Words occurring in the dictionaries of both languages (allowing for phonetic and orthographical variance) were tagged

3. <http://bangortalk.org.uk>

4. <http://childes.psy.cmu.edu/clan>

5. <http://www-01.sil.org/iso639-3/codes.asp>

as indeterminate (as @s:cym&eng, with the language tags in alphabetical order of the abbreviation).

One advantage of the language tagging is that it facilitates the identification of cognates and codeswitching in the corpus. Examples (1a) and (1b) below illustrate the different tiers as well as the language tags.

(1) a. ***ALN:** ond dw i ddim actually@s:eng isio mynd i wrando ar y stuff@s:cym&eng .
%eng: but I don't actually want to go and listen to the stuff.
[stammers4: 203]

b. ***LEI:** rhwng mynd â empanadas@s:spa a mynd â tarta@s:spa.
%eng: between taking empanadas and taking cake.
[patagonia25: 17]

In (1a) we have a codeswitch to English (*actually*) within a Welsh clause. Further, the sentence contains the word *stuff*, tagged as indeterminate (@s:*cym&eng)* because it occurs in both the Welsh and English dictionaries. In (1b) there are two codeswitches to Spanish (*empanadas; tarta*) within a Welsh clause.

4. Automatic glossing

The automatic glossing of text involves the separation of text into words, the look-up of each word in a dictionary that gives a list of possible lemmas and parts-of-speech (POS) for that word, and the selection of the correct lemma and POS for that word in its current context. The benefits of computer glossing of transcripts are already recognized by the CLAN project, which provides a POS tagging system called MOR (MacWhinney, 2009). However, MOR only handles 11 large (> 5 million speakers) languages, and post-tagging disambiguation (using the POST program) is only available for four languages. A separate pass over the file is required in order to tag each language, and this sometimes does not work well for short codeswitches. The MOR dictionary is also segmented by POS (i.e. one file for adjectives, one for verbs, etc.), which makes it difficult to directly re-use the contents in other contexts (e.g. spell-checkers, machine translation). The re-use of such material is especially important with minority languages, where resources may be limited.

Therefore, instead of building MOR and POST program modules to handle Welsh within CLAN, the ESRC Bilingualism Centre created a suite of tools called the Bangor Autoglosser. The Autoglosser (a) can handle multi-lingual texts in one pass, (b) uses existing free or open-source resources for lexical data and dis-

ambiguation, and (c) simplifies portability and repurposing by using "standard" database and scripting software (Donnelly & Deuchar, 2011a, 2011b).

Each language to be glossed with the Autoglosser requires a dictionary listing information about the words that may be encountered in that language. The dictionaries are based on lexical data under an open license (*Eurfa* for Welsh, Kevin Atkinson's *Moby*[6] list for English, and the *Apertium* rule-based machine translation dictionary[7] for Spanish) that have been reworked to improve consistency.

The dictionary data is held in a PostgreSQL(postgresql.org) database table (see Matthew & Stones, 2005). Table 1 shows the layout for Spanish. The layout will be familiar to any researcher who has compiled a glossary, making it easy to edit or add items, either directly to the database table, or to an exported spreadsheet version. The main benefit of this simple word-based approach is that it is possible to input any wordlist into the Autoglosser and receive output immediately. This means that it is easy to add another language.

Table 1. Spanish dictionary layout.

surface	*lemma*	*enlemma**	*pos*	*gender*	*number*	*tense*
perro	perro	dog	n	m	sg	
canciones	canción	song	n	f	pl	
empezar	empezar	start	v			infin
empieza	empezar	start	v		3s	pres
empieza	empezar	start	v		2s	imper
rojo	rojo	red	adj	m	sg	
rojas	rojo	red	adj	f	pl	
por	por	for	prep			

* *enlemma* is the English lemma for the word, and *pos* is the part-of-speech.

The autoglossing process begins by validating the CHAT transcripts using CLAN tools such as CHECK to reduce import issues arising from transcription errors[8] and importing each line of the transcript into a database table. This utterances table aligns each speaker tier with the other associated tiers, such as the translation or a previous manual glossing (if present). Figure 1 shows the database record for example (1a). The record is segmented to allow for the width of this page.

6. <http://wordlist.sourceforge.net>

7. <http://apertium.org>

8. Particularly common errors include using a space instead of a tab after the speaker ID and forgetting to insert a space before punctuation at the end of the speaker tier.

utterance_id	filename	speaker	surface
203	stammers4	ALN	ond # dw i (dd)im actu(ally)@s:eng [?] isio mynd i wrando ar y stuff@s:cym&eng .

	eng	com	comment	durbegin	durend	duration	precode
'	but I don't actually want to go and listen to the stuff.	*NULL*	*NULL*	447979	451009	3030	

Figure 1. Example (1a) in the utterances table.

The contents of the speaker tier in the *surface* field are then imported into a words table. CHAT markup is discarded and the remaining text is tokenized, as shown in Figure 2.

location	surface	langid
18	rhwng	cym
19	mynd	cym
20	â	cym
21	empanadas	spa
22	a	cym
23	mynd	cym
24	â	cym
26	tarta	spa

Figure 2. Example (1b) tokenized in the words table.

As seen in Figure 2, the language tags for each word are stripped off and retained in a separate field (*langid*) and they will be used to decide which dictionary should be used to look up the word for glossing. To simplify dictionary maintenance, the lookup process also carries out some basic segmentation of the word. For Spanish, clitic pronouns are removed (i.e. *ponerle, déjanos*). For Welsh, mutation[9] is removed (i.e. *gath* > *cath, phlant* > *plant*). For English, elisions and regular verb-endings are removed (i.e. *gonna, I'll, walking*).

The dictionary lookup gathers all matching entries for each word, and writes them out to a file (in the format required by a constraint grammar parser). Next, constraint grammar (Karlsson, 1990; Karlsson, Voutilainene, Heikkilä, & Anttila, 1995) is used to compile context-dependent rules into a grammar that selects the most appropriate tag for words in running text.[10]

The following extract gives lookup output for part of an utterance in the Miami corpus:

9. In Welsh, some word-initial consonants change ('mutate') to reflect morphological and syntactic relationships between the words of the utterance.

10. The parser used in the Autoglosser is *vislcg3*, developed by Eckhard Bick and Tino Didriksen; <http://beta.visl.sdu.dk/cg3.html>

(2) "la gente que se va a poner en foreclosure@s:eng" meaning *the people who are going to end up in foreclosure*:

```
"<la>"
      "la"  {303,14} [es] n m sg :la:
      "la"  {303,14} [es] det.def f sg :the:
      "la"  {303,14} [es] pron.obj f 3s :her:
"<gente>"
      "gente"  {303,15} [es] n f sg :people:
"<que>"
      "que"  {303,16} [es] conj :than:
      "que"  {303,16} [es] conj :that:
"<se>"
      "se"  {303,17} [es] pron.indir mf 3sp :to_him:
      "se"  {303,17} [es] pron.refl mf 3sp :self:
      "ser"  {303,17} [es] v 2p imper preclitic :be:
"<va>"
      "ir"  {303,18} [es] v 3s pres :go:
"<a>"
      "a"  {303,19} [es] prep :to:
"<poner>"
      "poner"  {303,20} [es] v infin :put:
"<en>"
      "en"  {303,21} [es] prep :in:
"<foreclosure>"
      "foreclosure"  {303,22} [en] n sg :foreclosure:
[zeledon5: 303]
```

There are multiple options shown for several words, but these can be disambiguated by having the constraint grammar parser apply grammar rules to this output. The grammar rules are written by hand, and use a simple syntax that reflects the researcher's linguistic awareness of the language. In the extract given above, for instance, *la* is ambiguous between the musical note, the feminine definite article, and the feminine object pronoun.

The correct option can be chosen here by means of the rule:

(3) select ([es] det.def f) if (1 ([es] n f) or ([es] adj f) or ([es] ord f) or ([en] adj) or ([en] n));

which states that the feminine definite article should be chosen if the following word is a Spanish feminine noun, adjective or ordinal, or an English noun or adjective (the latter is to handle codeswitching contexts). The correct option for *se* can be chosen by means of the rule:

(4) select ([es] pron.refl) if (1 (v 3s) or (v 3p));

which states that the reflexive pronoun should be chosen if it is followed by a verb in the third person singular or plural.

The application of the grammar rules results in a new file where all the words have been disambiguated:

```
“<la>”
      “la” {303,14} [es] det.def f sg :the:
“<gente>”
      “gente” {303,15} [es] n f sg :people:
“<que>”
      “que” {303,16} [es] pron.rel :that:
“<se>”
      “se” {303,17} [es] pron.refl mf 3sp :self:
“<va>”
      “ir” {303,18} [es] v 3s pres :go:
“<a>”
      “a” {303,19} [es] prep :to:
“<poner>”
      “poner” {303,20} [es] v infin :put:
“<en>”
      “en” {303,21} [es] prep :in:
“<foreclosure>”
      “foreclosure” {303,22} [en] n sg :foreclosure:
                        [zeledon5: 303]
```

The disambiguated words are then stored in the words table as a gloss, as shown in Figure 3:

location	surface	auto	langid
14	la	the.DET.DEF.F.SG	spa
15	gente	people.N.F.SG	spa
16	que	that.PRON.REL	spa
17	se	self.PRON.REFL.MF.3SP	spa
18	va	go.V.3S.PRES	spa
19	a	to.PREP	spa
20	poner	put.V.INFIN	spa
21	en	in.PREP	spa
22	foreclosure	foreclosure.N.SG	eng

Figure 3. Disambiguated words in the words table.

Finally, the CHAT file is written out of the database, inserting a new autogloss tier generated from the glossed words, as shown in Example (5):

(5) *ISA: la gente que se va a poner en foreclosure@s:eng.
%aut: the.DET.DEF.F.SG people.N.F.SG that.PRON.REL self.PRON.REFL.
MF.3SP go.V.3S.PRES to.PREP put.V.INFIN in.PREP foreclosure.N.SG
%eng: the people who are going to end up in foreclosure.
[zeledon5: 303]

The Autoglosser produces glossed text at a rate of around 1000 words/minute on a typical desktop PC, which means a transcription of a half-hour conversation can be glossed in around 6 minutes. The entire *Siarad* corpus was glossed in approximately 8.5 hours. Based on a series of manual spot-checks, we determined that the accuracy ranges from 97-99%, depending on the language.

Apart from the generated gloss, the fact that the contents of the transcribed conversation are now available in a database means that particular words or attributes of the text can be easily accessed. Although similar queries can in many cases be made using CLAN's dedicated interface, the general-purpose database language used here (SQL) is more versatile, and can of course be used in other contexts, whereas a dedicated corpus interface can only be used with that specific application.

Using a high-level language like PHP or Python to manipulate the database gives a powerful tool to begin analyzing the corpus data, one that can be customized on an ad hoc basis to handle changing research questions. We return to this in Section 7.

5. Data preparation

In order to answer specific research questions and test multiple variables, it is essential to prepare the data for statistical analysis. The data preparation consisted of several steps which we outline below. These steps may be applied to any corpus that requires a clause-based analysis.

To begin, we selected conversations from the corpus with only two speakers in order to keep the statistical analysis manageable by keeping the number of levels of the random variable *speaker* constant. This meant that we looked at only 52 of the 69 conversations in the corpus, dealing with 105 speakers out of the total of 159 listed.

Next, we filtered out all utterances that only contained interactional markers instead of words with semantic meaning or a syntactic function. These "interactional markers" include items such as *uhhuh*, *mmhm*, *oh*, *OK*, *aha*, etc, and constitute "noise" in the material.

After filtering for interactional markers, it was necessary to divide all of the complex clauses into simple clauses so that we could follow the Matrix Language Frame model (Myers-Scotton, 2002) in order to determine a base or matrix language for each clause, and therefore also determine if there was a codeswitch present. According to Myers-Scotton (2002), there is a matrix language that provides the morphosyntactic frame for the clause and an embedded language that contains inserted material, mostly content morphemes (p. 2). The Matrix Language Frame model posits that codeswitched items are part of the morphosyntactic frame whose language can be defined by the language of the finite verb in each clause. It follows that enumerating codeswitches between adjacent clauses depends on (1) segmenting an utterance into its constituent clauses, and (2) identifying the language of the finite verb in each of those clauses. Thus, we first had to divide all complex clauses into simple clauses before we were able to identify the language of the finite verb.

In previous studies using the *Siarad* corpus, clause-splitting was done manually (Carter et al., 2011; Davies & Deuchar, 2010) and involved several weeks of manual work by multiple researchers. For the triggered codeswitching study, an ad hoc approach was used. Since no Welsh parser currently exists (the lack of such tools is a limitation common to many minority languages; see Streiter, Scannell, & Stuflesser, 2006), a marker was added in the words table against every finite verb, and then moved where necessary. In the following examples (6) and (7), finite verbs are underlined, the words onto which the marker was moved are in bold, and clause-splits are marked with /.

(6) ond <u>mae</u> yna rei / <u>sydd</u> wedi / dw i meddwl / <u>bod</u> nhw wneud drwg mawr i ni felly
'*but there are some / who have / I think / that they are doing us a lot of harm really*' [fusser10: 499]

(7) <u>dw</u> i yn cofio / <u>o'n</u> i yn gweithio ar y nos / **pan** <u>o'n</u> i yn gweithio yn Beaumaris
'I remember / I was working nights / when I was working in Beaumaris' [davies10: 1331]

At this point we were ready to determine the matrix language of each clause. The matrix language was assigned to each clause by detecting the language of the finite verb within that clause. The language tagging of the words in the transcript allowed this to be done automatically. Once a matrix language was assigned, external codeswitches (i.e. switches extending over the clause boundary) were detected by comparing the matrix language of the current clause with that of the previous one, and counting a switch where they differed. In cases where either clause did not contain a finite verb, or consisted solely of indeterminate words, the comparison

was invalid, and therefore that clause was ignored. Internal codeswitches (i.e. switches within the clause boundary [Carter et al., 2011; Myers-Scotton, 2002]) were detected by looking at the language of each word in the clause, and counting a switch where the clause contained more than one language. Other data was then generated to characterize the clauses themselves:

a. location of the clause in its speaker-turn;
b. length of the clause in words;
c. length of the speaker-turn for that clause;
d. whether the clause contained cognates, and if so, how many;
e. location of the cognates in the clause;
f. sequence of each cognate in relation to other cognates in that clause;
g. type (part of speech) of each cognate;
h. the length (in letters) of each cognate;
i. the number of non-cognate words in the clause;
j. the language of the clause, ignoring cognates – either monolingual (Welsh or English), or bilingual.

Finally, summary numerical data was generated for conversation information external to the clause:

k. total number of words;
l. total number of clauses;
m. total number of cognates;
n. total number of codeswitches;
o. total number of words by that speaker;
p. total number of clauses by that speaker;
q. total number of cognates by that speaker;
r. total number of codeswitches by that speaker.

The enriched data for the almost 65,000 clauses in the database table were then exported to a comma-separated value file, ready for import into software for statistical computing, such as R (www.r-project.org – see Baayen, 2008; Gries, 2009; Gries, 2013).

It is clear that preparing the data manually for a corpus of this size would have been a daunting task. Although it took a significant amount of experimentation to arrive at the optimum set of attributes required, and test the output to ensure that it met our needs, once this was done, the entire corpus was processed in a couple of hours. There is inevitably an element of error involved in any automatic process, but we believe that this is an acceptable price to pay for being able for the first time to handle large quantities of corpus data without excessive time spent on data preparation.

6. Data analysis and results

6.1 Data analysis

In our study, we used *mixed-effects modeling* (Gelman & Hill, 2006; Zuur, Savelieve, & Ieno, 2012), a parametric method of data-analysis that has become increasingly common in many scientific disciplines, including the field of psycholinguistics (see Jaeger, 2008; Quené & van den Bergh, 2008). Although mixed-effects models are not yet commonplace in corpus linguistics (though see Baayen, 2008, Chapter 7), we argue, with Tagliamonte and Baayen (2012), that they provide an optimal tool for the statistical analysis of extensive language corpora. In our study, more specifically, we used *logit* mixed models (see Jaeger, 2008), which allow for the analysis of nominal data (in this case, presence or absence of code-switches).

Many corpus studies have made use of non-parametric tests, like chi-squared. Such tests have certain advantages: they are appropriate for the use with nominal data (e.g., presence or absence of a code-switch) or ordinal data (e.g., low, intermediate, or high proficiency), they make few assumptions (for example about the distribution of the data), and they are easy to use. There are, however, some important drawbacks: non-parametric tests have much lower statistical power than parametric tests, and they do not allow for the simultaneous assessment of larger numbers of variables (in sometimes complicated constellations) like parametric tests do. With the advent of large computerized corpora, containing huge numbers of observations, the reasons for using non-parametric tests are dwindling. Within the range of parametric tests available, we argue for the use of mixed-effect models, as they have important advantages over other parametric tests.

First, mixed-effect models are particularly suitable for use with unbalanced data sets. Unbalanced data pose serious problems for most parametric methods. In corpus studies, however, unbalanced data sets are unavoidable. In our study, the number of observations in e.g. Spanish and Welsh will be different, the number of observations will differ for every speaker in the corpus, some speakers will produce many sentences with grammatical construction A and few with B while others do the opposite, etc. Unequal numbers of observations at these different levels are a great challenge for statistical models. Analyses of Variance (ANOVAs) and t-tests make use of *F*- and *t*-distributions, which enable an exact calculation of significance levels for perfectly-balanced datasets, and are reasonably capable of estimating significance for slightly unbalanced datasets (such as experimental data with a few missing values). They are not suitable, however, for the analysis of strongly unbalanced datasets like corpus data. Mixed effect models, on the other hand, are particularly well suited for the analysis of such corpus data with unequal numbers of observations.

Another major advantage of mixed-effects models over other parametric methods for the analysis of corpus data is that these models can accurately deal with so-called random effects. In most corpora, data from several speakers are combined. These speakers form a subset of the population under study. In other words, they form a sample from all the possible speakers that the researchers could have selected for their study. Researchers are not interested in drawing conclusions about the characteristics of the speech of the specific speakers in the corpus; they are not aiming to show that those 3, 10, or 100 speakers display a particular pattern in (for example) their code-switching. Rather, they want to be able to show that speakers from "the Welsh-English population in Bangor" or "the Welsh-Spanish population in Patagonia" display a particular pattern in their codeswitching. In order to be able to generalize from the participants in the corpus to the population at large, the statistical model should treat speakers as a random variable.

Similarly, speaker dyads (or triplets, etc.) may need to be treated as random variables. In our study, all speakers carried out conversations in dyads, with each speaker participating in only one conversation. Like the speakers included in the corpus, the combinations of speakers into dyads formed a subset of all the possible combinations that could have been formed. Thus, speaker dyads had to be treated as a random variable too. As each dyad consisted of two (unique) speakers, the random variable of speakers was nested within the random variable of dyads. In many studies, specific items are selected from a larger set (e.g., a subset is taken from all the words in a corpus, or from all the words in a certain language). In those cases, items need to be treated as random variables. In our study, on the other hand, we used all items from the selected speakers in the corpus. As we did not select items, items were not treated as random variables.

Failure to treat random variables as such leads to serious statistical problems, which strongly reduce the validity of the conclusions. If random variables are not specified, the model will treat repeated observations from the same speaker, dyad, etc. as if each observation came from a different speaker, dyad, etc. This leads, among other things, to deficiencies in statistical power and in dealing with heteroskedasticity (that is, the statistical model assumes that all speakers or dyads show the same normally-distributed and uncorrelated variance, which does not vary with the independent variables under analysis). As a result, the outcomes of such analyses are unreliable.

Regular regression analysis, including ANOVA, does not provide satisfactory ways of dealing with random effects. A common method of dealing with random effects in experimental work is aggregating (i.e., averaging) the data, first over participants and then over items. This method, however, has long been known to be flawed (Clark, 1973). In corpus linguistics, there are no methods in common use that take random effects into account at all. In mixed-effects models, on the other

hand, several random variables (in our study: speakers and speaker dyads) can be simultaneously included in the statistical model. The variability for each of the predictors (i.e., independent variables) under study can be tested for each of the random variables. This is important because some combinations of predictors and random variables might require inclusion in the model, and others might not. For instance, the effect of a certain predictor might vary across speakers but not across speaker dyads; similarly, the effect of one predictor might vary across speakers while the effect of another predictor does not. *Random slopes* for each specific combination of predictors and random variables can be added to the model to capture the variability of the required combination. Therefore, mixed-effects models provide a much more sensitive test of the effect of the predictors on the dependent variable than other statistical methods.

6.2 Results

The use of mixed-effects modeling for our statistical analysis enabled us to investigate a series of research questions based on Clyne's *triggering hypothesis* (1967, 2003). Although a detailed account of the results is beyond the scope of this paper, we offer a summary of the main findings in order to demonstrate the validity of our methodology.

We tested the effect of the presence of trigger words on codeswitching and found that there were more codeswitches in clauses containing one or more trigger words than in clauses without. In other words, the presence of cognates led to a higher number of codeswitches. We also investigated the effect of the number of trigger words present in clauses with codeswitches. The number of trigger words only affected clause-external switches, and not clause-internal switches, with a larger number of trigger words leading to a higher proportion of switches. Our results also showed that individual speakers who produced more trigger words produced more codeswitches overall. In addition to the presence and number of trigger words, we assessed how cognates belonging to different word classes affected codeswitching. We found that all categories of trigger words led to a significant increase in clause-internal codeswitches (eg. noun, proper noun, verb, modifier); however, only nouns led to a significant increase in codeswitches in external clauses.

7. Tips and tricks for processing corpus data

A key point that should be considered with regards to corpora is that they are only as good as the access tools that are available to allow their contents to be inspected

and analyzed. If corpora are intended to capture real-world speech patterns, it follows that some patterns may be of very low frequency. For example, there appear to be only ten instances in *Siarad* where two clauses in different languages follow each other.

Many corpora provide access via custom interfaces, and some programs have also been developed which are intended to facilitate access to a variety of corpora, e.g. Corpus Workbench[11] or Corpus Presenter.[12] CLAN provides a suite of analysis programs to allow transcriptions in that format to be inspected (e.g. FREQ to look at word frequencies).

What we would like to emphasize here is that, although using these specialist tools may offer some benefits, there are many more advantages for researchers in trying to build up their own toolset, which will be "transferable" across multiple projects, multiple corpora and multiple languages. In other words, handling corpus text should not be considered a specialist "computer science" function, but part of any researcher's toolkit. For example, the CLAN command below picks out some text (*mytext*) from the *%mor* tier:

(8) `gem +t%mor +smytext +d mychat.cha`

This command is concise, and easy to understand once the researcher learns the various switches used (*+t, +s, +d)*. However, this learning experience has to be repeated for any other corpus package being used. For example, what switches/conventions do Corpus Presenter or corpus Workbench use for the same query. If the file is in a database where each tier is in its own field (column), as with the utterances table in the Autoglosser (Figure 1), the same information can be returned by a database query like:

(9) `select mor from mychat where mor ~ 'mytext'`

This is easier to read than the CLAN command but, more importantly, this is an example of "learn once, use often". This query can be used on *any* corpus data stored in a database. Researchers can therefore receive a better return on the investment of time spent learning how to use tools.

There is a learning curve involved in moving from "Excel and my standard corpus software" to a wider range of customizable software, but thankfully it is not necessary to learn everything at once. It is feasible to start with one area and move forward incrementally. The following paragraphs give some practical suggestions for this:

11. <http://cwb.sourceforge.net>

12. <http://www.uni-due.de/CP>

1. Use open tools and data wherever feasible. Open tools mean that there is less chance of your project being "ambushed" by changes in licensing terms. Open data means that you are free to expand and handle the data in whatever way suits you best, and you can usually access the entirety of the data as opposed to accessing a subset through a closed interface.
2. If possible, make your own data, and any tools you develop, available under an open license. For instance, the Bangor corpora and the Bangor Autoglosser are licensed under the Free Software Foundation's General Public License (GPL).[13] This aids "reproducible science," which is attracting increasing attention in the "hard" sciences.[14]
3. Do not maintain data in formats that require a graphical user interface (GUI) to view. Use a "lowest common denominator" format such as text (.txt) or comma-separated values (.csv). These can be imported by most graphical programs for easy viewing, but they also allow your data to be handled more easily by scripts (see item 7), or stored in a version-control system.
4. Instead of using built-in programs, try writing your own. Use a scripting language (PHP, Python) instead of a compiled language (C++, Java). Scripting (interpreted) languages run commands immediately without conversion to machine-code, while compiled languages compile the commands to machine-code first and then run them. Scripting languages are therefore usually slower, but easier to make incremental changes with; compiled languages are usually faster but there is the delay of compiling first. Therefore, with scripting languages you can get started more quickly and have immediate output. The Natural Language Toolkit (Bird, Klein & Loper, 2009) is a set of ready-made Python resources that may increase the efficiency of your work if you have decided to use Python as your language of choice. The point made earlier about databases also applies here — if you invest time in developing scripts for one corpus project, they can be used for other corpora or languages. Being able to transfer skills in this way means you will be more productive.
5. Transfer your data to a database. Various applications can work directly on the text files, but if you want to reconfigure the data in any way it is much easier to have it in a database. PostgreSQL (Douglas & Douglas, 2003; Matthew & Stones, 2005) offers a good balance of power and ease of use. Scripting languages usually have easy ways of manipulating databases: for instance, the following PHP snippet fetches (select) all (*) the records from *mytable,* and then

13. <http://www.gnu.org/licenses/gpl.html>

14. The Reproducibility Initiative has recently been established <https://www.scienceexchange.com/reproducibility>; and repositories such as Figshare <http://figshare.com> allow researchers to store or publish output in a way that makes it accessible to others.

does something to each of them (e.g. prints out all or part of them, manipulates them further, etc):

```
$sql=query("select * from mytable");
while ($row=pg_fetch_object($sql)
{
        <do something here>
}
```

6. Solve problems in small steps. Instead of trying to build a single monolithic script that does everything, build small scripts that do a single thing (sometimes referred to as the "Unix philosophy"), and then combine them in a "pipeline" or script. This allows incremental progress by focusing on one step at a time. As Wilson et al. (2012) note, researchers "often *can't* know what their programs should do next until the current version has produced some results" (p. 2). Because it is easy to check output at each stage, this approach also helps with testing, because only once a script is producing correct output will you be able to go on the next part. Another benefit is that revisions or improvements can be made at any stage with little impact on other stages.
7. Use the best tool available for statistical analysis. R, an open version of the S language (Becker & Chambers, 1984), is now used widely in a variety of fields, including corpus linguistics (Baayen, 2008; Gries, 2009). A number of books exist which allow you to learn statistics through the medium of R (see Crawley, 2005; Field, Miles, & Field, 2012; Gries, 2013).

8. Conclusions

In summary, by developing innovative computational tools and methods for bilingual corpora such as the Autoglosser and the clause-splitter, we were able to prepare and analyze our data in a quick and efficient manner. A manual analysis of a corpus consisting of almost 450 000 words would have taken thousands of hours and a much larger team of researchers than we had available. The use of mixed modeling allowed us to run a simultaneous assessment of a large number of variables, while taking random effects into account (see Carter et al., 2015). Finally, we have discussed how researchers without sophisticated training in computer science can use computing innovations to extract and analyze large quantities of monolingual and bilingual corpus data, and we have shared some tips and tricks. We hope that the methods discussed in this paper might contribute to the study of what we believe is the most fascinating topic in the world: bilingual language use in all its facets.

Acknowledgements

This research was supported by a Small Research Grant from the British Academy to the first and second author. We acknowledge the Bangor Bilingualism Centre and the Max Planck Institute for Psycholinguistics for their generous support. We would like to thank Edward Carter and Constance Crompton for their feedback and constructive comments. We would also like to thank Margaret Deuchar for her support and encouragement throughout many stages of the project. Finally we thank the audience at the American Association of Corpus Linguistics Conference for their inquiries and suggestions.

References

Baayen, R. (2008). *Analyzing Linguistic Data. A Practical Introduction to Statistics Using R*. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511801686

Baayen, R., Davidson, D., & Bates, D. (2008). Mixed-effects modeling with crossed random effects for subjects and items. *Journal of Memory and Language*, 59, 390–412. doi: 10.1016/j.jml.2007.12.005

Becker, R., & Chambers, J. (1984). *S: An Interactive Environment for Data Analysis and Graphics*. Ithaca, NY: CRC Press.

Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.

Broersma, M. (2009). Triggered codeswitching between cognate languages. *Bilingualism: Language and Cognition*, 12, 447–462. doi: 10.1017/S1366728909990204

Broersma, M., & de Bot, K. (2006). Triggered codeswitching: A corpus-based evaluation of the original triggering hypothesis and a new alternative. *Bilingualism: Language and Cognition*, 9, 1–13. doi: 10.1017/S1366728905002348

Broersma, M., Isurin, L., Bultena, S., & de Bot, K. (2009). Triggered codeswitching: Evidence from Dutch-English and Russian-English bilinguals. In L. Isurin, D. Winford, & K. de Bot (Eds.), *Multidisciplinary Approaches to Codeswitching* (pp. 103–128). Amsterdam: John Benjamins. doi: 10.1075/sibil.41.08bro

Carter, D., Broersma, M., Donnelly, K., & Konopka, A. (2015). How cognates affect codeswitching: A large-scale study of Welsh-English bilinguals. Ms. in Preparation.

Carter, D., Deuchar, M., Davies, P., & Parafita Couto, M. C. (2011). A systematic comparison of factors affecting the choice of matrix language in three bilingual communities. *Journal of Language Contact*, 4, 153–183. doi: 10.1163/187740911X592808

Clark, H. (1973). The language-as-fixed-effect fallacy: A critique of language statistics in psychological research. *Journal of Verbal Learning & Verbal Behavior*, 12, 335–359. doi: 10.1016/S0022-5371(73)80014-3

Clyne, M. (1967). *Transference and Triggering: Observations on the Language Assimilation of Postwar German-speaking Migrants in Australia*. The Hague: Martinus Nijhoff.

Clyne, M. (2003). *Dynamics of Language Contact: English and Immigrant Languages*. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511606526

Crawley, M. (2005). *Statistics: An Introduction Using R*. Chichester: Wiley & Sons. doi: 10.1002/9781119941750

Davies, P., & Deuchar, M. (2010). Using the Matrix Language Frame model to measure the extent of word order convergence in Welsh-English bilingual speech. In A. Breitbarth, C. Lucas, S. Watts & D. Willis (Eds.), *Continuity and Change in Grammar* (pp. 77–96). Amsterdam: John Benjamins. doi: 10.1075/la.159.04dav

Deuchar, M., Davies, P., & Donnelly, K. (2016). *Building and Using the Siarad Corpus of Spoken Welsh: Bilingual Conversations in Welsh and English*. Manuscript in preparation.

Deuchar, M., Davies, P., Herring, J., Parafita Couto, M.C., & Carter, D. (2014). Building bilingual corpora. In E. Thomas & I. Mennen (Eds.), *Advances in the Study of Bilingualism* (pp.93–110). Bristol: Multilingual Matters.

Donnelly, K., & Deuchar, M. (2011a). *The Bangor Autoglosser: A Multilingual Tagger for Conversational Text*. Paper presented at Internet Technologies and Applications, 11. Wrexham, Wales.

Donnelly, K., & Deuchar, M. (2011b). Using constraint grammar in the Bangor Autoglosser to disambiguate multilingual spoken text. *Constraint Grammar Applications: Proceedings of the NODALIDA 2011 Workshop*. Riga, Latvia: NEALT Proceedings Series, Tartu.

Douglas, K., & Douglas, S. (2003). *PostgreSQL: A Comprehensive Guide to Building, Programming, and Administering PostgreSQL Databases*. Indianapolis, IN: Sams Publishing.

Duran Eppler, E. (2010). *Emigranto: The Syntax of a German/English Mixed Code*. Vienna: Braumüller.

Field, A., Miles, J., & Field, Z. (2012). *Discovering Statistics Using R*. London: Sage.

Fernández Fuertes, R., Liceras, J. M., Pérez-Tattam, R., Martínez, C., Alba de la Fuente, A., & Carter, D. (2006). *The Nature of the Pronominal System and Verbal Morphology in Bilingual Spanish/English Child Data: Linguistic Theory and Learnability Issues*. Paper presented at the Hispanic Linguistic Symposium. London: University of Western Ontario.

Gelman, A., & Hill, J. (2006). *Data Analysis Using Regression and Multilevel/hierarchical Models*. Cambridge: Cambridge University Press. doi: 10.1017/CBO9780511790942

Gries, S. (2013). *Statistics for Linguistics with R: A Practical Introduction* (2nd ed.). Berlin: Mouton de Gruyter. doi: 10.1515/9783110307474

Gries, S. (2009). *Quantitative Corpus Linguistics with R: A Practical Introduction*. London: Routledge. doi: 10.1515/9783110216042

Herring, J., Deuchar, M., Parafita Couto, M. C., & Moro Quintanilla, M. (2010). 'I saw the madre': Evaluating predictions about codeswitched determiner-noun sequences using Spanish-English and Welsh-English data. *International Journal of Bilingual Education and Bilingualism*, 13, 553–573. doi: 10.1080/13670050.2010.488286

Jaeger, T. (2008). Categorical data analysis: Away from ANOVAs (transformation or not) and towards logit mixed models. *Journal of Memory and Language*, 59, 434–446. doi: 10.1016/j.jml.2007.11.007

Karlsson, F. (1990). Constraint grammar as a framework for parsing unrestricted text. In H. Karlgren, (Ed.), *Proceedings of the 13th International Conference of Computational Linguistics*, 3, (pp. 168–173). Stroudsurg, PA: Association for Computational Linguistics. doi: 10.3115/991146.991176

Karlsson, F., Voutilainen, A., Juha Heikkilä, J., & Anttila A. (1995). *Constraint grammar: A language-independent system for parsing running text*. *Natural Language Processing*, 4. Berlin: Mouton de Gruyter. doi: 10.1515/9783110882629

Labov, W. (1972). Some principles of linguistic methodology. *Language and Society*, 1, 97–120. doi: 10.1017/S0047404500006576

MacWhinney, B. (2000). *The CHILDES Project: Tools for Analyzing Talk* (3rd Ed.). Mahwah, NJ: Lawrence Erlbaum Associates.

MacWhinney, B. (2009). Enriching CHILDES for morphosyntactic analysis. *Department of Psychology*. Paper 175 Enriching CHILDES for morphosyntactic analysis <http://repository.cmu.edu/psychology/175>

Matthew, N., & Stones, R. (2005). *Beginning Databases with PostgreSQL: From Novice to Professional*. New York, NY: Apress.

Milroy, L. (1987). *Language and Social Networks*. Oxford: Blackwell.

Myers-Scotton, C. (2002). *Contact Linguistics: Bilingual Encounters and Grammatical Outcomes*. Oxford; NY: Oxford University Press.

Paradis, M. (2004). *A Neurolinguistic Theory of Bilingualism*. Amsterdam: John Benjamins. doi: 10.1075/sibil.18

Quené, H., & van den Bergh, H. (2008). Examples of mixed-effects modeling with crossed random effects and with binomial data. *Journal of Memory and Language*, 59, 413–425. doi: 10.1016/j.jml.2008.02.002

Streiter, O., Scannell, K., & Stuflesser. M. (2006). Implementing NLP projects for non-central languages: Instructions for funding bodies, strategies for developers. *Machine Translation*, 20, 267–289. doi: 10.1007/s10590-007-9026-x

Tagliamonte, S., & Baayen, R. (2012). Models, forests, and trees of York English: Was/were variation as a case study for statistical practice. *Language Variation and Change*, 24, 135–178. doi: 10.1017/S0954394512000129

Wilson, G., Aruliah, D., Brown, C., Hong, N., Davis, M., Guy, R., … Wilson, P. (2012). *Best Practices for Scientific Computing*. arXiv preprint *arXiv:1210.0530*.

Zuur, A., Saveliev, A., & Ieno, E. (2012). *Zero Inflated Models and Generalized Mixed Models with R*. Scotland: Highland Statistics.