Bayesian Models of Syntactic Category Acquisition

Stella C. Frank

Doctor of Philosophy
Institute for Language, Cognition and Computation
School of Informatics
University of Edinburgh
2013

Abstract

Discovering a word's part of speech is an essential step in acquiring the grammar of a language. In this thesis we examine a variety of computational Bayesian models that use linguistic input available to children, in the form of transcribed child directed speech, to learn part of speech categories. Part of speech categories are characterised by contextual (distributional/syntactic) and word-internal (morphological) similarity. In this thesis, we assume language learners will be aware of these types of cues, and investigate exactly how they can make use of them.

Firstly, we enrich the context of a standard model (the Bayesian Hidden Markov Model) by adding sentence type to the wider distributional context. We show that children are exposed to a much more diverse set of sentence types than evident in standard corpora used for NLP tasks, and previous work suggests that they are aware of the differences between sentence type as signalled by prosody and pragmatics. Sentence type affects local context distributions, and as such can be informative when relying on local context for categorisation. Adding sentence types to the model improves performance, depending on how it is integrated into our models. We discuss how to incorporate novel features into the model structure we use in a flexible manner, and present a second model type that learns to use sentence type as a distinguishing cue only when it is informative.

Secondly, we add a model of morphological segmentation to the part of speech categorisation model, in order to model joint learning of syntactic categories and morphology. These two tasks are closely linked: categorising words into syntactic categories is aided by morphological information, and finding morphological patterns in words is aided by knowing the syntactic categories of those words. In our joint model, we find improved performance vis-a-vis single-task baselines, but the nature of the improvement depends on the morphological typology of the language being modelled. This is the first token-based joint model of unsupervised morphology and part of speech category learning of which we are aware.

Acknowledgements

My two supervisors were efficiently complementary and I am grateful to them both. Frank Keller allowed me to pursue my interests freely, while keeping the thesis on track and making sure there was a big picture. Sharon Goldwater was an influence on the thesis even before her arrival in Edinburgh, and I have been very lucky to have her input and guidance. I can only hope I have had sufficient input data to have acquired her skill at identifying and asking the key awkward questions.

My deepest thanks to Alexander Clark and Steve Renals for being on my committee and making the viva an enjoyable experience.

I was fortunate to spend some time in Zürich working with Massimiliano Ciaramita at Google. Thank you also to Katja Filippova and Keith Hall for being excellent lunch companions and friends.

To everyone in Edinburgh who provided enthusiasm and distractions, my thanks. Abby Levenberg, Tamsin Maxwell, Gregor Stewart, Sujai Kumar, and Seymour Knowles-Barley were excellent company from one post-graduate degree to the next (when will it end?). In Buccleuch Place I met the amazing Alexandra Birch, Moreno Coco, and Vera Demberg. Finally, if the Informatics Forum is to be judged by its ability to bring together researchers, it is a great success, though I suspect table tennis skill was not quite what the architects and designers were trying to optimise. Nevertheless: Teju Deoskar, Desmond Elliot, Diego Frassinelli, Eva Hasler, Kate Ho, Tom Kwiatkowski, Mike Lewis, Oier Lopez de Lacalle, Dave Matthews, Silvia Pareti, Sasa Petrović, Federico Sagati, Emily Thomforde, Wolodja Wentland: it has been wonderful, thank you.

Particular gratitude goes to Lexi and Des, for their feedback on draft chapters of this thesis and general encouragement during the end times. To Louise Milne, for not only the roof over my head. To Conrad Hughes, for his unfaltering support and companionship. Last and first to Annya Tisher, for everything that has been written over the decades.

Zletscht — mami und papi und schtrudel, danke dass ihr zuegloset haent und gholfe haent und geduldig uf nachrichte gwartet haent. Di bsuech bi eu (und euri bsuech bi mir!) sind wichtig gsi; euri perspektive und erfarige zchoere haet mi bericheret und hoffnig gae. Euri liebi zur schprache und neugir ueber maensche und di waelt haent mich daane bracht. Kiitos vielmals!

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Stella C. Frank*)

Table of Contents

Chapter 1

Introduction

In order to understand and speak the language they hear around them, human infants have to infer a complex system of linguistic representations from a noisy, incomplete sample. Over a relatively short span of time, children learn to process acoustic cues, relate these to communicative intention, and learn to communicate their own intentions. They must infer the structure of the utterances they hear, and learn to structure their own productions accordingly.

In this thesis we present computational models that aim to infer one aspect of the structure necessary for linguistic competence, namely the categorisation of words into syntactic categories such as *noun* or *verb* from data similar to that available to children. These models do so on the basis of broader understandings of the input and internal structure available to children than those used in previous models, and we show that these extensions are beneficial.

The acquisition of syntax is one of the fundamental questions of cognitive science research, with a number of methodological approaches. Collecting naturalistic data from children undergoing the language learning process provides evidence about the input they hear as well as the developmental progression of their own productions. Experimental studies elicit clues about the mental representations and processes used by children at different stages of development. Finally, computational modelling provides a way of testing the hypotheses about language development that are suggested by the empirical and experimental data. The high-level goal of this thesis is to demonstrate the necessity of modelling language acquisition using complex models that acknowledge the multifaceted and parallel nature of the acquisition process.

Computational models cannot tell us what a human learner is doing. They do, however, allow us to investigate the input and internal representations that are required to replicate human behaviour. They can make predictions about the nature and amount of input necessary to learn certain aspects of language. They can also make predictions about the relative timecourse of learning: if successful learning of one task is conditional on structure learned by a second task, then this indicates sequential learning; conversely, successful parallel learning can benefit from bootstrapped representations emerging in both tasks at once.

Constructing a computational model requires making a number of decisions about the structure of the input, specifically, which features of the input are assumed to be relevant for the task at hand. One core hypothesis of this thesis is that models of acquisition can benefit from attending to features of the input that current models have ignored. In particular, we show that features from language components outwith syntax, such as prosody and pragmatics, can provide cues for syntax acquisition.

A second core hypothesis is that computational models can benefit from integrating information from multiple tasks into a complex learning process involving multiple goals. Human learning often involves the acquisition of multiple components of a task (or multiple tasks) in parallel, rather than sequentially, and a realistic computational model should take this into account. Furthermore, there is often a productive interaction between closely related tasks, which a joint model can exploit.

These hypotheses are general to computational modelling of cognition. We apply them to the specific task of lexical syntactic category learning, a relatively well-studied problem in both the fields of language acquisition modelling as well as natural language processing. Our approach is not focussed on engineering state of the art systems. Instead, we are interested in the implications of model structure in terms of acquisition: Which changes have positive effects, and why? What does this tell us about the structure of the input, and possible constraints on the part of the learner?

1.1 Our Approach

In this thesis, we take as our starting point a Bayesian model of unsupervised part of speech tagging. This model performs syntactic category induction, that is to say, it assigns words to clusters that are assumed to correspond to lexical syntactic categories (parts of speech), and it does so without access to any information about the correct categories (it is unsupervised). Learning about syntactic categories is a key step in the process of learning syntax, since they enable generalisations over words, greatly simplifying the representations required in subsequent levels of syntactic analysis.

The clusters learned by the model are a secondary result of the principal task of part of speech tagging: assigning each *token* in the input to a category label corresponding to parts of speech. This is in contrast to the bulk of work in modelling the acquisition of syntactic categories, which focusses on assigning word *types* to clusters, based on the complete set of all occurrences of a type. Our tagging model labels a single token at a time, and thus can naturally handle syntactic category ambiguity. It also has an explicit representation of word order, or rather, the order of syntactic categories, and is thus capable of making inferences over an utterance as a whole, rather than only over individual words.

The basic model we use, the Bayesian Hidden Markov Model (BHMM, Goldwater and Griffiths (2007)), is well-known. It models the underlying structure of a sentence as a sequence of part of speech labels, and as such, is a particularly simplified view of syntax, disregarding the hierarchical structure inherent to language. However, this simplification seems appropriate for the first stages of acquisition, when learners encounter short sentences with simple syntax. Most computational models of hierarchical syntax acquisition or grammar induction assume that syntactic categories are available and induce a grammar over syntactic categories rather than lexical items (e.g., Perfors (2008); Klein and Manning (2004)). The prerequisite task of finding the syntactic categories is left to simpler models, such as the BHMM.

The Bayesian nature of the BHMM, which we describe in detail in Chapter 2, makes it straightforward to extend in a mathematically sound fashion. Our general methodology is to compare the performance of the extended models with additional structure to simpler models to examine whether the additional complexity has come at a cost or whether it has added value by improving model performance. This thesis presents a number of such extensions, which we now briefly summarise.

1.1.1 Adding New Sources of Information: Sentence Type

In Chapter 3 we add information to the model by adding a new observed variable: *sentence type*. This variable designates a sentence as being e.g., a question, declarative, or imperative. Sentence type has a regular and clearly observable effect on word order in many languages, and thus could be a useful cue in a model that relies on local contexts, that is, the order of the immediate surrounding words.

However, adding an observed variable to a Bayesian network implies adding an additional conditional variable to the conditional distributions embodied in the model.

If the variable is uninformative, additional conditionality simply fractures the model without serving any useful purpose: performance will go down. This would correspond to a learner paying attention to the wrong cue, i.e. hypothesising a causal link that is actually uninformative. Indeed, we see that in some cases, depending on how we alter the model structure, performance does in fact decrease. However, in other cases, namely when the sentence type variable is conditioning the word order distributions, we see improved performance, indicating that the additional variable carries relevant information for the part of speech tagging task.

In Chapter 4 we attempt to ameliorate the fracturing effect of adding conditional variables by adding flexibility to the model structure: the new model includes *transition groups*, additional latent variables that specify whether or not a particular distribution should be conditioned on the sentence type observation. This flexibility captures the learner's ability to select which cues to pay attention to, and to disregard uninformative cues.

1.1.2 Joint Learning: Morphology and Syntactic Categories

In Chapter 5 we add a *morphological* component to the BHMM, in order to examine the interaction between morphological segmentation and syntactic categorisation based on local surrounding context. These two tasks are closely linked in the development of syntax, occurring at approximately the same stage of language development. Both local context, in the sense of word order, and morphology serve as signals for syntactic categories and syntactic roles. As in previous chapters, we evaluate this model on typologically distinct languages. We find differing patterns of results, with morphology improving the tagging task more clearly in the language with greater morphological richness (Spanish) and local context improving morphological segmentation in the language with stricter word order (English). These results demonstrate the flexibility of our model, and argue for a general need for realistic models of category and morphology acquisition to take both tasks into account jointly, in order to be applicable across languages.

1.2 Contributions

This thesis investigates the nature of the information available to language learners. The learners in this thesis are computational models, but we strive to ensure that the information given to the models is compatible and consistent with information available to human learners.

This methodology allows us to test hypotheses about the utility of novel cues in a model of acquisition that includes sources of linguistic and non-linguistic information. Specifically, we demonstrate that representing the novel cue sentence type explicitly provides additional information beyond the limited local contexts used in previous models. We posit that sentence type is inferred on the basis of non-syntactic information such as prosody and communicative intent, cues that are available before the acquisition of syntax.

Our success in this task is an example of how a computational model can raise hypotheses about human learning: our models link prosody and word order representations in an explicit way that has, to our knowledge, never been examined in human learners. Our results show that this link is beneficial for the model, and thus indicate an avenue for future experimental research with human learners.

The joint model of morphology and syntactic category acquisition demonstrates the value of models of parallel learning of multiple tasks. Such models can not only better recreate the process of language acquisition of a single learner, who does not wait to perfect one aspect of language before beginning to learn the next, but we find that they are necessary in order to reflect language diversity. Different language typologies result in varying availability and informativeness of certain cues, such as word order or suffixing patterns, for the morphology and syntactic category acquisition tasks. A joint model is able to accommodate linguistic diversity better than single task models, confirming our hypothesis regarding the necessity of complex models of learning.

Much previous work in modelling language acquisition has acknowledged the need for complex models of full learning, but few models have gone beyond finding results for a single task. This work demonstrates both the tractability of joint learning as well as the potential benefits, in terms of modelling performance, thereof.

Chapter 2

Background

In this chapter we present the general framework used in this thesis, Bayesian modelling. We begin with a description of Bayesian models in the context of computational models of cognition, and discuss what kinds of hypotheses they can investigate. We then examine syntactic category acquisition, both in human learners and models thereof. In the second half of this chapter we give a brief introduction to Bayesian models, with a focus on the Bayesian Hidden Markov Model, a model used throughout this thesis. We conclude with a description of evaluation methods used to assess the clusterings generated by our and other models of syntactic category acquisition.

2.1 Computational Models of Cognition

Computational models of cognition aim to describe cognition at a high level of abstraction (Marr, 1982): the goal is to "reverse engineer" cognitive abilities and cognitive processes in order to gain understanding about the requirements and constraints of these processes. Even without making claims about the mechanistic nature of cognitive processes, computational models can be informative about the nature of the environment and cognitive representations that enable and facilitate cognitive abilities, as well as providing a deeper understanding of the nature of the cognitive process being investigated.

Rational models of cognition (Anderson, 1990; Griffiths et al., 2007) work at an additional level of abstraction, by aiming to demonstrate *optimal* behaviour (in a given context), on the assumption that human beings behave optimally within their environment.

> ... we can look in detail at what is outside the human head and try to de-

> termine what would be optimal behaviour given the structure of the environment and the goals of the human. The claim is that we can predict behaviour of humans by assuming that they will do what is optimal. (Anderson, 1990)

The assumption of optimality is merely a working assumption until there is clear evidence, for a given task, to the contrary. However, it offers a clear goal for models: to match optimal behaviour, and also requires a precise definition of what that entails.

Within the rational-computational modelling framework, the specification of the cognitive task is thus of primary importance. The researcher must define the 'environment', the 'goals of the human', and 'what is optimal'. None of these is straightforward: which aspects of the environment are relevant to the task? How does one precisely specify the goals? How will we know when they have been met? What does behaving optimally mean, in which context, under which constraints and pressures? In the process of specifying a model that can behave optimally, important questions about representations and causal interactions between components also arise: What kind of representations are sufficiently descriptive and powerful? Adequately parsimonious? Which components of the model interact, and in which ways? Being able to specify possible answers, or even to rule out certain answers, to these questions gives us a better idea of the problem we are trying to solve, a useful endeavour in itself.

In the experiments presented in this thesis, which involve rational computational models, we focus on the following issues:

Environment: Which elements of the input environment are relevant and helpful for the task at hand?

Representation: What form must the model-internal representations take to be able to solve the task?

Structure: How do the different components of the input and within the model interact?

Evaluation: How does the model output compare with what we expect, and how do we justify and quantify our expectations?

An important characteristic of humans' interaction with their environment is the continuous lack of complete knowledge. All observations are noisy, prone to distortion in the world and also through error-prone perception mechanisms. Other aspects of the environment are extremely hard to observe, due to physical or time scale or other factors. A consequence of these distortions is that humans are constantly engaged in reasoning of one form or another. Moreover, they engage in *inference*, both temporally, by making predictions about the future, and conceptually, by seeking causal understanding of their environment. In order to model this behaviour, we require a theory of reasoning and inference under uncertainty. *Bayesian probability theory* is such a theory, and one that has been fruitful in the field of cognitive modelling (e.g., Tenenbaum et al. (2006); Kemp et al. (2010); Perfors et al. (2011); Griffiths et al. (2007)).

Probability theory gives a representation of uncertainty as well as a system for manipulating sets of uncertain observations. Bayesian probability theory adds a method of performing induction in the form of belief updates. Observations are integrated with prior knowledge (hypotheses), resulting in updated hypotheses that can then be used for prediction, or re-updated with new observations. This process of updating beliefs based on observations is fundamentally what learning from experience entails. Additionally, using *generative models* gives us an intuitive way of including both production and comprehension aspects of domain (i.e. language) understanding.

The models in this thesis are not *process* models: they do not make claims about the mechanisms of learning or strong claims about the timecourse of learning. Process models are an important element of cognitive modelling, but they also require making additional assumptions about the nature of the learning process. We feel it is important to tackle the computational-level questions, described above, first. Then, armed with answers (or at least a better idea about the nature of possible answers), we can attempt a more mechanistic, process-oriented, model.

Computational models interact productively with experimental and behavioural research and aim to replicate behavioural results, while offering deeper insight into the necessary operations than may be accessible through human introspection or experiments. Developmental research with pre-verbal or quasi-verbal humans is perhaps particularly limited in this regard, given the difference between comprehension and production abilities at this stage. In the other direction, computational models may offer productive leads for behavioural research, by determining possible factors or interactions that have not yet been taken into account in experimental studies.

2.1.1 Rational Models of Language

Rational models of language are in an interesting position: remember, rational models have the goal of determining the optimal solution given a particular environment.

In most cases (e.g., vision) the environment is pre-determined and reasonably stable. Compared to the physics of optics, humans have a limited effect on their visual environment. Language, however, is an entirely human product. As a successful communication system, it must, by definition, be learnable. The field of *language evolution* (Brighton et al., 2005) aims to understand language as the result of cultural evolution. (Note that this requires a notion of language as a system inferred from noisy input — as the result of *iterated (Bayesian) learning* (Kirby et al., 2007) – rather than a nativist perspective, where the primary pressure on language change would be the result of biological evolution.)

This view of language, i.e., as a system that has been adapted to be transmissable and learnable, raises the question of which characteristics of language arise as a consequence of this adaptation. One clear candidate is *systematicity*, which arises directly from the need for language to be generalisable and productive: without, for example, a systematic morphology, neologisms would be impossible to understand. This systematicity is precisely what our models capture.

Rational models thus have a dual purpose in the context of language acquisition: we are both interested in *how* language is acquired by an (ideal) learner, as well as the characteristics of language that make it *able* to be learned. In the first case, the models we propose are claims about the necessary representation of the learner; in the second case, the models are hypotheses about useful cues for learning in the input environment.

2.2 Computational Models of Language Acquisition

One of the most complex cognitive systems is that of natural language. Every child learns language effortlessly (barring disability or extreme deprivation of environment), and yet a century of linguistic research has not achieved consensus on even a general characterisation of linguistic structure (Chomsky, 1957; Langacker, 1987; Croft, 2001).

One of the least controversial aspects of syntax is that languages require an abstraction of words into categories. Children do not learn an explicit class of ‘nouns’ or ‘prepositions’, but they must learn that words within a syntactic category are (at least to some extent (Croft, 2001)) exchangeable — they can fill the same syntactic roles. In Chomsky’s famous sentence, “Colourless green ideas sleep furiously”, the verb *sleep* could be replaced by other intransitive verbs (“Colourless green ideas argue furiously”); the noun can be replaced by another nouns (“Colourless green patriarchies sleep furiously”), and likewise for the adjectives and adverb. These sentences may not make much semantic sense, but they are grammatically correct.

A key aspect of syntactic similarity is the distributional similarity between words, i.e., words that appear in the same surface contexts (similar distributions) tend to belong to the same syntactic categories. Children are aware of distributional similarity (Saffran et al., 1996) and use it to infer category membership (Gómez and Gerken, 1999; Gómez and Lakusta, 2004; Gerken et al., 2005), and computational models of syntactic category induction rely almost entirely on distributional cues. In the remainder of this section we first discuss the acquisition of syntactic categories by humans, and thereafter describe the models that have been previously proposed for this task.

2.2.1 Acquiring Syntactic Categories

The task of language acquisition is composed of many subtasks that are performed over the course of many years, both concurrently and in sequence, building on previously acquired skills. This series of learning tasks begins with learning gross phonetic regularities of the mother tongue while still in the womb (Mehler et al., 1988; Mampe et al., 2009) and continues for nearly a decade, although the vast bulk of learning happens before the age of four.

In this thesis we are primarily concerned with early morpho-syntactic acquisition, in particular, the categorisation of words into parts of speech, based on their appearance in certain contexts and also based on the appearance of (morphological) patterns within the words themselves.

This process occurs mainly in the second and third years of life (Valian, 1986; Bernal et al., 2010; Tomasello and Olguin, 1993; Olguin and Tomasello, 1993; Booth and Waxman, 2003). The acquisition of syntactic categories is inextricably linked to other processes such as the acquisition of word meaning (Macnamara, 1978; Scott and Fisher, 2009), word order (Matthews et al., 2007), phonology (Kelly, 1992), and morphology (Akhtar and Tomasello, 1997).

Language learning happens within a rich environment of grounded physical and visual interaction with the world and social interaction with other humans. Separating the effects of all these different factors within an experimental setting can be very difficult.

The distributional hypothesis (Maratsos and Chalkley, 1980) posits that children use distributional information, by tracking surface co-occurrence of words, to bootstrap the acquisition of syntactic categories. Artificial language learning experiments (Gómez and Gerken, 1999; Gómez and Lakusta, 2004; Gervain et al., 2008; Onnis et al., 2008; Thothathiri et al., 2011) confirm that infants and children do perform inference about the structure of language streams that are devoid of non-linguistic semantic and contextual cues, based on statistical co-occurrence alone. While real-world language learning is undoubtedly richer and more powerful than learning based on purely distributional information, distributional information plays a vital part in language learning, and it is worth investigating and interrogating.

In this thesis we are primarily interested in demonstrating the informativeness of the data with regard to learning from distributional information. Distributional information is often used to refer only to local word-external syntactic contexts, but we use it more broadly, to apply to (roughly) all information contained within the utterance string (which corresponds to all the information we can glean from the database of child-directed speech we use as input for our models). It thus includes word-internal distributions of morphology as well as higher-level non-local syntactic contexts. However, it excludes semantic information about the meaning of the components and whole of the utterance, and also vital relevant information about the current (environmental/scene) context as well as possible pragmatic considerations. How to add these types of information robustly, at a large scale, to computational models is still an open question, although smaller scale experiments and approximations have been attempted (Kwiatkowski et al., 2012; Frank et al., 2009; Alishahi and Chrupala, 2012).

2.2.2 Computational Models of Syntactic Category Induction

Computational models of lexical syntactic category learning or induction appear both in the field of cognitive modelling as well as in natural language processing, where it is termed *unsupervised part of speech tagging*. Cognitive models generally attempt to be realistic by using learner-accessible input data (child directed speech) whereas NLP models focus on performance on larger sets of less realistic data (such as newswire). Furthermore, cognitive models tend to focus on category induction, i.e. clustering word *types* into groups that correspond to word classes, especially the basic word classes such as nouns and verbs, whereas part of speech tagging is a *token* labelling task, with performance measured using the entire data set (all tokens).

A central question is what kind of context statistics are sufficient for accurate category induction. Early work in category induction categorised word types using *context vectors*, where each word type is represented by a vector of counts of the local context in which it appeared; standard clustering techniques can then be applied to these vectors (Schütze, 1995; Redington et al., 1993, 1998; Mintz et al., 2002; Schütze and Walsh, 2008). An alternative representation is the *frame*, the joint occurrence of two words on either side of the target word (e.g., *the _ is*) (Mintz, 2003, 2006; Chemla et al., 2009; St Clair et al., 2010). Frames enable high accuracy clusterings but suffer from low coverage: only a small percentage of word types appear within frequent frames (Monaghan and Christiansen, 2004). Another method of increasing coverage is to use surrounding (inferred) categories as additional context, rather than words alone (Cartwright and Brent, 1997). All these methods operate over word types: they categorise words on the basis of the sum of all their appearances within the input data. This makes it for the most part impossible for these models to separate homographs or syntactically ambiguous words, which nevertheless appear frequently in natural language (although see Clark (2000) for one solution to this problem). This leads to both an unrealistic representation of words, and also adds noise to the models, when, for example, *'give me a _'* and *'don't _ the dog'* both appear as contexts for the word *kiss*, but not *hand* or *annoy*. It is also somewhat unclear how these models could operate in an incremental fashion, given their need for a large amount of data in order to have sufficient contexts for clustering.

Along with using the surrounding words as context, word-internal patterns are often helpful, if they successfully capture morphology (Brent, 1993; Onnis and Christiansen, 2005; Clark, 2003a; Monaghan et al., 2005, 2007). (See Chapter 5 for a more detailed exposition of the interaction between morphology and tagging.)

A few incremental models of category acquisition have been proposed (Parisien et al., 2008; Chrupala and Alishahi, 2010). These add word (tokens) to clusters as they appear in the data, without using complete statistics, unlike the above models. These models require an extra consolidation or merging step, and tend to create a large number of clusters.

For unsupervised POS tagging, in which each token is assigned a part of speech tag, the classic model used is the Hidden Markov Model (Brown et al., 1992; Saul and Pereira, 1997; Och, 1999; Banko and Moore, 2004; Merialdo, 1994; Christodoulopoulos et al., 2010) and more recently its Bayesian variant (Goldwater and Griffiths, 2007; Johnson, 2007; Gao and Johnson, 2008; Gael et al., 2009; Graca et al., 2011; Blunsom and Cohn, 2011; Hasan and Ng, 2009; Moon et al., 2009; Teichert and Daumé III, 2009). The Bayesian HMM will be described in full detail in Section 2.3.3, since we use it extensively.

These models can naturally incorporate syntactic ambiguity, since different tokens of the same word type can be assigned to different categories. In fact, a weakness of the HMM structure is that it has no overt way of constraining the level of tag-ambiguity (the number of tags that a word type can be assigned to) (Clark, 2003a).

HMMs use the previous tags as context, but note that since they model the full sequence of tags, the actual dependencies between all words (even non-neighbours) are stronger than in models that treat words+context as independent draws (such as the clustering models described above; see also (Toutanova and Johnson, 2007; Chrupala and Alishahi, 2010; Christodoulopoulos et al., 2011)). Along with local context, word-internal features have often found to be helpful (Dasgupta and Ng, 2007b; Abend et al., 2010; Christodoulopoulos et al., 2011; Haghighi and Klein, 2006; Blunsom and Cohn, 2011; Toutanova et al., 2003; Hasan and Ng, 2009).

Adding a constraint limiting each word type to a single tag (effectively turning the task into clustering as above) increases performance by decreasing spurious ambiguity, but also limits the linguistic realism of the model (Christodoulopoulos et al., 2010; Blunsom and Cohn, 2011; Lee et al., 2010).

2.3 Bayesian Foundations, or how the HMM became Bayesian: A Tutorial

In the rest of this chapter we go into some detail about the workings of Bayesian models in general and the Bayesian HMM in particular. This is intended to give background to the methods used in this thesis; for a more general introduction to Bayesian modelling see e.g. MacKay (2003); Bishop (2006). We first give a brief overview of Bayesian methodology, then describe the (non-Bayesian) HMM and subsequently the Bayesian version of the HMM, which appears repeatedly in this thesis. Finally, we discuss how inference is performed in these models, in the form of MCMC sampling and how it is implemented for the BHMM.

In this overview, we concern ourselves only with *directed* graphical models, also known as *Bayesian networks*; these represent causal structure in the form of conditional distributions. Our models are also *generative models*: we learn a full probability distribution over both observed and latent variables.

The high-level structure of such a model is usually suggested by the problem to be solved or the situation we wish to model. The input data constitute the observed variables and latent variables correspond to factors affecting the observed variables, meaning the observed variables will be conditioned on the latent variables in some way. In the Bayesian formulation, there is no technical distinction between parameters and latent variables; they are treated in the same manner. However, we find it useful to distinguish between (continuous) parameters and (discrete in our models, though this is not necessarily the case) latent variables. The distinction is based on our human understanding of the model: the latent variables are the hidden factors that we are aiming to discover, while parameters govern the distributions of interest.

The *joint probability distribution* over all observed, fixed, and latent variables and parameters defines the model structure. Given a set of observations generated by the model, the task of finding appropriate parameters (and latent variables; we shall group them together for now) is called inference.

Here the Bayesian methodology departs from frequentist statistics. Frequentists assume fixed ‘true’ parameter values that are to be estimated by a single point estimate, and thus aim to find the best set of parameters Θ, given some data $\mathcal{D}$, for instance by maximising data likelihood: $\hat{\Theta}_{\text{ML}} = argmax_{\Theta}\mathcal{L}(\Theta;\mathcal{D})$.

Bayesians infer a density estimate over all possible parameter values, to capture inherent uncertainty about the values of parameters. Uncertainty will diminish with increased observations, as each observation results in an update of the density estimate. This allows for better prediction of new observations or latent variables, as it does not require assuming any more knowledge about the true parameter values than is available from the data. The data likelihood is simply the probability distribution of the data given the parameters: $\mathcal{L}(\Theta;\mathcal{D}) = P(\mathcal{D}|\Theta)$. The goal is to estimate the posterior distribution $P(\Theta|\mathcal{D})$, since we know $\mathcal{D}$ but not Θ. Using Bayes’ Rule, we find:

$$P(\Theta|\mathcal{D}) = \frac{P(\mathcal{D}|\Theta)P(\Theta)}{P(\mathcal{D})} = \frac{P(\mathcal{D}|\Theta)P(\Theta)}{\int_{\Theta} P(\mathcal{D}|\Theta)d\Theta} \tag{2.1}$$

There are two important things to note here: firstly, the fact that we now require a prior probability over the parameters, $P(\Theta)$, and secondly, the integral in the denominator foreshadowing difficulties for inference. In Section 2.3.3 we investigate how using certain forms of priors can help simplify the posterior, and in Section 2.3.4 we discuss how inference can be performed via numerical sampling methods.

Having found a distribution over parameters given the data seen so far, we use this posterior distribution to make predictions about future events. As mentioned before,

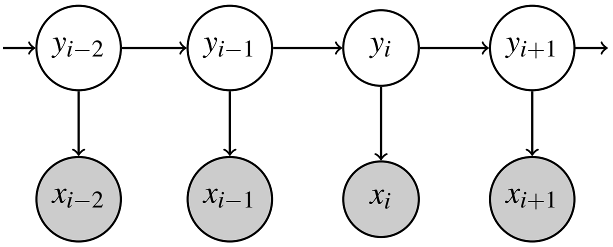

Figure 2.1: First order Hidden Markov Model

this involves integrating over the full distribution of all parameter values:

$$P(x|\mathcal{D}) = \int_{\Theta} P(x|\Theta)P(\Theta|\mathcal{D})d\Theta \tag{2.2}$$

In the following two sections, we first describe a classic non-Bayesian model, the Hidden Markov Model, and then recreate it as a Bayesian model. There is discussion of the basic structure and inference methods associated with these models. For more substatial tutorials and descriptions of the HMM, see Ghahramani (2001); Welch (2003); Bishop (2006); Manning and Schütze (1999).

2.3.1 Hidden Markov Model

The Hidden Markov Model is a *sequence model*: it models a particular sequence of observations $\mathcal{X}$, rather than simply a set of observations. The observations take the form of a vector $\boldsymbol{x}$, with observation x_i at position i. If we rearrange $\boldsymbol{x}$, so that the observation at x_i is changed, the resulting inferred model will be different. (Compare to mixture or Latent Dirichlet Allocation models, where the order of $\mathcal{X}$ does not matter, and rearrangements of $\mathcal{X}$ will lead to the same inferred model.) The positions i are often called timesteps (and the time metaphor is pervasive in descriptions of HMMs), but the HMM is applicable to any kind of sequence, whether temporal or physical or other.

Secondly, the Hidden Markov Model, as its name implies, includes latent (hidden) variables. More precisely, as the graphical model in Figure 2.1 shows, each observation x_i is generated from a latent variable y_i at the same timestep. The HMM can thus be thought of as a sequence of mixture model draws, linked together by *transitions* between mixture model components.

Finally, the HMM is called a *Markov* model because it exhibits the Markov property: only a restricted, truncated, history is used to predict subsequent states. Instead of taking into account the full sequence of previously seen states to predict the next state, only a fixed number of previous states are used (one previous state in a first order

HMM, two in a second order HMM, etc.). So, by definition, in a first order HMM,

$$p(y_{i+1}|y_i, y_{i-1}, y_{i-2}, ..., y_2, y_1, y_0) = p(y_{i+1}|y_i), \tag{2.3}$$

that is to say, all recent histories are considered to be the same, regardless of the full history chain. This is clearly evident from examination of the graphical model: y_{i+1} is d-separated from y_{i-1} by y_i, which indicates that they are conditionally independent given y_i.

2.3.2 Model Structure

An HMM is defined using two probability distributions, the *emission* distribution and the *transition* distribution.

2.3.2.1 Emission Distribution

The emission distribution $p(x|y)$ defines the probability of an observed output x given hidden state y. The output vocabulary X, i.e. the set of possible emissions, is set in advance, as is Y, the set of possible states. We can describe the emission distribution as a row-wise stochastic matrix E of size $|Y| \times |X|$. For example, if we have three states and two possible outcomes, a possible E is:

$$E = \begin{bmatrix} 0.3 & 0.7 \\ 0.1 & 0.9 \\ 0.0 & 1.0 \end{bmatrix} \tag{2.4}$$

where $e_{1,0} = 0.1$, the probability of seeing outcome 0 from state 1. Note that as a stochastic matrix, all $e_{ij} \geq 0$ and $\sum_j e_{ij} = 1$ for all e_i. Apart from these constraints, E is free to vary; generally it is inferred on the basis of observations $\boldsymbol{x}$, as we shall see shortly. In this example (and throughout this thesis) the observations $\boldsymbol{x}$ are discrete; other variants of the HMM may emit real values (e.g. draws from a normal distribution).

2.3.2.2 Transition Distribution

The transition distribution describes the probability of moving from one hidden state to the next, and can also be described using a stochastic matrix T. For a first order HMM T will have dimension $|Y| \times |Y|$; a second order HMM will be a tensor of size

$|Y| \times |Y| \times |Y|$. For example, in a first order HMM with three states, T might be:

$$T = \begin{bmatrix} 0.3 & 0.3 & 0.4 \\ 0.7 & 0.2 & 0.1 \\ 0.4 & 0.1 & 0.5 \end{bmatrix} \tag{2.5}$$

where $t_{1,1}$, the probability of remaining in state 1 for two timesteps (i.e. starting in state 1 at timestep i and transitioning to state 1 at timestep $i+1$) is 0.2.

The joint probability of a (first order) HMM for a string of length N is thus:

$$P(\boldsymbol{x},\boldsymbol{y}|T,E) = \prod_{i=0}^{N} T_{y_{i-1}y_i} E_{y_i x_i} \tag{2.6}$$

where we add a dummy start state at the beginning (at $i = -1$) to initialise the transition sequence. (This means that T will have an extra 'first state' row; other descriptions of HMMs store this distribution, often called π, separately.)

2.3.2.3 Parameter Estimation for HMM: EM

In an unsupervised setting, we have observations $\boldsymbol{x}$ and wish to estimate $\boldsymbol{y}$, the values of the hidden states, as well as parameters E and T. The parameters can be found using the classic Baum-Welch algorithm (Baum et al., 1970; Welch, 2003). This is an earlier, HMM-specific variant of the more general Expectation Maximisation algorithm (Dempster et al., 1977). Once the parameters are set, they can then be used to find the most probable sequence of $\boldsymbol{y}$ given observations $\boldsymbol{x}$, using the Viterbi algorithm (Viterbi, 1967; Forney, 1973).

The EM algorithm iterates between an Expectation step, in which a maximum likelihood expectation for the current log-likelihood, given current parameter values, is found, and the Maximisation step, where the parameters are updated so as to maximise the expected log-likelihood. The expectation step requires finding the marginal posteriors of the hidden states, which is done by using the forward-backward algorithm, an HMM-specific variant of the more general message-passing algorithm. More detailed descriptions of EM, the Baum-Welch variant, and full expositions of the forward-backward algorithm can be found in e.g. Manning and Schütze (1999); Bishop (2006).

EM is only guaranteed to find a local optimum of the log-likelihood. Additionally, as a maximum likelihood method, EM finds only a single point estimate of the parameters (E and T). This means that, especially for small or sparse datasets, the estimate can be quite brittle; e.g. if an observation is never seen, it will have zero probability (unless some kind of smoothing is added).

Goldwater and Griffiths (2007) and Johnson (2007) find that EM-estimated HMMs, optimised to find a maximum likelihood solution, performed poorly on the task of unsupervised part of speech tagging. This was due to the found distribution of words to tags being much more uniform than the empirical distribution, which is severely skewed, with a few large tag clusters and several small and even tiny ones (e.g. *not*, *to*).

2.3.3 Bayesian HMM

As explained earlier, in a general Bayesian model we are interested in finding the posterior of the parameters given the observations $\mathcal{X}$. If we separate out the (discrete) latent variables $\mathcal{Y}$ from Θ in Equation 2.1, the posterior distribution becomes:

$$\frac{P(\mathcal{X}|\Theta,\mathcal{Y})P(\mathcal{Y}|\Theta)P(\Theta)}{P(\mathcal{X})} = \frac{P(\mathcal{X}|\Theta,\mathcal{Y})P(\mathcal{Y}|\Theta)P(\Theta)}{\int_\Theta \sum_\mathcal{Y} P(\mathcal{X}|\Theta,\mathcal{Y})d\Theta} \tag{2.7}$$

In both cases, this requires determining a prior over the parameters (more precisely, a prior for each parameter). In the HMM, the principal distributions (transitions and emissions, which we now denote with τ and ω, respectively, to emphasize that they are random variables) are *multinomials*; it turns out that if we pick *Dirichlet* distribution priors for these, calculation of the posterior becomes easier. We now describe why.

2.3.3.1 Multinomial Distributions

A multinomial distribution describes the probability of a set of draws from a discrete set of K values. When $K = 2$, it is equivalent to the Binomial distribution. A multinomial is parameterised as $\text{Mult}(\langle\theta_1, \theta_2, ...\theta_k\rangle)$; each θ_k represents the probability of seeing a draw from bin k. Since $\boldsymbol{\theta}$ is a probability distribution, $\sum_k \theta_k = 1$ and all $\theta_k \geq 0$. The probability mass function of n draws, of which n_k are from bin k, from a distribution of the above form, is:

$$P(X_1 = n_1, X_2 = n_2, \cdots, X_k = n_k|\boldsymbol{\theta}) = \frac{n!}{\prod_k n_k!} \prod_k \theta_k^{n_k} \tag{2.8}$$

When only a single value is drawn, the multinomial is equivalent to the categorical distribution. (This is the same distinction that is made between the binomial and Bernoulli distribution.)

2.3.3.2 Dirichlet Distribution

In order to put a prior over the parameters $\boldsymbol{\theta}$ in the multinomial, we need a prior distribution from which we can draw the $\boldsymbol{\theta}$ probability distribution required by the multinomial. The *Dirichlet* distribution is precisely such a distribution: it is a distribution over probability distributions, written $\text{Dirichlet}(\boldsymbol{\alpha}) = \text{Dirichlet}(\langle\alpha_1, \alpha_2, ...\alpha_k\rangle)$. Draws from this distribution are likewise discrete probability distributions of size K, summing to 1.

The probability density function of a Dirichlet distribution $\boldsymbol{\theta}$ takes the following form:

$$P(\boldsymbol{\theta}|\boldsymbol{\alpha}) = \frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_k \theta_k^{\alpha_k - 1} \tag{2.9}$$

where $\Gamma(x)$ is the Gamma function. When x is an integer, $\Gamma(x) = (x-1)!$; note the similarity to the multinomial above.

In most cases, we use a symmetric Dirichlet in which all k αs are set to the same value, in which case we simply write α. A non-symmetric Dirichlet would encode prior knowledge about which components are more or less probable. If we do not have such prior knowledge, using a symmetric Dirichlet prior is more appropriate.

The $\boldsymbol{\alpha}$ or α parameter is called the *concentration parameter*; it controls how peaked the (most probable) distributions drawn from the Dirichlet are. Smaller values of α generate more peaked (sparser) distributions. Examining Eq. 2.9, we can see that α functions as a kind of smoothing: exponentiating θ by $\alpha - 1$ will result in higher values when $0 < \alpha < 1$ and lower values for $\alpha > 1$. Setting $\alpha = 1$ results in a uniform prior over all distributions.

By adding a Dirichlet prior over the multinomials in the BHMM, we can encode desirability of *sparsity*. In the part of speech tagging task, we want hidden states to emit only some of the possible outputs with high probability, and similarly most transitions between hidden states should be very unlikely. For example, if a given hidden state represents the set of adjectives, it should emit only a small fraction of the full vocabulary with high probability, and give very small probability to all non-adjectives (nouns, verbs, function word, etc.). Likewise, within the transition distribution, this hidden state should assign a high probability to a following noun or perhaps adjective, but a low probability to most other parts of speech, since e.g, `ADJ VERB` or `ADJ DET` are rare sequences within the data.

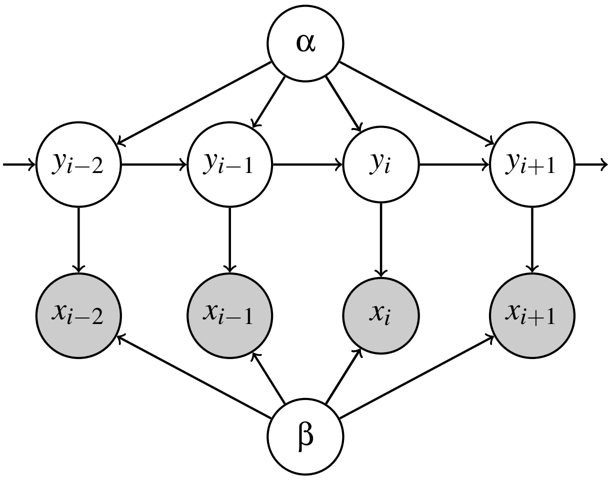

Figure 2.2: First order Bayesian Hidden Markov Model

2.3.3.3 Dirichlet-Multinomial Conjugacy

In order to find the posterior probability we are interested in, we must multiply the prior and the likelihood together. This multiplication is made considerably easier by using *conjugate priors*: these are families of probability distributions which, when multiplied together (prior times likelihood), result in a posterior probability distribution of the same form as the prior. In our case, we have multinomial likelihood distributions; the Dirichlet distribution is the conjugate prior of the multinomial, so the posterior will also have the form of a Dirichlet:

$$\begin{aligned} P(\boldsymbol{\theta}|\mathcal{X},\alpha) &\propto P(\mathcal{X}|\boldsymbol{\theta})P(\boldsymbol{\theta}|\alpha) && (2.10)\\ &\propto \prod_k \theta_k^{n_k} \prod_k \theta_k^{\alpha_k-1} && (2.11)\\ &\propto \prod_k \theta_k^{n_k+\alpha_k-1} && (2.12)\\ &\propto \text{Dirichlet}(\langle n_1+\alpha_1,\ldots,n_k+\alpha_k\rangle). && (2.13) \end{aligned}$$

This form also demonstrates why the Dirichlet parameters (α, called hyperparameters to distinguish them from the parameters of the multinomial likelihood) are sometimes called *pseudocounts*; they have a similar effect as add-*n* smoothing.

2.3.3.4 Bayesian HMM

Returning to the BHMM: we add a Dirichlet prior over each of the multinomial distributions, the transitions τ and emissions ω. These Dirichlets are governed by the hyperparameters α and β, respectively. Aside from the priors, the remainder of the model is the same as the HMM: at each timestep, the transition parameters $\tau_{(y_{i-1})}$ (a multinomial distribution) are used to draw a new hidden state y_i based on the previous one y_{i-1}. Each observation x_i is drawn from the emission distribution (also a multinomial) indexed by the current hidden state $\omega_{(y_i)}$. This results in the following model (see also Fig. 2.2):

$$\tau_{(y)} \sim \text{Dirichlet}(\alpha) \tag{2.14}$$

$$\omega_{(y)} \sim \text{Dirichlet}(\beta) \tag{2.15}$$

$$y_i | y_{i-1} = y' \sim \text{Mult}(\tau_{(y)}) \tag{2.16}$$

$$x_i | y_i = y \sim \text{Mult}(\omega_{(y)}) \tag{2.17}$$

Once we have determined the form of the priors and likelihood in the BHMM, we can write down the full joint probability for a first order BHMM (compare to 2.6):

$$p(\boldsymbol{x}, \boldsymbol{y}, \tau, \omega | \alpha, \beta) = p(\omega|\beta) p(\tau|\alpha) \prod_{i=0}^{N} \tau_{(y_{i-1}, y_i)} \omega_{(y_i, x_i)} \tag{2.18}$$

However, within this model we are principally interested in the values of the latent variables $\boldsymbol{y}$, rather than the parameters. Thus we integrate over all possible values of the parameters, leading to a marginal joint probability over $\boldsymbol{x}$ and $\boldsymbol{y}$.

$$p(\boldsymbol{x}, \boldsymbol{y} | \alpha, \beta) = \int_\tau \int_\omega p(\boldsymbol{x}|\boldsymbol{y}, \omega) p(\boldsymbol{y}|\tau) p(\omega|\beta) p(\tau|\alpha) d\omega d\tau \tag{2.19}$$

$$= \int_\tau \int_\omega \prod_{y \in Y}^{Y} \prod_{x \in X}^{X} \omega_{(yx)}^{n_{yx}} \prod_{y,y' \in Y}^{Y} \tau_{(yy')}^{n_{yy'}} \frac{\Gamma(\beta)}{\Gamma(Y\beta)} \prod_{y \in Y}^{Y} \prod_{x \in X}^{X} \omega_{(yx)}^{\beta-1} \frac{\Gamma(\alpha)}{\Gamma(Y\alpha)} \prod_{y,y' \in Y}^{Y} \tau_{(yy')}^{\alpha-1} d\omega d\tau \tag{2.20}$$

$$= \frac{\Gamma(\alpha)}{\Gamma(Y\alpha)} \frac{\Gamma(\beta)}{\Gamma(Y\beta)} \int_\omega \prod_{y \in Y}^{Y} \prod_{x \in X}^{X} \omega_{(yx)}^{n_{yx}+\beta-1} d\omega \int_\tau \prod_{y,y' \in Y}^{Y} \tau_{(yy')}^{n_{yy'}+\alpha-1} d\tau \tag{2.21}$$

$$= \frac{\Gamma(\alpha)}{\Gamma(Y\alpha)} \frac{\Gamma(\beta)}{\Gamma(Y\beta)} \frac{\prod_{xy} \Gamma(n_{yx}+\beta)}{\Gamma(\sum_{y \in Y}(n_y+\beta))} \frac{\prod_{yy'} \Gamma(n_{yy'}+\alpha)}{\Gamma(\sum_{y \in Y}(n_y+\alpha))} \tag{2.22}$$

In the third line, in order to get rid of the integrals, we make use of the fact that the parameters must be proper probability distributions and sum to 1. Within the integrals we have (posterior) Dirichlets, and so the following holds, using the definition of the Dirichlet (Eq. 2.9):

$$\int_\theta \prod_k \theta_k^{\alpha-1} d\theta = \frac{K\Gamma\alpha}{\Gamma(\alpha)} \tag{2.23}$$

Note that the sufficient statistics in the marginal joint distribution are the counts of the occurrences of tag-word pairs and bigrams (trigrams in a second order HMM), as well as the hyperparameters. Importantly, the counts alone are sufficient; the order in which they occurred is irrelevant. This characteristic means the model is *exchangeable*, which will be important later on for sampling.

2.3.4 Inference in Bayesian Models: Sampling

In this section we discuss how to do sampling-based inference within a Bayesian model. We introduce Markov Chain Monte Carlo (MCMC) and Gibbs sampling, and then show how the latter can be used in the BHMM to find values for the hidden states $\boldsymbol{y}$.

Exact inference in (reasonably complex) Bayesian models is intractable. To see why, recall that the posterior includes the marginal likelihood of the data, which takes the form of a hefty integration in the denominator. If we have discrete latent variables, such as $\boldsymbol{y}$ in the BHMM, the marginal also includes a sum over all configurations of those variables.

Here, we focus on random sampling methods, which produce a chain of random samples from the posterior. There are also deterministic approximation methods, such as Variational Bayes and Expectation Propagation, which we do not cover: see Beal (2003); Bishop (2006); Minka (2001) for introductions. For more detailed expositions of sampling methods, see Neal (1993); MacKay (2003); Bishop (2006); Besag (2004); Andrieu et al. (2003).

Sampling methods side-step the problem of integration by simply drawing a large number of samples from $\tilde{P} = P(\mathcal{D}|\Theta)P(\Theta)$, i.e., the unnormalised posterior. It is thus important that the samples are, firstly, indeed from the correct distribution, and secondly, that they are independent. Only so will they provide an accurate representation of the full distribution.

Producing samples from $\tilde{P}$ is not straightforward, since it requires knowing about the nature of $\tilde{P}$, precisely what we are trying to discover. MCMC methods use an initial number of 'burn-in' samples to transition to the correct distribution. Non-MCMC sampling methods, such as rejection sampling, do not require burn-in; instead they use a proposal distribution that is approximate to $\tilde{P}$ and then check whether to accept the proposed sample. However, in high dimensional state spaces (e.g., $\boldsymbol{y}$ in the BHMM) too many proposals will be rejected, making these methods intractable.

2.3.4.1 Markov Chain Monte Carlo (MCMC) Sampling

The name *Monte Carlo* implies random sampling of a quantity (in our case, $\tilde{P}$). MCMC sampling methods produce a sequence — a Markov chain — of samples, each of which depends on the previous one, i.e. sample z_i will be drawn conditionally on z_{i-1}. The probability of transitioning to the next sample is given by the transition matrix T, which must be defined such that the sequence of samples will eventually converge on the true distribution $\tilde{P}$.

There are two theoretical requirements for an appropriate Markov chain for finding $\tilde{P}$:

Invariance: Once the Markov chain has reached the target distribution $\tilde{P}$, it must remain there. (This is also known as a Markov chain with a stationary transition distribution.) By definition, once this is the case: $\tilde{P}(z) = \sum_z T(z', z)\tilde{P}(z')$, i.e. after any transition from a state in the invariant distribution, the next state will also be from the invariant distribution.

Ergodicity: The Markov chain must eventually converge to $\tilde{P}$, as time goes to infinity. (Also known as: the Markov chain must have $\tilde{P}$, the invariant distribution, as its equilibrium distribution.) The simplest way to satisfy this requirement is to ensure that T is non-zero everywhere, including at self-transitions (i.e. that we draw the same sample twice in a row). This means that eventually, as the result of random draws, the sampler will transition into a state that is in $\tilde{P}$, and because of invariance, the sampler will then remain there. (Note that there is no good guarantee or estimate of how long it will take the sampler to converge to $\tilde{P}$.)

2.3.4.2 Gibbs Sampling

We now describe one form of MCMC sampling, Gibbs sampling (Geman and Geman, 1984), and show that it meets the theoretical requirements above. Gibbs sampling is extremely straightforward and effective for high dimensional problems. Given a multi-dimensional $\mathcal{X}$, the sampler iterates through all components of $\mathcal{X}$, $x_1 ... x_n$, and updates the value of each component using the current value of all the other components, based on the predictive posterior probability $p(x_i|\mathcal{X}^{\setminus i})$. The notation $\mathcal{X}^{\setminus i}$ means 'all x except x_i'; in practice, e.g. in a multinomial distribution, this means that at each sampling step, we decrement the counts for the current values of the variables at x_i, sample x_i based on the decremented counts, and then add new counts for the resampled x_i. We continue to iterate over $\mathcal{X}$ until convergence.

Gibbs sampling fills the ergodicity requirement if we ensure $p(x_i|\mathcal{X}^{\setminus i}) > 0$ for all x_i. Invariance can be shown by noting that, once we are sampling from $\tilde{P}$, at every step the marginal probability $P(\mathcal{X}^{\setminus i})$ remains constant, and the new x_i is drawn from the correct (invariant) distribution. The ensuing joint $P(x_i, \mathcal{X}^{\setminus i})$ is also invariant: once the invariant distribution has been reached, all subsequent samples will also be from the invariant distribution.

Due to the strong conditional dependence between samples, care must be taken to obtain independent samples, at intervals of i sufficiently far apart.

Gibbs sampling is significantly simplified when the distribution to be sampled from is *exchangeable*, i.e., the sufficient statistics necessary for sampling do not depend on the ordering of the observations. This means that we can treat each component x_i as the last one in the sequence. In Section 2.3.3.4 we noted that the Dirichlet-multinomial posteriors in the BHMM are exchangeable; we now show that this leads to a simple form for the Gibbs sampler for the BHMM.

2.3.5 Gibbs Sampling for BHMM

Gibbs sampling for the BHMM is straightforward: we simply iterate over the sequence of hidden states $\boldsymbol{y}$ and update each y_i in turn, conditioned on the current values of all other $\boldsymbol{y}^{\setminus i}$.

The parameters τ and ω may be resampled explicitly by drawing from $p(\tau_{(t)}|\boldsymbol{x}, \boldsymbol{y}, \tau_{(\setminus t)})$ and analogously for ω. We can also take advantage of the Dirichlet-multinomial conjugacy to *collapse* (integrate out) τ and ω, following Eq. 2.22.

We show how to derive the closed form for a simple Dirichlet-multinomial with observed draws $\mathcal{X}$ from Mult(θ) and a prior Dirichlet($\boldsymbol{\alpha}$) and then use this form in the full BHMM Gibbs sampling equation. (Because the posterior Dirichlet is not symmetric, we use the non-symmetric α_k notation here rather than a symmetric α; of course these equations also hold for symmetric Dirichlet(α).)

$$\begin{aligned}
p(x_i = x|\mathcal{X}^{\backslash i}; \boldsymbol{\alpha}) &= \int_\theta p(x|\theta) p(\theta|\mathcal{X}^{\backslash i}, \boldsymbol{\alpha}) d\theta && (2.24)\\
&= \int_\theta \theta^x \frac{\Gamma(\sum_k n_k + \alpha_k)}{\prod_k \Gamma(n_k + \alpha_k)} \prod_k \theta^{n_{x_k} + \alpha_k - 1} d\theta && (2.25)\\
&= \frac{\Gamma(\sum_k n_k + \alpha_k)}{\prod_k \Gamma(n_k + \alpha_k)} \int_\theta \theta^{n_x + \alpha_x} \prod_{k \neq x} \theta^{n_k + \alpha_k - 1} d\theta && (2.26)\\
&= \frac{\Gamma(\sum_k n_k + \alpha_k)}{\prod_k \Gamma(n_k + \alpha_k)} \frac{\Gamma(n_x + \alpha_x + 1) \prod_{k \neq x} \Gamma(n_k + \alpha_k)}{\Gamma(1 + \sum_k (n_k + \alpha_k))} && (2.27)\\
&= \frac{(n_x + \alpha_x) \Gamma(\sum_k n_k + \alpha_k) \Gamma(n_x + \alpha_x) \prod_{k \neq x} \Gamma(n_k + \alpha_k)}{\sum_k (n_k + \alpha_k) \Gamma(\sum_k n_k + \alpha_k) \prod_k \Gamma(n_k + \alpha_k)} && (2.28)\\
&= \frac{n_x + \alpha_x}{\sum_k n_k + \alpha_k} && (2.29)
\end{aligned}$$

In Eq. 2.27 we use the Dirichlet normalisation constant to replace the integral over θ (as before in Eq. 2.22, using Eq. 2.23). In Eq. 2.28 we make use of the $\Gamma(x+1) = x\Gamma(x)$ identity and rearrange factors to make it clear that they all cancel out, leaving us with a tidy, extremely simple form for the predictive probability. When using a symmetric α, the predictive probability is $p(x_i = x|\mathcal{X}^{\backslash i}; \alpha) = \frac{n_x + \alpha}{n + K\alpha}$.

Within the BHMM, the predictive posterior for the last y_i given all $Y^{0..i-1}$ is the product of the posteriors for the emission and transition distributions:

$$p(y_i = y|x_i = x, y_{i-1} = y', \mathbf{y}^{0 \cdots y_{i-1}}; \alpha, \beta) = \frac{n_{y'y} + \alpha}{n_{y'} + X\alpha} \times \frac{n_{yx} + \beta}{n_y + Y\beta}. \tag{2.30}$$

However, the predictive probability above only holds for the last y_n. When doing Gibbs sampling, we will be predicting a y_i anywhere in $\mathbf{y}$. This results in a dependence not only between y_{i-1} and y_i, but also between the values of y_i and y_{i+1}, even y_{i+2} in a trigram model. In a trigram model, y_i is included in three trigrams: (y_{i-2}, y_{i-1}, y_i), (y_{i-1}, y_i, y_{i+1}) and (y_i, y_{i+1}, y_{i+2}). Each of these trigrams must be included in the sampling probability. If any of these trigrams are the same additional counts must be added to the later trigram probabilities. The full Gibbs sampling probability for a trigram

BHMM thus becomes a bit complex:

$$p(y_i = y|x_i = x, \mathbf{y}^{\setminus i}, \tau, \omega, \alpha, \beta) \tag{2.31}$$

$$\propto \frac{n_{xy} + \beta}{n_y + X\alpha} \times \frac{n_{yy_{i-1}y_{i-2}} + \alpha}{n_{y_{i-1}y_{i-2}} + Y\alpha} \tag{2.32}$$

$$\times \frac{n_{y_{i+1}yy_{i-1}} + [y_{i+1} = y = y_{i-1} = y_{i-2}] + \alpha}{n_{yy_{i-1}} + [y_i = y_{i-1} = y_{i-2}] + Y\alpha} \tag{2.33}$$

$$\times \frac{n_{y_{i+2}y_{i+1}y} + [y_{i+2} = y_{i-1} = y = y_{i+1}] + [y_{i+2} = y = y_{i-2} \wedge y_{i+1} = y_{i-1}] + \alpha}{n_{y_{i+1}y} + [y_{i+1} = y = y_{i-1}] + [y_{i+1} = y \wedge y_{i-1} = y_{i-2}] + Y\alpha} \tag{2.34}$$

where [ ] are Iverson brackets that equal 1 if the condition within them is true, and Y is the number of states in the BHMM.

2.3.6 Using Sampling as Search

After convergence, MCMC samplers such as the Gibbs sampler produce samples from the unnormalised posterior distribution $\tilde{P}$. These samples can be used to calculate various statistics and distributions of interest, such as means and marginal distributions. For example, samples from a normal distribution will have an expected mean that approximates the distribution mean with increasing precision as the number of samples increases.

However, the marginal distribution of a sampled multinomial will be flat if the sampler has converged. This is due to the non-identifiability problem, also known as the label-switching problem, namely that the categories in the multinomial are *a priori* indistinguishable given a symmetric prior. Any relabelling of categories will have the same posterior probability under the model. A converged sampler will sample from all $k!$ relabellings, each corresponding to a symmetric mode of $\tilde{P}$ evenly, resulting in a uniform, uninformative, posterior.

Some remedies to this problem have been proposed (Celeux et al., 2000; Stephens, 2000; Jasra et al., 2005), which essentially attempt to enforce a consistent labelling over many samples, either with regards to a loss function or a priori ordering of some parameter. These proposals, however, have only been evaluated on simple models (a single small multinomial), and it is doubtful whether they would scale to models involving many large, tightly coupled, multinomials, such as the BHMM.

Even if we were able to infer clean marginal posteriors over the tags in our dataset, it is unclear how to evaluate these distributions. Our gold tagged data give us only a single ‘right answer’ against which to compare our model. (We note that this does not necessarily imply that humans have a representation of syntactic categories that does not include distributional information, as in our model. Indeed, grammaticality and category prototypicality effects imply precisely such a cognitive representation. In this case, gold tags can be understood as a ‘sample’ of human judgements.)

The standard solution to this dilemma is to use a single sample for evaluation (Goldwater and Griffiths, 2007; Johnson, 2007; Liang et al., 2010; Perfors et al., 2011). The maximum a posterior (MAP) solution (i.e., the sample with the highest posterior probability), or rather, the closest approximation found by the sampler, is taken as a point estimate of the full distribution. In this case, the sampling procedure becomes an optimisation procedure: the aim is to find the closest approximation to the MAP solution, rather than producing representative samples of $\tilde{P}$.

To this end, we use simulated annealing, a technique to manipulate the shape of $\tilde{P}$ during sampling by lowering an exponentiating temperature parameter T, i.e. setting $p(x) = p(x)^{1/T}$. At the start, T is set to a large value, with the effect of flattening the distributions in order to explore the search space, and is gradually decreased to 1. Finally, in the last iterations the temperature is lowered still further, pushing the sampler to a local optimum (a MAP estimate).

It may be worthwhile to reiterate that despite the fact that we are only using a single sample of the hidden variables $\boldsymbol{y}$ from $\tilde{P}$, rather than using the full distribution, we are nevertheless still integrating over all the parameters (θ and ϕ in the BHMM), and thus representing our uncertainty over possible parameter values; in this sense we remain Bayesian.

2.3.7 Other Inference/Sampling Methods for BHMMs

The Gibbs sampler described above is extremely straightforward. However, due to extensive sequential coupling in the structure of the model, it can be slow to ‘mix’, remaining in a small sub-portion of the full sampling space for long periods, resulting in slow convergence.

Other types of Gibbs samplers have been proposed, in particular for the part of speech tagging task, which try to deal with this problem by resampling multiple token points at once:

Block samplers resample a whole sentence at once, using the Forward-Backward algorithm to find a good resampling of a whole sequence of tags, rather than just a pointwise resample. Gao and Johnson (2008) did not find any consistent improvement over pointwise collapsed Gibbs, described above, when using a block sampler for POS tagging.

Type samplers resample all instances of a given 'type' of hidden state, i.e. states with the same contexts, at once. Liang et al. (2010) found very moderate improvements in the case of HMMs for POS tagging.

1 Tag/Word Type samplers add a restriction, pertinent to unsupervised part of speech tagging, that limits each word type to a single tag. A number of recent unsupervised POS tagging systems have added this restriction to their models, some in the inference section (Blunsom and Cohn, 2011), others within the model structure (Lee et al., 2011b; Christodoulopoulos et al., 2011). In these models, all tokens of a word type are resampled at once, and constrained to be equal (same hidden state value). This often improves accuracy at the expense of model realism, by ignoring tag ambiguity.

Finally, Variational Bayes (VB) for BHMM is also straightforward (MacKay, 1997; Johnson, 2007). The resulting optimisation updates iteratively optimise the current estimations of the latent variables $\boldsymbol{y}$ and of the parameters, resulting in an algorithm closely resembling EM. For larger problems, VB is significantly easier to parallelise within e.g. a MapReduce framework. In contrast, parallel variants of MCMC samplers are much less straightforward due to its sequential, highly-coupled, nature. However, VB requires making an independence assumption between the parameters and latent variable values that sampling methods do not require.

In our experiments, we use the collapsed Gibbs sampler described above, eschewing the more complex samplers, since they have not proven to be sufficiently more effective.

2.4 Evaluating Clusters

In this section we describe an number of methods for evaluating the tags found by the BHMM (or other unsupervised part of speech tagger) against the gold tags. The sequence of tags induced by the BHMM can be thought of as a clustering of tokens into part of speech categories; likewise a type-based system induces a clustering over word types into part of speech categories.

The key difficulty in evaluating these induced clusters, of either types or tokens, is that the clusters have no labels: we do not know which (if any) gold part of speech tag each cluster is supposed to correspond to. If labels were given, evaluating accuracy (i.e., percentage of items with correct labels) would be straightforward, but without labels, the task becomes much more ambiguous. There are a number of commonly used measures which we now describe briefly. In this thesis we primarily report *V-measure*, since it has been found to be more robustly comparable across different settings than the other measures (Christodoulopoulos et al., 2010).

2.4.1 Matched Accuracy

The simplest cluster evaluation metric, *Matched Accuracy*, builds on the idea of labelled accuracy. Each induced cluster is matched to the gold categories using a *many to one* mapping strategy, in which each cluster receives the label of the gold category of the majority of its members. Multiple clusters may be labelled with the same gold category, if the gold category is split over two or more clusters. The labels are then used to calculate accuracy, as the percentage of words that have been mapped to the correct gold category.

Matched accuracy is widely used for the evaluation of unsupervised part of speech tagging systems. If the number of clusters is greater than the number of gold categories, matched accuracy does not penalise the model in any way; this evaluation metric is thus problematic when evaluating multiple systems or clusterings with varying numbers of clusters, since the clustering with the most clusters will have a significant advantage. An alternative mapping is *one to one*, in which each gold category may only be mapped to once by a model cluster. If there are more clusters than gold categories, the unmapped words are considered wrong. This removes the incentive to create too many clusters, but restricts the usefulness of this mapping.

Moreover, both of these mappings suffer from the "problem of matching" (Meila, 2007): matched accuracy only evaluates whether or not the items in the cluster match the gold label. The non-matching items within a cluster might all be from a single second gold category, or they might be from many different categories. Intuitively, the former clustering should be evaluated as better, but matched accuracy is the same for both clusterings.

2.4.2 Variation of Information

Variation of Information (VI) (Meila, 2007) is a clustering evaluation measure that avoids the matching problem. It measures the amount of information lost and gained when moving between two clusterings C and K using conditional entropy, where K denotes the found clustering and C as the gold categories (since this measure is symmetric, the distinction is arbitrary). More precisely:

$$\begin{aligned} VI(C,K) &= H(C) + H(K) - 2I(C,K) && (2.35) \\ &= H(C|K) + H(K|C) && (2.36) \end{aligned}$$

A lower score implies closer clusterings, since each clustering has less unique information not shared with the other. Two identical clusterings have a VI of zero. However, VI's upper bound is dependent on the maximum number of clusters in C or K, making it again difficult to compare clustering results with different numbers of clusters. VI is closely related to other information-theoretic measures for clustering evaluation, including informativeness (Redington et al., 1998) and conditional entropy (Clark, 2003a).

2.4.3 Pairwise Precision and Recall

Pairwise Precision and Recall are widely used in the cognitive literature on category acquisition (e.g., Redington et al. (1998); Mintz (2003)), and are sometimes referred to as accuracy and completeness. To compute them, consider all possible word pairs. If the words in a pair are correctly grouped together by the model (i.e., they are in same gold-standard category), a true positive (*tp*) is recorded; if they are not in the same gold-standard category, a false positive (*fp*) is recorded. If the two words are not grouped together by the model, but are in the same gold-standard category, then a false negative (*fn*) is recorded. Pairwise precision and recall is then defined as:

$$PP = \frac{tp}{tp + fp} \quad PR = \frac{tp}{tp + fn} \tag{2.37}$$

Pairwise precision and recall suffers from a bias towards large clusters: words in larger clusters are members of more pairs, and hence get counted more often than words in smaller clusters. In the context of syntactic category induction, particularly when evaluating on types, this rewards models that do well on the broad productive categories (nouns, verbs), and does not penalise models as heavily for misclustering the smaller function word categories.

2.4.4 V-Measure

Finally, *V-measure* (VM; Rosenberg and Hirschberg (2007)) is our primary measure for evaluating clusterings. VM uses the conditional entropy of clusters and categories to evaluate clusterings, similar to VI. Like pairwise precision and recall, it has the useful characteristic of being analogous to the precision and recall measures commonly used in NLP. Homogeneity, the precision analogue, is defined as

$$VH = 1 - \frac{H(C|K)}{H(C)}.$$

VH is highest when the distribution of categories within each cluster is highly skewed towards a small number of categories, such that the conditional entropy is low. Completeness (recall) is defined symmetrically to VH as:

$$VC = 1 - \frac{H(K|C)}{H(K)}.$$

VC measures the conditional entropy of the clusters within each gold standard category, and is highest if each category maps to a single cluster so that each model cluster completely contains a category. The V-measure VM is simply the harmonic mean of VH and VC, analogous to traditional F-score, and as such, ranges from zero to one. Unlike MA and VI, VM is invariant with regards to both the number of items in the dataset and to the number of clusters used, and consequently it is best suited for comparing results across different corpora and models.

2.5 Conclusion

The Bayesian methods described in this chapter will provide the foundation for the models presented in the following chapters. These methods have been shown to be useful for modelling cognitive behaviours in other domains as well as language, giving us a general framework for proposing and evaluating rational models of cognition.

Chapter 3

Adding sentence-type labels to syntactic category acquistion models

The distributional hypothesis (Maratsos and Chalkley, 1980), which states that children can hypothesise syntactic categories on the basis of co-occurrence statistics in the input, has motivated previous models of syntactic category acquisition. These have demonstrated that distributional contexts, defined using only the closest surrounding items, can be sufficient as a starting point to learning reasonable syntactic categories.

However, non-local syntactic effects can influence local contexts. In this chapter, we add a form of non-local, sentence-level context to models using only local context, to investigate whether the added context improves performance. Specifically, we add *sentence type* as a known feature to the local context, i.e., whether the local context is within a question, declarative sentence, short fragment, etc. Sentence type often affects sentence structure and word order, and thereby can change the nature of local contexts. Taking sentence type into account may thus lead to clustering on the basis of more informative context distributions. An improvement in performance would indicate that this new information is useful to language learners, but it could also decrease performance if it is too noisy or does not correlate with syntactic category sequences.

Our enhanced models assume that children are aware of different sentence types and can make use of them at the stage of learning syntactic categories. A great deal of evidence from language development supports this assumption. Sentence types are strongly signalled by prosody in most languages (Hirst and Cristo, 1998). Prosody is, along with phonology, the first step in language learning; areas in the brains of three month olds are already sensitive to the prosody of the surrounding language (Homae et al., 2006) and experiments with newborn infants have demonstrated their ability to distinguish their native language using prosody alone (Mehler et al., 1988). Two month olds use prosody to remember heard utterances (Mandel et al., 1994). Significantly, Mandel et al. (1996) showed that natural sentence level prosody aided memory of word order in two month old infants, which is essential for remembering and using distributional information.

Infants are aided in language learning by the fact that intonation (pitch) contours of child and infant directed speech (CDS) are especially well differentiated between sentence types, more than in adult directed speech (Stern et al., 1982; Fernald, 1989). It is specifically the pitch contours of CDS that infants prefer over adult directed speech (Fernald and Kuhl, 1987) — the same contours that signal sentence type. CDS tends to be more interactive (as measured by the proportion of questions) than adult directed speech (Newport et al., 1977; Fernald and Mazzie, 1991), resulting in a greater variety of frequent sentential prosody patterns and potentially making sentential prosody a more salient feature at the beginning of language learning (Stern et al., 1983). Visual cues, particularly the speaker's facial expression, can also be used to distinguish between questions and statements (Srinivasan and Massaro, 2003).

Infants' awareness of sentence types can be demonstrated by their sensitivity to the pragmatic function signaled by sentence type. For example, mothers will stop asking questions if infants do not react appropriately, as when the mother is interacting with time-delayed video feed of the infant (Murray and Trevarthen, 1986). Since CDS is characterised by a high proportion of questions, this demonstrates that in normal caretaker-child interactions infants as young as three months are 'holding up' their side of the conversation in some basic sense. Infants produce appropriate intonational melodies to communicate their own intentions at the one word stage, before they develop productive syntax (Snow and Balog, 2002; Balog and Brentari, 2008; Galligan, 1987).

Based on these experimental results, we conclude that children who are at the point of learning syntax — at two to three years of age — are well equipped to use sentential prosody as part of their armory of potentially relevant input features. The experiments in this chapter investigate whether it would be advantageous for them to do so, given a classic computational model of syntactic category learning. To this end we annotate a corpus of child directed speech with sentence types, and extend the model to enable it to use these features.

We are not aware of previous work investigating the usefulness of sentence type information for syntactic category acquisition models. However, sentence types (identified by prosody) have been used to improve the performance of speech recognition systems. For example, Taylor et al. (1998) classified utterances into dialogue acts based on intonation. Using a specialised language model for each dialogue act enabled a significant decrease in word error rate for 'initiating' (non-response) dialogue acts.

In this chapter, we first examine the corpus data motivating our use of sentence types in syntactic category learning, and describe how we label sentence types. We then experiment with three different ways of incorporating sentence type into a token-based tagging model, the Bayesian Hidden Markov Model (BHMM). Our results demonstrate that sentence type is a beneficial feature for representing word order (or more precisely, syntactic category order).

3.1 Data

One of the first detailed investigations of CDS (Newport et al., 1977) found that it differs from adult directed speech in a number of key ways; for example, child directed utterances are significantly shorter and more intelligible than adult directed speech, with fewer false starts and corrections. This emphasizes the need to use realistic (i.e., CDS) corpora when modelling acquisition: the linguistic environment in which children acquire language is unlike the standard corpora used in computational linguistics.

More immediately relevant to our current work is the fact that CDS is far more heterogeneous in terms of sentence type than either adult written or spoken language. Whereas adult directed speech is largely made up of declarative utterances, CDS includes many more questions and imperative statements (Newport et al., 1977; Fernald and Mazzie, 1991). Indeed, one of the arguments for the utility of CDS (Gleitman et al., 1984) is that it is the range and the complexity of input that enables a learner to delimit the linguistic space, that is, to successfully separate grammatical sentences from non-grammatical. If a learner was given an overly constrained language to begin with, she could construct wrong hypotheses that would not admit the more complex adult language she would be faced with later on.

The data we we use come from CHILDES (MacWhinney, 2000), a collection of corpora shared by language development researchers. We use the Eve corpus (Brown, 1973) and the Manchester corpus (Theakston et al., 2001). The Eve corpus is a longitudinal study of a single US American child from the age of 1;6 to 2;3 years, whereas the Manchester corpus follows a cohort of 12 British children from the ages of 2 to 3. We remove all utterances with any unintelligible words or words tagged as `quote` (about

5% of all utterances and between 2 and 3% of CDS utterances). Corpus statistics are presented in Table 3.1. The Manchester corpus is over twenty times as large as the Eve corpus; by training and evaluating our models on both corpora we can investigate the effects of data set size.

Sentence types are not annotated in these corpora, so we use simple heuristics to label the sentences with their sentence type. The Cambridge Grammar of the English Language (Huddleston and Pullum, 2002) identifies the following five clause types:

Declarative *I have eaten the plums in the icebox.*

Closed Interrogative/*yes/no* questions *Were you saving them for breakfast?*

Open Interrogative/*wh*-questions *Why were they so cold?*

Exclamatory *What a sweet fruit!*

Imperative *Please forgive me!*

We use these clause types as a starting point. (To stress that we are labelling a full utterance/sentence, we will use the term *sentence type* rather than *clause type*.) We do not use the exclamatory sentence type due to its scarcity in the corpus; it is also difficult to identify automatically. Additionally, we add a short utterance category to distinguish probable fragments (verb-less clauses). The resulting sentence types with their identifying characteristics are:

Open Interrogative/*wh*-questions (W): Utterances ending with a question mark and beginning (in the first two words) with a *wh*-word (one of *who*, *what*, *where*, *when*, *why*, *how*, *which*).

Closed Interrogative/*yes/no* questions (Q): Utterances ending with a question mark but not beginning with a *wh*-word. This includes tag questions and echo questions with declarative (unmarked) word order.

Imperative (I): Utterances with an imperative-mood verb in the first two words[1].

Short (S): One- or two-word non-question utterances, typically interjections and fragments.

---

[1] Since the corpus we use, CHILDES, does not annotate the imperative mood in English, we use all utterances with a 'base' verb in the first two words without a pronoun or noun preceding it (e.g. *well go and get your telephone*).

	Eve		Manchester	
	All	CDS only	All	CDS only
Utterances	25295	14450	582375	318349
Mean Length	4.68	5.38	4.66	4.31
Word Tokens	118372	77766	2713672	1371936
Word Types	2235	1995	11739	11030

Table 3.1: Summary of all and child-directed speech (CDS) in Eve and Manchester corpora.

Sentence Type	Eve	Manchester
wh-Questions	2273 (4.03)	33461 (4.72)
Other Questions	2577 (4.41)	74327 (5.80)
Declaratives	6181 (5.79)	99318 (5.83)
Short Utterances	2752 (1.27)	95518 (1.28)
Imperatives	665 (5.30)	15725 (5.17)

Table 3.2: Number of child-directed utterances by sentence type. Average utterance length is in parentheses.

Declarative (D): All remaining utterances.

It would be preferable to use audio cues to categorise utterances, especially with regard to the difference between declaratives, closed interrogatives, and imperatives (since short utterances are easily identified by their length, and *wh*-words are a reliable cue for open interrogatives). Unfortunately the speech data is not available for our corpora, so we must approximate the audio cues available to children with the above orthographic (punctuation) and lexical-syntactic (*wh*-word and verb identification) cues. Note that the CHILDES annotation guidelines state that all utterances with question-characteristic intonation must be transcribed with a question mark, even if the utterance is syntactically declarative (*You ate the plums?*). In any case, even prosody data would not include all the cues available to the child such as visual cues, facial expressions, and so on.

Table 3.2 gives the number of each type of utterance in each corpus. Notably, while declaratives are the largest category, they make up only about a third of total utterances, while questions make up a further third of all utterances. (In contrast, questions make up only 3% of the Wall Street Journal corpus of news text (Marcus et al., 1999) and 7% of the Switchboard corpus of adult conversations (Godfrey et al., 1992).)

These sentence-labelling heuristics are rather coarse and prone to noise. Arguably, this is in line with the noisy input that children receive, presumably leading to errors on their part as well. Any model must be equally robust against noise and miscategorisation. We hand-checked a subset of the Eve (section 10, the dev section) to verify our sentence type heuristics. Of 497 utterances, only 10 (2%) were misclassified[2].

3.2 Models

Having motivated the salience of sentence type via intonation in the previous section, we now examine how to add sentence type as an observed variable to a simple part of speech induction model, the Bayesian Hidden Markov Model (BHMM) (Goldwater and Griffiths, 2007). The BHMM was introduced in Chapter 2. We now reiterate the model briefly in order to make the subsequent extensions, which include sentence type, more clear.

The BHMM is defined by two principal probability distributions: the transition distribution, which gives the probability of transitioning to a tag given the surrounding context (tags), and the emission distribution, which gives the probability of a particular word given its tag. The transition distributions τ generate the sequence of tags $\boldsymbol{t}$, and thus can represent a low-level, local syntax concerned with word order (but not movement or long distance dependencies). The Markov independence assumption limits the size of the context that is used for conditioning the transitions. In this chapter we present both bigram and trigram (first and second-order) models, which respectively use the one or two previous tags as context.

The emission distributions ω define the set of words W that are likely to appear as a given part of speech, i.e., the categorisation of the words into categories. Each of these distributions takes the form of a multinomial, over which we place a Dirichlet prior distribution, parameterised by the hyperparameters α (transition distributions) and β (emission distributions), resulting in the following generative model for a trigram BHMM:

---

[2] The misclassified sentences were: 8 imperatives classified as declaratives (mostly sentences of the form *Eve please listen*); one declarative sentence that is clearly a question in context but was not annotated with a question mark; one declarative sentence that was mistagged as an imperative due to a annotation error (*people* tagged as a verb).

$$\begin{aligned}
\tau_{(t',t'')}|\alpha &\sim \text{Dirichlet}(\alpha) && (3.1)\\
\omega_{(t)}|\beta &\sim \text{Dirichlet}(\beta) && (3.2)\\
t_i|t_{i-1}=t',t_{i-2}=t'',\tau_{(t',t'')} &\sim \text{Mult}(\tau_{(t',t'')}) && (3.3)\\
w_i|t_i=t,\omega_{(t)} &\sim \text{Mult}(\omega_{(t)}) && (3.4)
\end{aligned}$$

As we showed in Section 2.3.3, using Dirichlet priors allows the multinomial parameters τ and ω to be integrated out, leading to the following conditional posterior distributions:

$$P(t_i|\mathbf{t}^{\backslash i},\alpha) = \frac{n_{t_{i-2},t_{i-1},t_i}+\alpha}{n_{t_{i-2},t_{i-1}}+T\alpha} \qquad (3.5)$$

$$P(w_i|t_i,\mathbf{t}^{\backslash i},\mathbf{w}^{\backslash i},\beta) = \frac{n_{t_i,w_i}+\beta}{n_{t_i}+W_{t_i}\beta} \qquad (3.6)$$

where n_{t_i,w_i} is shorthand for $n_{t=t_i,w=w_i}$, the number of occurrences of the tag word pair (t,w) in $\mathbf{w}^{\backslash i}$, and likewise for n_{t_{i-2},t_{i-1},t_i}. The set $\mathbf{t}^{\backslash i}=t_1\dots t_{i-1}$ designates all tags but t_i, and likewise $\mathbf{w}^{\backslash i}=w_1\dots w_{i-1}$, all words but w_i. T is the size of the tagset and W_t is the number of word types emitted by the tag t.

3.2.1 BHMM with Sentence Types

The BHMM depends solely on local information — the previous tags — for tagging. However global information, such as sentence type, can play a role in syntax at the tag sequence level, by requiring shifts in word order. For this reason, adding sentence type to a part of speech tagger is likely to add helpful information, by enriching the impoverished local context representation.

In order to incorporate sentence type information into the BHMM, we add an observed variable to each time-step in the model with the value set to the current sentence type[3]. Given that the BHMM consists of two principal distributions, there are two straightforward ways that sentence type could be incorporated into the BHMM: either by influencing the transition probabilities or the emission probabilities. The former would reflect the effect of sentence type on word order, whereas the latter would investigate whether sentence type affects the set of words categorised as a single part of speech. We discuss both, as well as their combination.

---

[3] Arguably sentence type only needs to be included in the model once per sentence, rather than at each time-step, since sentence type never changes within a sentence. However, since sentence type is an observed variable, replicating it has no effect, and it makes the notation clearer.

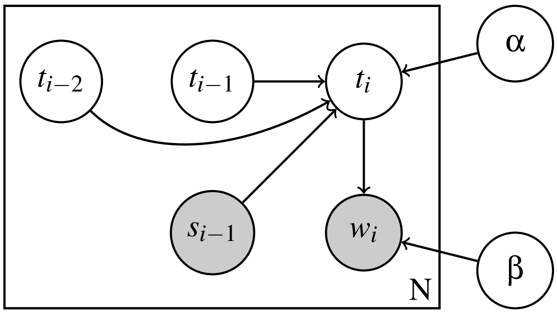

Figure 3.1: Graphical model representation of the BHMM-T, which includes sentence type as an observed variable on tag transitions (but not emissions).

3.2.1.1 BHMM-T

In the first case, transitions are conditioned not only on previous context, as in the BHMM, but also on the context's sentence type. This leads different sentence types to assign different probabilities to the same sequence of tags, so that, for example, `PRONOUN` will be more likely to be followed by a `VERB` in declaratives than in imperatives. (Note however that the estimated tag clusters will not necessarily correspond to gold tags.) By separating out the transitions, the model will have more flexibility to accommodate word order changes between sentence types.

Formally, the observed sentence type s_{i-1} is added as a conditioning variable when choosing t_i, i.e., we replace line 3.3 from the BHMM definition with the following:

$$t_i | s_{i-1} = s, t_{i-1} = t', t_{i-2} = t'', \tau_{(s,t',t'')} \sim \text{Mult}(\tau_{(s,t',t'')}) \tag{3.7}$$

We refer to this model, illustrated graphically in Figure 3.1, as the BHMM-T (for transitions).

The BHMM-T has a larger number of parameters than the BHMM, which has $T^{o+1} + TV$ parameters (where T is the number of tags, o is the model order, and V is the size of the vocabulary), whereas the BHMM-T has $ST^{o+1} + TV$ (S being the number of sentence types)[4].

---

[4] Since probability distributions are constrained to sum to one, the last parameter in each distribution is not a free variable, and so the true number of necessary emission parameters is $T(V-1)$ (and likewise for transition parameters), but we omit this technicality in favor of clarity.

3.2.1.2 BHMM-E

Analogously, we can add sentence type as a conditioning variable in the emission distribution by replacing line 3.4 from the BHMM with

$$w_i | s_i = s, t_i = t, \omega_{(s,t)} \sim \text{Mult}(\omega_{(s,t)}) \tag{3.8}$$

This model, the BHMM-E (for emissions), results in models in which each sentence type has a separate distribution of probable words for each tag, but the transitions between those tags are shared between all sentence types, as in the BHMM. This does not correspond well to the word-order effect that sentence type has in many languages, but may capture vocabulary differences between sentence types, if these exist.

The model size is $T^{o+1} + STV$, which in practice is significantly larger than the BHMM-T model, given $V \gg T > S$ and model orders of one and two (bigram and trigram models).

3.2.1.3 BHMM-ET

The combination of the two, BHMM-T plus BHMM-E, is also possible. In the BHMM-ET, sentence type conditions both transition and emission probabilities. Each sentence type now has a separate set of transition and emission distributions (both transitions and emissions are conditionally independent given sentence type). Without any shared information, tags are not in any sense equivalent between sentence types, so this model is equivalent to training a separate BHMM on each type of sentence, albeit with shared hyperparameters.

These three models have an extra conditioning variable, sentence type, which has the effect of splitting the counts for transitions, emissions, or both. The split distributions will therefore be estimated using less data, which could degrade performance if sentence type is not a useful predictor of tag sequences or tag-word pairings. This will be especially vital to the performance of the BHMM-ET, without any shared information at all. If the separate models in the BHMM-ET match the BHMM's performance, this would indicate that sentence type is as reliable an indicator of tagging information as a large amount of additional data from other sentence types. However, it is cognitively implausible for there to be no sharing of information at all between sentence types: this model serves principally as a measure of sentence type informativeness.

Our prediction is that sentence type is more likely to be useful as a conditioning variable for transition probabilities (BHMM-T) than for emission probabilities

(BHMM-E). For example, the auxiliary inversion in questions is likely to increase the probability of the `AUX` → `PRONOUN` transition, compared to declaratives. Knowing that the sentence is a question may also affect emission probabilities, e.g. it might increase the probability the word *you* given a `PRONOUN` and decrease the probability of *I*; one would certainly expect *wh*-words to have much higher probability in *wh*-questions than in declaratives. However, many other variables also affect the particular words used in a sentence (principally, the current semantic and pragmatic context). We expect that sentence type plays a relatively small role compared to these other factors. The ordering of tags within an utterance, on the other hand, is primarily constrained by sentence type, especially in the short and grammatically simple utterances found in child-directed speech.

3.3 English experiments: Procedure

3.3.1 Corpora

We use the Eve and Manchester corpora from CHILDES, described in Section 3.1, for our experiments. From both corpora we remove all utterances spoken by a child; the remaining utterances are nearly exclusively CDS. Although our model is fully unsupervised, files from the chronological middle of each corpus are set aside for development and testing evaluation[5]. The BHMM is inferred using either (train+dev) or (train+test). In the Eve corpus, we set use file 10 for development, 11 for testing; in Manchester: file 16 from each child for development, file 17 for testing. When evaluating initial experiments using the development data (dev+train), we evaluate only on the dev portion. Final evaluations using the test portion are evaluated on the full train+test set. Note that although all models have the training portion (approximately 80% of the entire data set) available as unlabelled data, we only evaluate the training portion in the final test evaluation[6].

Both corpora have been tagged using the relatively rich CHILDES tagset, which we

---

[5] Unsupervised models do not suffer from overfitting in the same way as supervised models, so often models are trained and results reported on the entire set of input data. However, since different model structures and parameterisations may be explored during development, this methodology still leaves open the possibility that the final model structure may be better suited to the particular corpus being used than to others. To avoid this issue, we use separate test and development sets.

[6] The term ‘training data’ may be a misnomer: the unsupervised models have no *labelled* training data available. We used the name to designate the largest chunk of the data, given the usual training/dev/test split design common to (supervised) NLP. Because the data is unlabelled, we can ‘test’, i.e. evaluate, the ‘training’ data.

collapse to a smaller set of thirteen tags: adjectives, adverbs, auxiliaries, conjunctions, determiners, infinitival-*to*, nouns, negation, participles, prepositions, pronouns, verbs and other (communicators, interjections, fillers and the like). *wh*-words are tagged as adverbs (*why*, *where*, *when* and *how*), pronouns (*who* and *what*), or determiners (*which*).

Each sentence is labelled with its sentence type using the heuristics described earlier. Dummy inter-sentence markers are added, so transitions to the beginning of, as well as from the end of, a sentence will be from the fixed inter-sentence hidden state. The inter-sentence markers are assigned a separate dummy sentence type.

In our experiments, we experiment with coarser sentence type categorisations as well as the full five types. This enables us to discover which sentence types are most informative for the tagging task. Specifically, we try:

QD in which all questions (*wh*- and other) are collapsed to one question category and all other utterances are collapsed to declaratives.

WQD in which the question categories are separated but the non-questions are in a single category.

SWQD as above, but with short (declarative) utterances distinguished from other non-question utterances.

ISWQD in which all five sentence types are separated: imperatives, short utterances, *wh*-questions, other questions, and declaratives.

3.3.2 Inference and evaluation procedure

We use the Gibbs sampler described in Section 2.3.5 for BHMM inference, adjusting the tracked counts (the sufficient statistics) as necessary for the BHMM variants. The trigram Gibbs sampling equations for each of the models are thus:

$$P_{BHMM}(t|\mathbf{t}^{\setminus i},\mathbf{w},\alpha,\beta_t) \propto \frac{n_{t,w_i}+\beta_t}{n_t+V\beta_t} \times \frac{n_{t_{i-2},t_{i-1},t}+\alpha}{n_{t_{i-2},t_{i-1}}+T\alpha} \tag{3.9}$$

$$P_{BHMM-T}(t|\mathbf{t}^{\setminus i},\mathbf{w},\mathbf{s},\alpha,\beta_t) \propto \frac{n_{t,w_i}+\beta_t}{n_t+V\beta_t} \times \frac{n_{t_{i-2},t_i,s_{i-1},t_i}+\alpha}{n_{s_{i-1},t_{i-2},t_{i-1}}+T\alpha} \tag{3.10}$$

$$P_{BHMM-E}(t|\mathbf{t}^{\setminus i},\mathbf{w},\mathbf{s},\alpha,\beta_t) \propto \frac{n_{s_i,t,w_i,}+\beta_t}{n_{s_i,t}+V\beta_t} \times \frac{n_{t_{i-2},t_{i-1},t}+\alpha}{n_{t_{i-2},t_{i-1}}+T\alpha} \tag{3.11}$$

$$P_{BHMM-ET}(t|\mathbf{t}^{\setminus i},\mathbf{w},\mathbf{s},\alpha,\beta_t) \propto \frac{n_{s_i,t,w_i,}+\beta_{t_i}}{n_{s_i,t}+V\beta_t} \times \frac{n_{t_{i-2},t_{i-1},s_{i-1},t}+\alpha}{n_{s_{i-1},t_{i-2},t_{i-1}}+T\alpha} \tag{3.12}$$

Transition factors for trigrams (t_{i-1}, t, t_{i+1}) and (t, t_{i+1}, t_{i+2}) are not shown but must be included (see Equation 2.34; bigrams have analogous factors).

The hyperparameters over emission distributions, β, is estimated separately for each distribution (i.e., each tag has a separate β_t), whereas α is constrained to be the same for all transition distributions. The hyperparameters α and β are re-estimated every ten iterations using a Metropolis-Hastings step, following Goldwater and Griffiths (2007).

Simulated annealing is used to encourage the sampler to explore during the first phase of the sampler and thereafter 'exploit' a high-likelihood area of the search space. During the last few iterations, the temperature is lowered still further, resulting in a final sample that is a fair approximation of the nearest local maximum. This final sample is used for evaluation purposes (see Section 2.3.6).

We set the number of hidden states, corresponding to tag clusters, to the number of gold tags in the corpus (i.e., 13).

We run the Gibbs sampler for 10000 iterations, with hyperparameter resampling and simulated annealing. (On the Manchester corpus, due to its size, we only manage 5000 iterations.) Since Gibbs sampling is a stochastic algorithm, we run all models ten times and report average values for all evaluation measures as well as confidence intervals at the 95% level. Significance is measured using the non-parametric Wilcoxon signed-rank test. The principal evaluation measure we use is VM, described in Section 2.4.

3.4 Results and Discussion

We now present results for the three BHMM variants discussed in the previous section: the BHMM-ET, a model in which both transitions and emission distributions are separated by sentence type; the BHMM-E, a model in which only emission distributions are separated by sentence type and transitions are shared; and the converse, the BHMM-T, in which transitions are separated and emission distributions are shared. We find that these models perform in very different ways, demonstrating the effect of sentence type on word order rather than word usage.

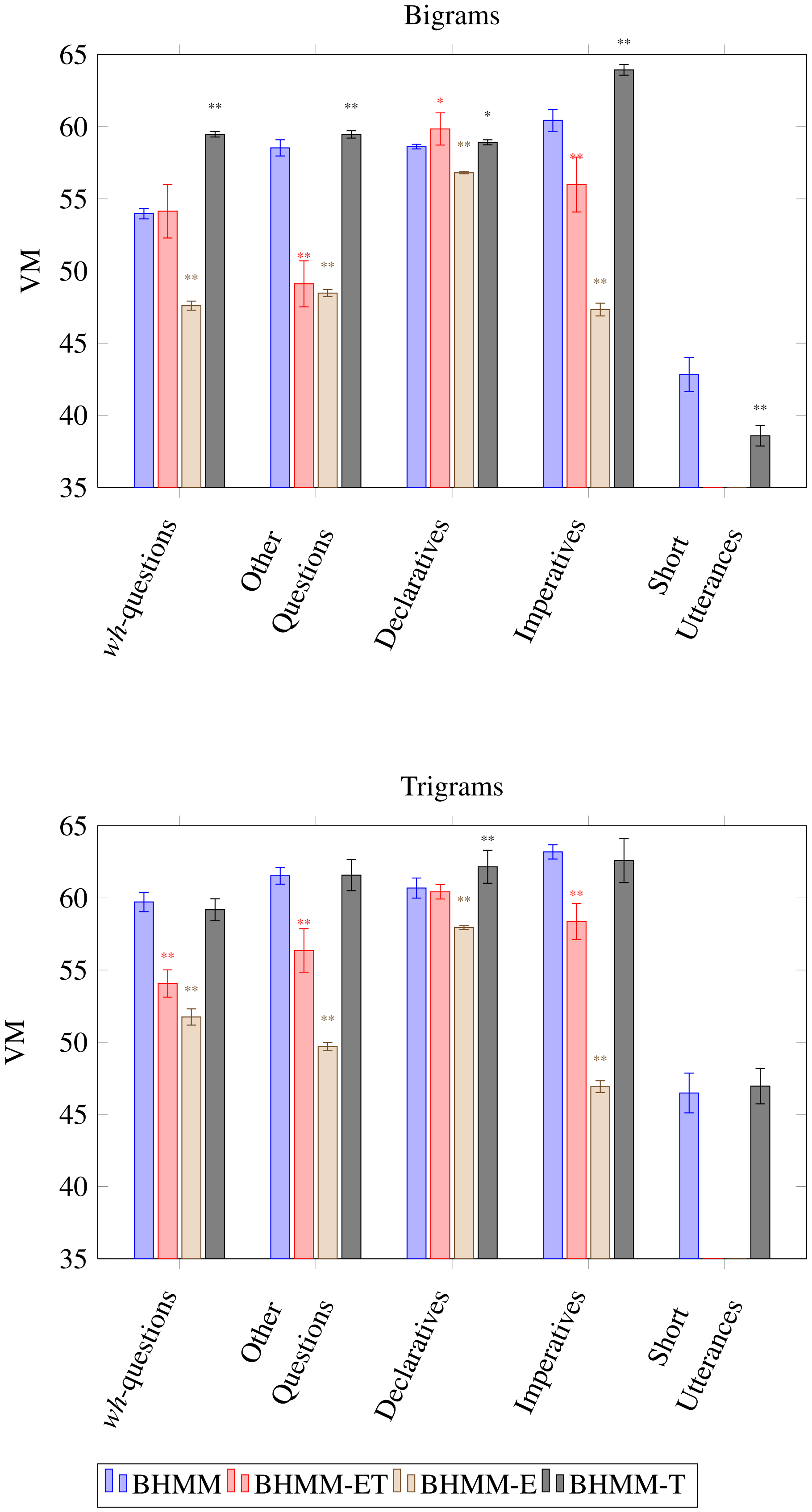

Figure 3.2: Performance of bigram and trigram BHMM and variants by sentence type on the Eve corpus, evaluated on test+train portion: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

3.4.1 BHMM-ET: sentence-type-specific sub-models

By including a sentence type indicator in both the transition and emission distributions, the BHMM-ET separates both transition and emission probabilities by sentence type, effectively creating separate sub-models for each sentence type. The resulting tags are thus not equivalent between sentence types, i.e., the tag `TAG-9` used in a declarative sentence is not the ‘same’ `TAG-9` as used in questions, since they have distinct transition and emission distributions. Consequently, when evaluating the tagged output each sentence type must be evaluated separately, to avoid conflating incompatible clusterings.

Baseline BHMM and BHMM-ET results for the Eve corpus, split by sentence type, are in Figure 3.2 (test+train datasets). This figure shows results for all four models; at present we only examine the first two. We will return to this figure in the discussion of the BHMM-E and BHMM-T models shortly. We see that in this relatively small corpus, as expected, splitting the sentence types results in decreased performance as compared to the baseline BHMM, in which all counts are shared. Only the declaratives, the most frequent sentence type, provide enough information on their own to match and exceed baseline performance. (Bigram *wh*-questions also perform equally to the baseline, but with much higher variance over ten runs.)

Figure 3.3 shows results for the much larger Manchester corpus. The BHMM-ET models here perform much closer to the baseline, due to the larger amount of data available to each sentence type sub-model. Each of the sentence types contain enough information to learn approximately equivalent taggers using either only single sentence type data or the full data set. *Wh*-questions suffer a slight but significant performance drop compared to the baseline, as do other questions in the trigram models; however performance is still similar.

The real exceptions are the short utterances models, which suffer from the lack of sufficiently informative contexts. Since all short utterances are one or two words long, there is a limited amount of context to identify clusters. While the vast majority of single word utterances are interjections assigned to the `OTH` tag, and these are for the most part clustered together, fragments of longer utterances are almost impossible to tag correctly without the informative distributional contexts from longer sentences. Lacking this information, there is no basis for assigning a different set of tags to *now then* than *just salami*. This effect is also evident in the short utterances in the Manchester corpus, despite the fact that the number of short utterance words is much larger:

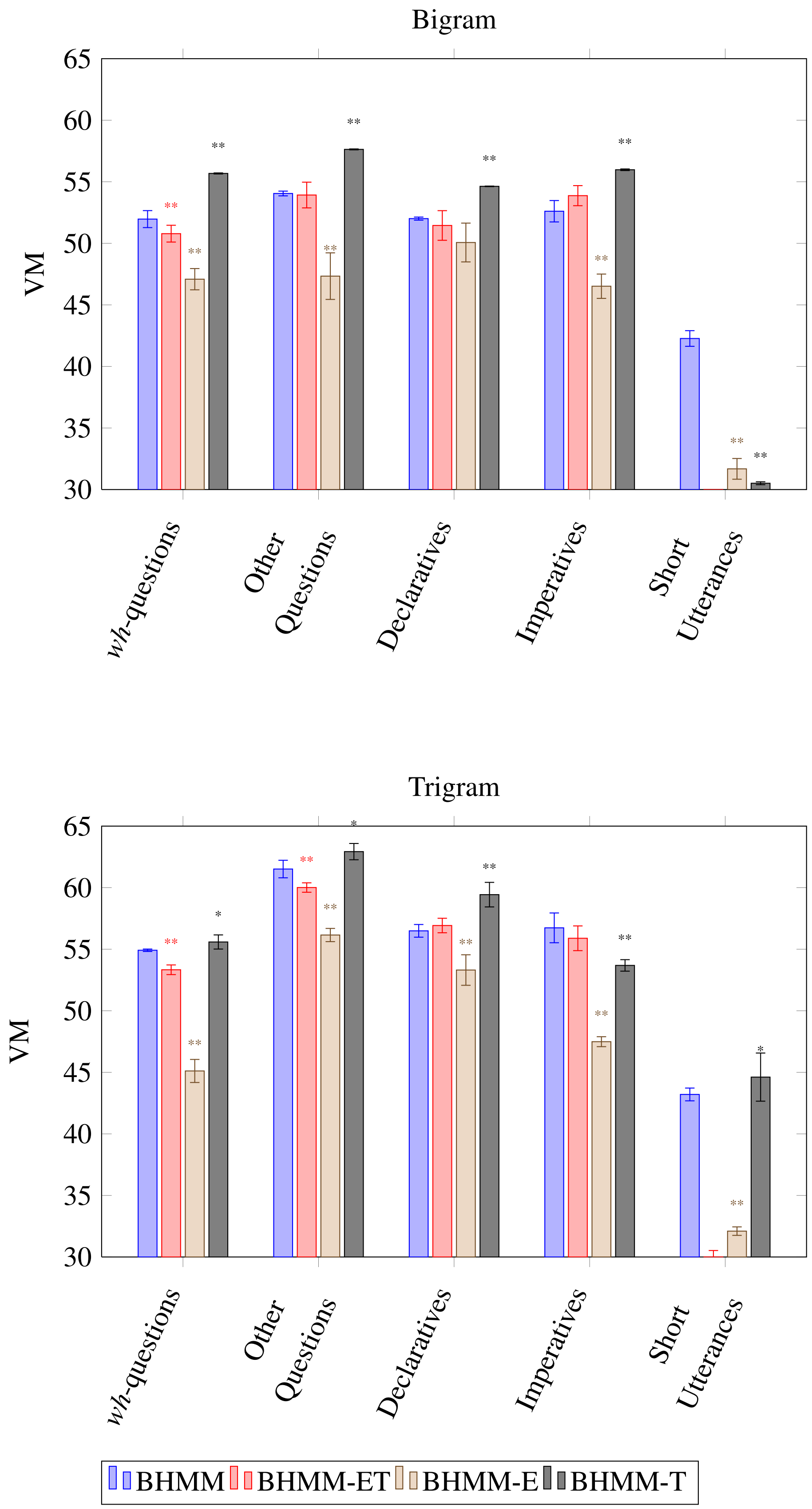

Figure 3.3: Performance of bigram and trigram BHMM and variants by sentence type on the Manchester corpus, evaluated on test+train portion: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

the required context is still missing.

3.4.2 BHMM-E: sentence-type-specific emissions

We now turn to the BHMM-E, in which emission probability distributions are sentence-type-specific, but transition probabilities are shared between all sentence types. In this model a given sequence of tags is equally likely among all sentence types, but those tags can correspond to different words in different sentence types.

Returning to Figures 3.2 (Eve corpus) and Figures 3.3 (Manchester corpus), we see that for almost every sentence type, the BHMM-E performs drastically worse than both the BHMM-ET and the baseline BHMM. The negative effect of the split emissions is most striking on the Manchester corpus, where small dataset size cannot be a problem. Whereas with the BHMM-ET we might posit that, given enough data, sentence-type-specific models would learn an equivalent model to the shared baseline model (apart from the short utterance distinction), here we see that adding the sentence type feature to only the emissions is actively harmful.

3.4.3 BHMM-T: sentence-type-specific transitions

Lastly, we evaluate the BHMM-T, which shares emission probabilities among sentence types and uses sentence-type-specific transition probabilities. This describes a model in which all sentence types use the same set of tags, but those tags can appear in different orders in different sentence type. Since this situation corresponds best to the behavior of sentence type in English and many other languages — word order changes according to sentence type, but word usage does not — we expect the BHMM-T to perform the best of all the sentence type BHMM models, and to improve over the baseline BHMM.

In the bigram models in both corpora we see the BHMM-T improve upon the baseline BHMM in all sentence types apart from short utterances. Dataset size makes a larger difference for trigram models: the Manchester-trained models outperform the baseline consistently, whereas the Eve-trained trigram models only match the baseline, apart from declaratives, the most frequent sentence type, which significantly improve over the baseline. On the other hand, the least frequent sentence type, imperatives, is the only sentence type to underperform the baseline in the Manchester setting.

Trigram models must estimate significantly more parameters than bigram models; adding sentence type increases the number of parameters still further. Where BHMM-T matches and exceeds baseline performance, adding sentence type is creating split transition distributions that are more or equally accurate to the fully shared distribution, despite being estimated on far less data. This demonstrates the potential effectiveness of sentence type information for transitions.

Not all sentence types seem to be useful: bigram models perform poorly on short utterances, due to the limited context. The shared transitions in the baseline BHMM contains necessary information for accurate tagging of short utterances, whereas the BHMM-T transitions do not. Rare sentence types such as imperatives are difficult to learn, particularly for trigam models. These differences indicate that it may be advantageous to only distinguish certain sentence types.

Figure 3.4 (Eve) and Figure 3.5 (Manchester) show the results of BHMM-T models trained on data sets using a variety of less fine-grained sentence types, as described earlier. The shared emission distributions in the BHMM-T allow us to evaluate the corpus as a whole, using a single clustering for all sentence types. We also show results obtained using other evaluation metrics in Table 3.3 and Table 3.4.

Examining the sentence type variants, on the Eve datasets we see bigram models do best with data annotated with three sentence types (WQD), although all sentence type splits improve over the baseline. Trigrams benefit most from a simpler two-way question/other (QD) distinction. On the Manchester corpus, bigram models also perform best with the WQD split, but again nearly all bigram BHMM-T models improve significantly over the baseline, regardless of sentence type distinctions. Trigram BHMM-T are less consistently better than the baseline, but also do not perform worse. The best performance is achieved when using the full set of sentence types (ISWQD).

The other evaluation metrics in Table 3.3 and Table 3.4 generally agree with VM as to the best model, especially in the trigram models, despite the small relatively differences between the models' results. This gives us additional confirmation as to the effect of the different model structures on the found clusterings.

When trained on the Eve corpus, the trigram BHMM-T does not have sufficient data to accurately infer categories when the transitions are split between too many sentence types, and performs best with only two sentence types. On the other hand, when more data is available, the trigram BHMM-T is able to estimate good parameters for all five sentence types: the models trained on the Manchester corpus with all sentence types outperform all the others. The improvement over the baseline and other models is slight, however, indicating that sentence type is providing minimal additional information in this case.

However, it is important to note that even in cases where sentence type does not

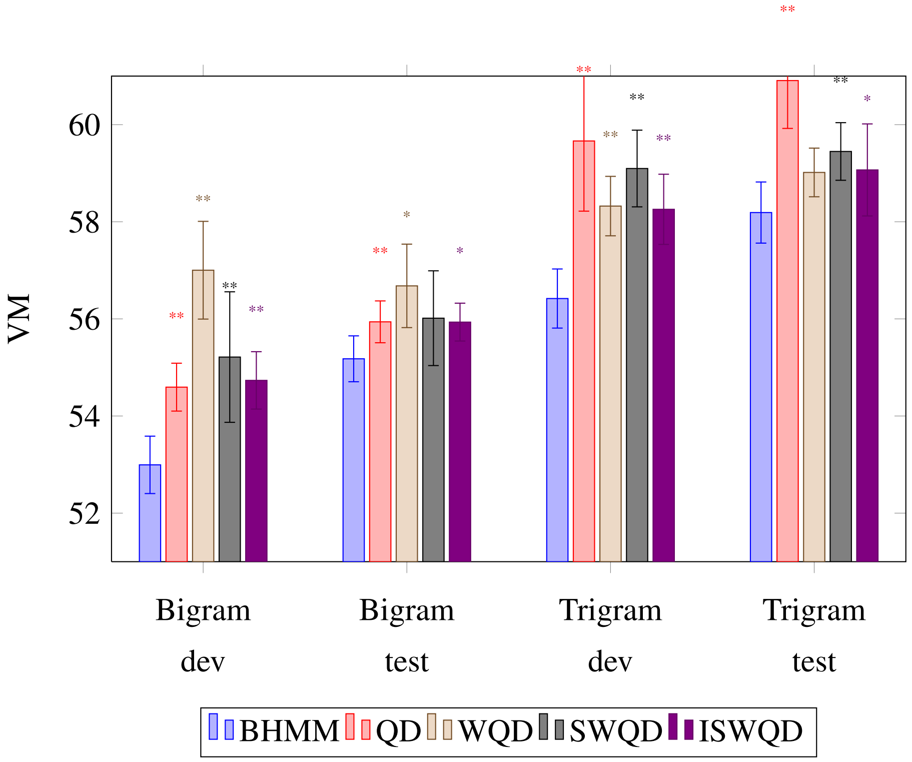

Figure 3.4: BHMM-T performance on the Eve corpus: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

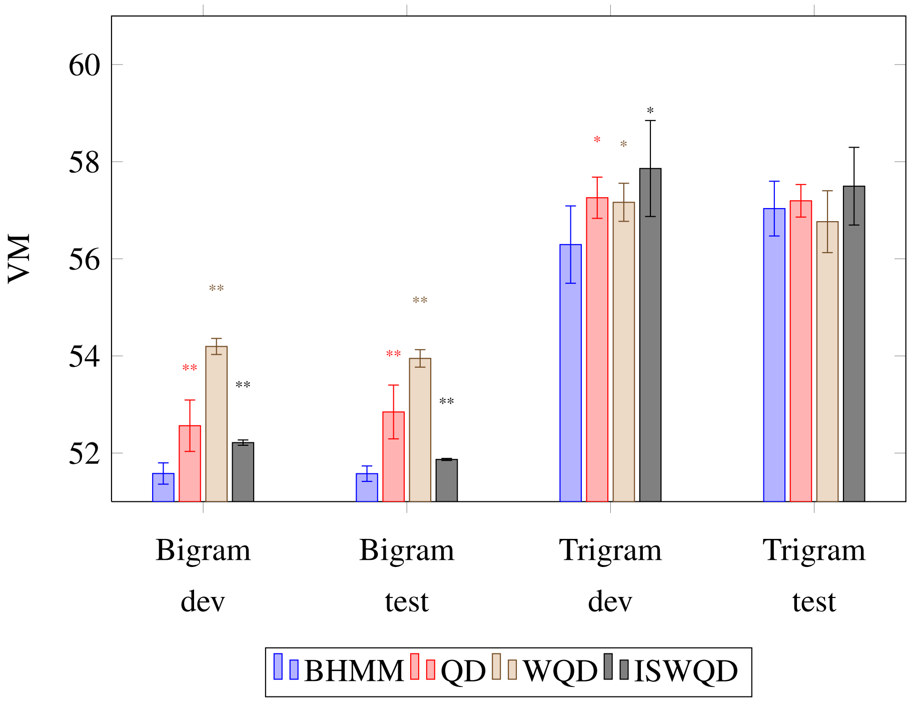

Figure 3.5: BHMM-T performance on the Manchester corpus: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

	VM	VH	VC	VI	M1	11
Bigram - BHMM	55.2 [54.7, 55.7]	58.0 [57.6, 58.4]	52.6 [52.1, 53.2]	3.08 [3.04, 3.12]	70.0 [69.6, 70.3]	43.3 [41.3, 45.4]
QD	55.9 [55.5, 56.4]	59.1 [58.7, 59.4]	53.1 [52.7, 53.6]	3.04 [3.01, 3.08]	69.9 [69.7, 70.0]	41.2 [40.3, 42.1]
WQD	56.7 [55.8, 57.5]	59.7 [58.9, 60.5]	53.9 [53.0, 54.9]	2.98 [2.92, 3.05]	69.5 [69.0, 70.0]	40.9 [39.4, 42.3]
SWQD	56.0 [55.0, 57.0]	59.0 [58.0, 59.9]	53.3 [52.3, 54.3]	3.03 [2.96, 3.10]	69.8 [68.6, 71.0]	40.6 [39.3, 42.0]
ISWQD	55.9 [55.5, 56.3]	59.1 [58.6, 59.5]	53.1 [52.8, 53.5]	3.04 [3.02, 3.07]	70.5 [69.7, 71.3]	37.9 [36.5, 39.3]
Trigram - BHMM	58.2 [57.6, 58.8]	61.5 [60.8, 62.2]	55.2 [54.6, 55.8]	2.89 [2.85, 2.93]	71.3 [70.3, 72.2]	40.7 [38.7, 42.7]
QD	60.9 [59.9, 61.9]	64.4 [63.4, 65.4]	57.8 [56.8, 58.8]	2.70 [2.63, 2.77]	74.2 [72.7, 75.7]	42.8 [39.6, 46.0]
WQD	59.0 [58.5, 59.5]	62.5 [61.9, 63.0]	55.9 [55.5, 56.4]	2.84 [2.80, 2.87]	70.4 [70.0, 70.7]	40.0 [37.8, 42.2]
SWQD	59.5 [58.9, 60.0]	63.0 [62.4, 63.5]	56.3 [55.7, 56.9]	2.81 [2.76, 2.85]	71.8 [70.7, 72.9]	42.8 [41.6, 44.0]
ISWQD	59.0 [58.1, 60.0]	62.5 [61.5, 63.5]	56.0 [55.1, 56.9]	2.83 [2.76, 2.90]	72.1 [70.8, 73.4]	42.2 [39.9, 44.5]

Table 3.3: Mean BHMM-T performance on the Eve corpus ($N = 10$) with 95% Confidence Intervals in brackets.

Evaluation measure are: V-measure (VM), V-homogenity (VH), V-completeness (VC), Variation of Information (VI), Many-to-One (M1) and One-to-One (11) matched accuracy.

	VM	VH	VC	VI	M1	11
Bigram - BHMM	51.6 [51.4, 51.7]	53.8 [53.6, 54.0]	45.0 [49.4, 49.7]	3.38 [3.36, 3.39]	65.4 [64.9, 65.8]	42.5 [41.5, 43.5]
QD	52.9 [52.3, 53.4]	54.8 [54.3, 55.3]	51.0 [50.4, 51.6]	3.27 [3.23, 3.31]	65.5 [64.7, 66.4]	44.2 [42.9, 45.5]
WQD	54.0 [53.8, 54.1]	55.8 [55.6, 56.0]	52.2 [52.0, 52.4]	3.18 [3.17, 3.20]	65.0 [64.4, 65.6]	44.9 [42.8, 47.0]
ISWQD	51.9 [51.9, 51.9]	53.7 [53.7, 53.8]	50.1 [50.1, 50.1]	3.33 [3.33, 3.33]	63.1 [63.1, 63.1]	42.2 [41.0, 43.3]
Trigram - BHMM	57.0 [56.5, 57.6]	59.5 [58.8, 60.1]	54.8 [54.3, 55.3]	2.99 [2.96, 3.03]	69.7 [69.1, 70.4]	46.7 [44.1, 49.2]
QD	57.2 [56.9, 57.5]	59.6 [59.2, 60.0]	55.0 [54.6, 55.3]	2.98 [2.96, 3.01]	69.7 [69.6, 69.9]	45.5 [43.4, 47.5]
WQD	56.8 [56.1, 57.4]	59.3 [58.6, 60.0]	54.5 [53.9, 55.1]	3.02 [2.97, 3.06]	69.6 [68.8, 70.5]	46.2 [44.5, 47.8]
ISWQD	57.5 [56.7, 58.3]	59.8 [59.0, 60.7]	55.4 [54.6, 56.1]	2.96 [2.90, 3.01]	70.5 [69.5, 71.5]	48.1 [46.0, 50.2]

Table 3.4: Mean BHMM-T performance on Manchester corpus ($N = 10$) with 95% Confidence Intervals in brackets. Evaluation measure are: V-measure (VM), V-homogenity (VH), V-completeness (VC), Variation of Information (VI), Many-to-One (M1) and One-to-One (11) matched accuracy.

seem to add additional information in the transition distribution, it never decreases performance (as it did when added to the emission distribution). This indicates that at worst, sentence type carries the same information (but no more) as the context history already available in the BHMM. However, in some scenarios, as when evaluated on the Eve development corpus, the bigram model with the best set of sentence type labels (WQD) performs as well as the trigram model without sentence types. In this case, sentence-type-specific bigram transitions are as informative as transitions with twice as much local context information, leading to a model with fewer parameters but equal performance.

In summary, based on experiments using English corpora we have found that separate emission distributions between sentence type are harmful (BHMM-ET and BHMM-E), whereas separate transitions for sentence types may be helpful. This is in line with our predictions, based on the fact that in English sentence types primarily affect word order.

Cognitively, separate emission distributions would be hard to justify, since they result in non-corresponding syntactic categories between sentence types. In these models, each sentence type has a separate set of syntactic categories, which means that e.g. *cat* and *mouse* must be clustered together separately for each sentence type. Such models, in which categories are replicated multiple times and differ between sentence types, clearly do not make efficient use of limited input data.

Unlike the words making up syntactic categories, word order does change between sentence types in many languages, and taking this into account by learning separate word orders for each sentence type seems to be an effective strategy. Here we found that the choice of sentence type categories matters, and is dependent on the amount of input data available: with larger amounts of data, finer sentence type categories can be used.

3.5 Crosslinguistic Experiments with BHMM-T

In the previous section we found that sentence type information improved syntactic categorisation in English. In this section, we evaluate the BHMM's performance on two languages other than English, and investigate whether sentence type information is useful across languages.

Nearly all human languages distinguish between closed *yes/no*-questions and declaratives in intonation. Closed questions are most commonly marked by rising into-

Sentence Type	Spanish (Ornat)	Cantonese (LWL)
Total	8759 (4.29)	12544 (4.16)
wh-Questions	1507 (3.72)	2287 (4.80)
Other Questions	2427 (4.40)	3568 (4.34)
Declaratives	4825 (4.41)	6689 (3.85)

Table 3.5: Number of child-directed utterances by sentence type in the Spanish and Cantonese training sets. Average utterance length is in parentheses.

nation (Hirst and Cristo, 1998). *Wh*-questions do not always have a distinct intonation type, but they are signalled by the presence of members of the small class of *wh*-words.

We use tagged corpora for Spanish and Cantonese from the CHILDES collection: the Ornat corpus (Ornat, 1994) and the Lee Wong Leung (LWL) corpus (Lee et al., 1994) respectively. We describe each corpus in turn below. Table 3.5 lists their relative sizes.

3.5.1 Spanish

The Ornat corpus is a longitudinal study of a single child between the ages of one and a half and nearly four years, consisting of 17 files. Files 08 and 09 are used for testing and development. We collapse the Spanish tagset used in the Ornat corpus in a similar fashion to the English corpora. There are 11 tags in the final set: adjectives, adverbs, conjuncts, determiners, nouns, prepositions, pronouns, relative pronouns, auxiliaries, verbs, and other.

Spanish *wh*-questions are formed by fronting the *wh*-word (but without the auxiliary verbs added in English); *yes/no*-questions involve raising the main verb (again without the auxiliary inversion in English). Spanish word order in declaratives is generally freer than English word order. Verb- and object-fronting is more common, and pronouns may be dropped (since verbs are marked for gender and number). Note that verb-fronted declaratives will have the same structure as closed questions. This suggests that there will be fewer reliable differences between transition distributions in the various sentence types.

3.5.2 Cantonese

The LWL corpus consists of transcripts from a set of eight children followed over the course of a year, totalling 128 files. The ages of the children are not matched, but they range between one and three years old. Our training set consists of the first 500 utterances of all training files, in order to create a data set of similar size as the other corpora used. Files from children aged two years and five months are used as the test set; files from two years and six months make up the development set.

The tagset used in the LWL, which we use directly, is larger than the collapsed English tagset. It consists of 20 tags: adjective, adverb, aspectual marker, auxiliary or modal verb, classifier, communicator, connective, determiners, genitive marker, preposition or locative, noun, negation, pronouns, quantifiers, sentence final particle, verbs, *wh*-words, foreign word, and other. We remove all sentences that are encoded as being entirely in English but leave single foreign, mainly English, words (generally nouns) in a Cantonese context.

Cantonese follows the same basic SVO word order as English, but with a much higher frequency of topic-raising. Questions are not marked by different word order. Instead, particles are inserted to signal questioning. These particles can signal either a yes/no-question or a *wh*-question; in the case of *wh*-questions they replace the item being questioned (e.g., *playing-you what?*), without *wh*-raising as in English or Spanish. Strictly speaking the only syntactic change in transitions would thus be an increase in transitions to and from the *wh*-particles in questions. However, there may be other systematic differences between questions and declaratives.

3.5.3 Results

We trained BHMM and BHMM-T models in the same manner as with the English corpora (10 runs each, 10000 iterations, with simulated annealling and hyperparameter estimation).

Due to inconsistent annotation and lack of familiarity with the languages, we used only three sentence types: open/*wh*-questions, other questions, and declaratives. Punctuation was used to distinguish between questions and declaratives. *wh*-questions were identified by using a list of *wh*-words for Spanish; the Cantonese corpus included a *wh*-word tag.

In English, the BHMM-T was able to improve performance by taking into account the distinct word orders characteristic of the different sentence types. Spanish does

	VM	VH	VC	VI	M1	11
Bigram - BHMM	41.5 [41.2, 41.9]	43.4 [43.1, 43.7]	39.9 [39.4, 40.3]	3.76 [3.73, 3.80]	54.3 [53.9, 54.6]	13.0 [10.7, 15.3]
WQD	40.9 [36.9, 44.8]	43.0 [38.8, 47.2]	38.9 [35.1, 42.7]	3.84 [3.58, 4.10]	54.4 [50.4, 58.4]	12.4 [9.87, 14.9]
Trigram - BHMM	45.1 [44.3, 45.9]	47.6 [47.0, 48.2]	42.9 [42.0, 43.8]	3.57 [3.51, 3.64]	57.9 [57.6, 58.2]	14.0 [11.3, 16.6]
WQD	44.9 [41.9, 47.9]	47.2 [43.9, 50.5]	42.8 [40.2, 45.5]	3.57 [3.39, 3.75]	58.6 [54.1, 63.2]	13.0 [10.8, 15.2]

Table 3.6: Mean BHMM-T performance on the Spanish (Ornat) corpus ($N = 10$) with 95% Confidence Intervals in brackets. Evaluation measure are: V-measure (VM), V-homogenity (VH), V-completeness (VC), Variation of Information (VI), Many-to-One (M1) and One-to-One (11) matched accuracy.

	VM	VH	VC	VI	M1	11
Bigram - BHMM	46.1 [45.3, 46.9]	51.5 [50.6, 52.5]	41.7 [41.0, 42.4]	4.00 [3.95, 4.06]	64.4 [63.6, 65.2]	15.4 [13.3, 17.5]
WQD	47.4 [46.6, 48.1]	53.5 [52.6, 54.4]	42.5 [41.8, 43.2]	3.95 [3.89, 4.01]	66.5 [65.7, 67.2]	15.6 [14.5, 16.8]
Trigram - BHMM	49.0 [47.9, 50.0]	55.0 [53.8, 56.2]	44.1 [43.2, 45.1]	3.81 [3.73, 3.89]	66.9 [64.9, 69.0]	15.5 [13.4, 17.6]
WQD	50.7 [49.8, 51.6]	57.2 [56.1, 58.3]	45.5 [44.8, 46.2]	3.70 [3.64, 3.76]	69.0 [67.7, 70.4]	18.2 [16.9, 19.5]

Table 3.7: Mean BHMM-T performance on the Cantonese (LWL) corpus ($N = 10$) with 95% Confidence Intervals in brackets. Evaluation measure are: V-measure (VM), V-homogenity (VH), V-completeness (VC), Variation of Information (VI), Many-to-One (M1) and One-to-One (11) matched accuracy.

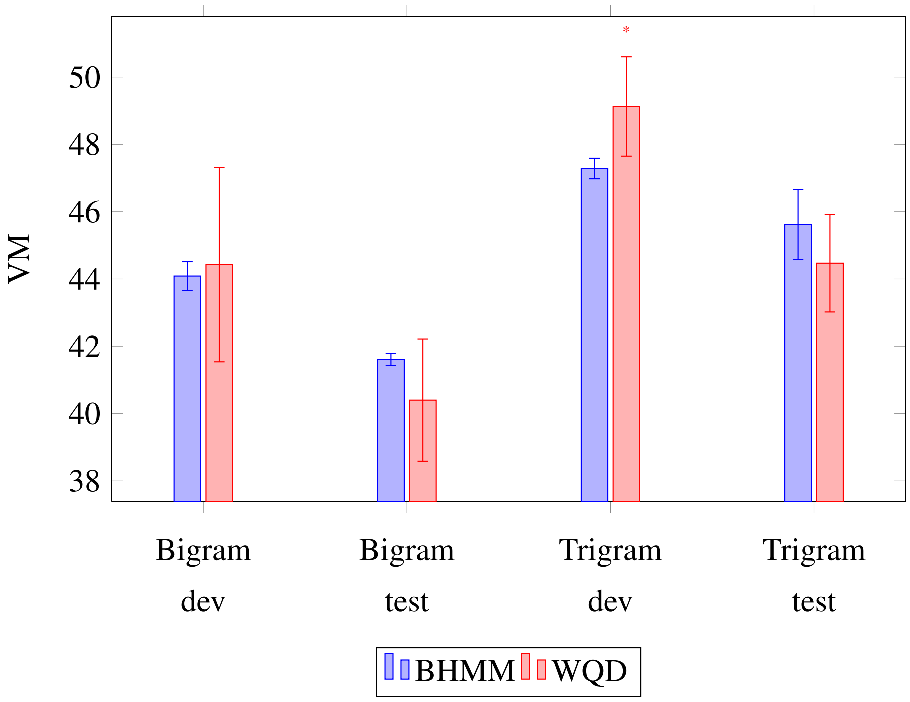

Figure 3.6: BHMM-T performance on the Spanish (Ornat) corpus: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

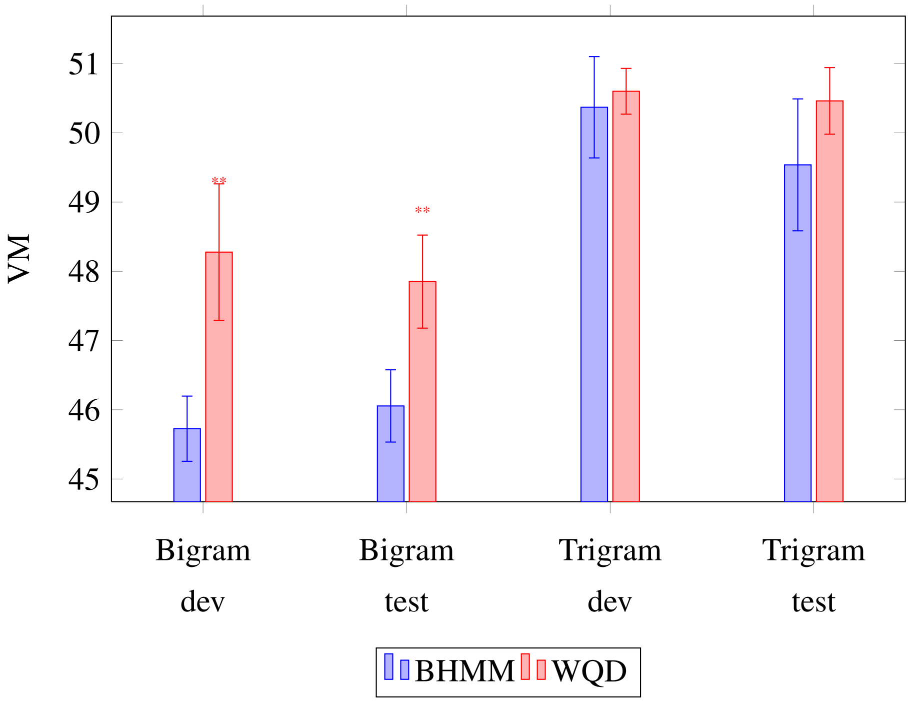

Figure 3.7: BHMM-T performance on the Cantonese (LWL) corpus: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

not show the same improvement (Figure 3.6). The estimated BHMM-T models do not differ significantly from the baseline BHMM; however they have much higher variance. This indicates that the BHMM-T is harder to estimate, presumably because the separate transitions merely introduce more parameters without offering the same benefits as in English. Given that sentence type does not affect word order in the same way in Spanish as in English, this is an unsurprising result.

In Cantonese, we see a significant improvement for the bigram BHMM-T (Figure 3.7). This is despite the fact that Cantonese has relatively little word order marking of questions; the BHMM-T was able to make use of the extra information. The tagging of *wh*-questions improves most in the BHMM-T in bigram Cantonese models, but declaratives and other questions also improve slightly. Trigram BHMM-T models do not outperform the baseline; as in the larger English models, sentence type does not add significant new information to the trigram context history.

As in English, none of the Spanish or Cantonese BHMM-T models perform significantly worse than the BHMM baseline. Even when sentence type is not an entirely reliable signal of word order, the separately estimated transitions still match the performance of shared transitions.

3.6 Distributional Clustering With Sentence Types

The BHMM tagging model used in the previous experiments is a *token-based* tagger, which means that it tags individual occurrences of words. Many of the most successful unsupervised taggers are *type-based* (Christodoulopoulos et al., 2010): they find a single tag for all occurrences of a word. They rely on the fact that few words are evenly syntactically ambiguous, so even for ambiguous tags, correctly finding the most frequent tag is sufficient for good performance. Token-based models such as the BHMM allow words to have multiple tags, but often result in more ambiguity than is actually present in language, leading to noisy clusters.

In this section we add sentence type information to a type-based tagging model, to investigate whether additional context information helps a system that uses the sum of local contexts to tag a word. One of the earliest and simplest type-based models for syntactic category acquisition is presented in Redington et al. (1998) (henceforth RCF). They use a standard agglomerative clustering algorithm to cluster words based on their context vectors. The context vectors consist of counts of word co-occurrence within a small window (two words on either side). The distance between the context vectors, as measured by Spearman's rank correlation, determines their similarity. More similar items are clustered together at lower points in the resulting dendrogram.

In order to increase the robustness (and tractability) of the algorithm, RCF only cluster the 1000 most frequent words. Likewise, only a small number (150) of very frequent words are used as 'context words', i.e. words whose co-occurence with target words are counted. Co-occurrence with all other words is ignored.

A further parameter, 'similarity limit', is needed to transform the dendrogram produced by the agglomerative clustering algorithm into flat clusters. Clusters are formed by cutting the tree at nodes with children with similarity lower than the limit, which RCF set to 0.2. Using similarity to define clusters in this way means that the number of clusters found by the model cannot be predicted. Additionally, RCF ignore all clusters with fewer than 10 words. This pruning step results in large, relatively clean clusters.

A context vector is an amalgamation of all contexts of all occurrences of a word, and thus is arguably more robust to context variations due to sentence type. However, conflating contexts from all sentence types may still have an negative effect on clustering. For example, a word's appearance before the context word *is* would be evidence that the word in question is a noun, if it is appears in a declarative; in a *wh*-question, the word is more likely to be a *wh*-word, tagged either as a pronoun or adverb.

Our approach to adding sentence types to the context vectors is straightforward: we annotate each context word with sentence type, resulting in context words *is*-`W`, *is*-`D`, and so on. This effectively multiplies the length of the context vector by the number of sentence types. Data sparsity thus becomes a problem once again, since with more context word types, there are fewer examples of each.

3.6.1 Experiments

We run the model described above in two settings, one with the longer context vectors with sentence types, and using the original context vectors without sentence types. We test the model using both 50 and 150 context words. In all other respects we use RCF's settings, except that we also test the full set of clusters without removing small clusters.

Evaluation is done over types, not tokens, using the same reduced tagset as before (which is quite similar to the tagset used by RCF). For words with multiple tags (173 of 1000 target words), we use the most frequent tag as the gold standard, following RCF.

For these experiments we evaluate using VM as before, and also include pairwise

Model	*C*	*W*	VM	VH	VC	PF	PP	PR
1000 target words, 150 context words, clusters of size ≥ 10								
Without s-types	18	642	**38.1**	52.3	30.0	32.7	72.3	21.1
With s-types	16	763	25.4	30.5	21.7	**35.9**	48.8	28.4
1000 target words, 150 context words, all clusters								
Without s-types	135	1000	**32.7**	58.9	22.6	15.8	71.9	08.9
With s-types	87	1000	25.0	38.2	18.6	**25.2**	48.8	17.0
1000 target words, 50 context words, clusters of size ≥ 10								
Without s-types	16	755	**33.2**	43.9	26.7	32.4	61.9	21.9
With s-types	11	784	27.1	30.9	24.1	**41.7**	58.0	32.6
1000 target words, 50 context words, all clusters								
Without s-types	85	1000	**29.9**	48.1	21.7	21.4	61.6	12.9
With s-types	67	1000	26.2	37.6	20.2	**30.6**	57.9	20.8

Table 3.8: Performance of hierarchical clustering models with and without word features separated by sentence-type. C is the number of clusters and W is the number of words in the final clustering. The best result for each setting as measured by V-measure (VM, composed of VH and VC) and Pairwise F-score (PF, composed of PP and PR) is in bold.

precision and recall as an evaluation method, since RCF use it. Pairwise precision and recall, due to the pairwise nature, over-emphasises larger clusters; conversely, as pointed out by Vlachos et al. (2009), higher VM scores are more easily achieved by models with a larger number of smaller clusters.

The results, in Table 3.8, demonstrate the difficulty of evaluating and comparing unlabeled clusterings, particularly clusterings with different numbers of clusters. Each setting creates a different dendrogram, resulting in different numbers of clusters and even different numbers of clustered words, when words in clusters of size < 10 are removed.

Pairwise F-score is consistently higher for clusterings using sentence type information, but it is unclear how much this is due simply to the fact that these clusters are larger. VM scores are higher for the models without sentence type, but again, these models cluster fewer words into a larger number of clusters, resulting in small clusters.

It is perhaps counterintuitive that extending the context vectors, by adding sentence type to each context word, would result in fewer clusters at the same similarity level. In effect, the clustered words have higher similarity when clustered using the sparser vectors than the denser vectors without sentence types. This indicates that distinguishing between context words in different sentence types is capturing a meaningful distinction, and not merely adding noise (which would result in lower similarity, given the larger context vector).

Unsuprisingly, the larger clusters are less homogenous, simply by virtue of including more words: both VH and pairwise precision decrease in all settings. Pairwise recall is higher for models with sentence types, but the analogous V-completeness is not, leading to the suspicion that this increase is caused to a large extent by the larger clusters. These measures are also low, due to the mismatch between the number of found clusters and the much smaller number number of gold categories.

Including the smallest clusters decreases V-completeness much less than pairwise recall, demonstrating VM's greater robustness to cluster size and number. Neither precision measure increases, indicating that the smaller clusters remain noisy. Interestingly, performance on the models with sentence type declines less drastically when all clusters are evaluated, again indicating that sentence type is adding a degree of robustness.

On a methodological level, this type-based model demonstrates the importance of developing models that are not dependent on somewhat arbitrary parameters (here: similarity cutoff and cluster size cutoff). This model does avoid a parameter setting the number of clusters, necessary in the BHMM, but this results in clusterings that are difficult to compare over a variety of settings, making it hard to draw conclusions from experiments involving model structure.

3.7 Discussion

The BHMM-T models with sentence-type-specific transitions introduced in this chapter demonstrate that sentence type can be an informative cue for word and tag order, especially in models with limited amounts of local context. The amount of input data available to the model and the number of model parameters affected which set of sentence types performed best; further research is necessary to characterise these interactions more precisely. However, we showed that at the very least sentence-type-specific distributions never found worse syntactic clusters than shared distributions, and in many cases found better clusters. Arbitrarily adding parameters — and thereby splitting distributions — in unsupervised models is unlikely to improve performance, due to ensuing data sparsity, unless the parameters are genuinely useful (unlike supervised models, in which unhelpful parameters will be ignored). This problem arose when the sentence type parameter was added to the emission distribution. On the other hand, adding this parameter to the transition distribution resulted in an larger number of distributions estimated on smaller amounts of data, but these models were able to recover or improve on the original performance. This demonstrates that the sentence type parameter added useful additional information to the model in this setting.

These modelling results have shown that computationally there may be an advantage to a representation of word/syntactic category order that includes sentence type information. Experimental work has demonstrated infants' awareness of adjacent contexts as cues for categorisation at 12 months (Gómez and Lakusta, 2004). At this stage, they are also clearly aware of prosody, both in its function to signal dialogue pragmatics as well as using it as a feature for storage of heard phrases (Mandel et al., 1996). However, it has yet to be shown that infants' representation of early syntax (while learning syntactic categories) allows for a link between prosody and word order, i.e., that infants can make use of the fact that different prosodies may signal different word orders. An artificial language learning experiment for learning categories, similar to Gómez and Lakusta's (2004) study, but in which transitions probabilities are manipulated along with prosody cues, could provide evidence for infants' ability (or otherwise) to link word order and prosody. Crucially, infants would also have to be able to ignore prosody when it is not informative, given that the link between word order and prosody is language dependent and only informative in certain contexts.

This link between prosody and sentence structure is also relevant for sentence processing in adults as well as children. The effect of prosody on sentence processing has primarily been investigated in terms of how prosodic boundaries affect structural disambiguation, and has found that children acquire this ability quite late. However, sentential prosody may also signal an expectation for a certain word order or grammatical structure, which could reduce processing effort by boosting the probability of sentences structures that comply with heard prosody. In a recent eye-tracking study, Zhou et al. (2012) investigated the ability of adults and four-year-old children to use prosody to distinguish between otherwise surface-identical Mandarin questions and declarative sentences in an on-line fashion. They found that children performed at an adult level, which is not the case for other types of structural disambiguation, such as attachment, at this age. In other languages prosody is often redundant; sentence structure can be inferred from words alone. However, this redundancy may facilitate online processing and disambiguation. Demonstrating this experimentally would strengthen the case for sentence type being a salient cue during prosody, as it is for memory (Mandel et al., 1994).

This chapter has investigated the potential benefits of linking aspects of language that are usually treated separately, such as prosody and syntax. In other work, models of word learning have shown the importance of incorporating higher-level features such as discourse structure (Frank et al., 2009), continuity of discourse, and non-linguistic features such as joint attention (Tomasello and Farrar, 1986). The BHMM-T presents initial evidence that similar higher-level, non-local features may be useful in syntax acquisition as well.

3.8 Conclusion

We have investigated whether sentence type can be a useful cue for models of syntactic category acquisition. The structure of the BHMM made it possible to distinguish between adding sentence types to either the parameters governing the order of syntactic categories in an utterance (transition distributions), or to the parameters describing the words making up the syntactic categories (emission distributions). We found that, as expected, adding sentence type to emission distributions resulted in degraded performance due to the decreased amount of data available to estimate each separate sentence type emission distribution; additionally these models are awkward since they create separate sets of clusters for each sentence type. This is contrary to syntactic categories as they are generally understood and leads to a representation with high levels of redundancy. Hence we dismiss this model structure for syntactic category acquisition.

However, the model that included sentence type in the transition distribution (BHMM-T) performed better, by capturing word order difference across sentence types. The cross-linguistic results showed that even in languages in which word order variation between sentence types is not as strong as in English were occasionally able to benefit from the additional information. Where sentence type did not add additional information, as in Spanish, the BHMM-T models again matched baseline performance.

We also investigated adding sentence type to a type-based model using agglomerative clustering of context vectors. Results were inconclusive, with the variable number of clusters found in the different settings having a greater effect than adding sentence types. However, adding sentence types allowed the model to find larger clusters of roughly comparable quality, indicating that including sentence type in contexts did not result in data sparsity issues, and may allow for more general, robust clusterings.

The BHMM-T is fairly brittle, without any sharing of knowledge of transitions between sentence types. A more realistic model would interpolate between sentence-type-specific transition distributions, where these are more informative, and general distributions, where sentence type information is lacking, in order to be more robust to issues of data sparsity. It would also make use of the fact that some transitions are often shared between sentence types (i.e., nouns are often preceded by determiners in both questions and declaratives). In this chapter, we have demonstrated that adding a link between sentence type prosody and word order at minimum does not add noise, and in many cases is a source of additional information. In the following chapter we will investigate adding flexibility to the inclusion of sentence type in the model.

Chapter 4

BHMM with Transition Groups

In the previous chapter, we introduced sentence type as an observed variable within the BHMM. We found that when transitions were conditioned on sentence type, the models were able to reflect the fact that different sentence types are characterised by differences in word order, and this led to improved part of speech tagging performance.

However, not all aspects of word order are affected by sentence type. Noun phrases, for instance, have the same structure regardless of sentence type. In terms of transitions, this structural consistency should be evident in similarly high probability of `DET NOUN` trasitions, and low probability of `DET VERB`, for all sentence types. In the BHMM-T, when all transitions are separated by sentence type, the probabilities for these transitions must be estimated separately for each sentence type, leading to redundancy, and potentially raising issues of data sparsity due to insufficient counts. The results of the BHMM-T reflect this difficulty: in many cases increasing the number of sentence types decreased performance despite the greater refinement of the categories.

A second problem with the BHMM-T is that it requires us to define the number and kind of sentence types. The five sentence types we identifed were linguistically motivated and, we argued, distinguishable by language learners. However, we found differences in performance between different *partitions* of the sentence types. For instance, what we called the WQD sentence types, in which *wh*-questions and other questions are distinguished, but short and imperative utterances are grouped together with declaratives, can also be thought of as a partitioning of five sentence types into three groups: W, Q, and ISD. This partition resulted in better performance for bigram models than the full five-way partition W,Q,I,S,D. However, perhaps a QW,DS,I, or W,QD,IS partition would have performed even better: we did not investigate all possiblities. The number of possible partitions is given by the Bell number and grows large quickly. Testing the full set of possible partitions in an exhaustive search is prohibitive.

Moreover, not all tag transitions may benefit from the same sentence type distinctions. For example, verbs may be sensitive to imperative mood and have a different transition distribution in imperatives, but similar transitions in declaratives and questions; whereas the transition distribution governing *wh*-word adverbs (*how,why*) may be more sensitive to the *wh*-question distinctions. In this case, verbs may benefit from a sentence type partition of I,DQWS while adverbs would benefit from transitions partitioned as W,QDSI. These examples are hypothetical: the key points are that firstly, it is difficult to intuit *a priori* which partitions are optimal, and secondly that these partitions may not be the same for all transition distributions.

In this chapter, we add an additional set of latent variables to the BHMM-T, namely *transition groups*, that add additional flexibility to the transition distributions. Broadly speaking, where in the BHMM-T each transition distribution was split according to sentence type, in the BHMM-TG — the BHMM with transition groups — transitions can be shared or split by sentence type. Transitions between states are now conditioned on the transition groups of the previous tags, not on the previous tags themselves; in this way, tags that share transition groups will share transition distributions. For example, imperatives and questions might have separate sentence-type-specific transitions from a `VERB` state, since these transition distributions are likely to be very different in these two sentence types, but could share transitions from a `DET` state, by being in the same transition group, which would represent the fact that noun phrases are stable across sentence types.

The structure of this chapter is as follows: in the following section, Section 4.1, we formally describe the BHMM-TG. In Section 4.2 we present two samplers for performing inference over the BHMM-TG, one which changes the transition groups a single sentence type at a time, and one which resamples the full partition of all sentence types. Experimental results follow in Section 4.3. We first confirm our hypothesis about the utility of transition groups by fixing the tags to gold values, estimating only the transition groups, and show that the inferred transition groups roughly match linguistic intuition. We then infer both tags and transition groups, in a full model, and compare the results to the models from the previous chapter. We find that the BHMM-TG does not consistently outperform the BHMM-T; in the concluding discussion of this chapter we discuss why, as well as present ideas for future work in this line.

4.1 Model Definition

In the BHMM-T, sentence type is used as a conditioning term in the transition distribution:

$$p(t_i|t_{i-1}=t',s_{i-1}=s)=\tau_{(t',s)}. \tag{4.1}$$

This improves performance by modelling the sentence type variation in word order, but at the cost of fractioning the counts for transition distributions that do not vary between sentence types.

In the BHMM-TG — the BHMM with transition groups — sentence type remains an observed variable, as in the BHMM-T. We add inferred transition groups $\boldsymbol{g}$, so that sentence types can share transitions. Each tag (type) t has a set of transition groups $\boldsymbol{g}_t$, which represent a partitioning of the sentence types. For example, `Tag9` might have three (active) transition groups: one group with both question types (Q and W), one with declarative and short utterances (D and S), and one with imperatives (i.e., a QW,DS,I partitioning), while `Tag2` might partition all sentence types into a single transition group (i.e., ISWQD). A sentence type's group for a particular tag is denoted $g_t(s)$.

When calculating transition probabilities, we condition on the transition group of the previous tag and sentence type, rather than using the previous tag directly:

$$p(t_i|t_{i-1}=t',s_{i-1}=s,\boldsymbol{g},\tau)=\tau_{(g_{t'}(s))} \tag{4.2}$$

In trigram models, we use the transition groups of the previous two tags:

$$p(t_i|t_{i-1}=t',t_{i-2}=t'',s_{i-1}=s,s_{i-2}=s',\boldsymbol{g},\tau)=\tau_{(g_{t'}(s),g_{t''}(s'))} \tag{4.3}$$

The emission distributions remain the same as in the BHMM, conditioned only on the current tag (not on the transition group).

$$p(w_i|t_i=t,\omega)=\omega_{(t)} \tag{4.4}$$

The assignment of each sentence type to a transition group $g_t(s)=g$ is drawn from a multinomial ϕ with a Dirichlet prior γ. The multinomial distribution ranges over G, the number of transition groups available; this is usually set to be equal to S, the number of sentence types. If two sentence types draw the same transition group, they are in the same partition. The prior γ is set to encourage mildly sparseness ($\gamma=0.1$), which in this case results in multiple sentence types drawing the same transition group, and other transition groups remaining empty.

The full model description for a bigram BHMM-TG, depicted in Figure 4.1, is:

$$\begin{aligned}
\phi_{(t)}|\gamma &\sim \text{Dirichlet}(\gamma)\\
\tau_{(g)}|\alpha &\sim \text{Dirichlet}(\alpha)\\
\omega_{(t)}|\beta &\sim \text{Dirichlet}(\beta)\\
g_t(s)|t &\sim \text{Mult}(\phi_{(t)})\\
t_i|g_{t_{i-1}}(s_{i-1}) = g,\tau &\sim \text{Mult}(\tau_{(g)})\\
w_i|t_i = t,\omega &\sim \text{Mult}(\omega_{(t)})
\end{aligned}$$

The full joint posterior of a sequence of words $\boldsymbol{w}$, tags $\boldsymbol{t}$, groups $\boldsymbol{g}$, integrating over parameters τ, ω, and ϕ, for a bigram BHMM-TG model is:

$$\begin{aligned}
P(\boldsymbol{w},\boldsymbol{t},\boldsymbol{g}|\boldsymbol{s},\alpha,\beta,\gamma) &= P(\boldsymbol{g}|\gamma)P(\boldsymbol{t}|\boldsymbol{g},\boldsymbol{s},\alpha)P(\boldsymbol{w}|\boldsymbol{t},\beta) && (4.5)\\
&= \prod_{t\in T}\prod_{g\in G_t}\frac{\Gamma(S+\gamma)}{\Gamma(1+\gamma)}\frac{\Gamma(1+G\gamma)}{\Gamma(m_g+\gamma)} && (4.6)\\
&\times\prod_{t\in T}\prod_{g\in G_t}\prod_{t'\in T}\frac{\Gamma(n_{g_t}+\alpha)}{\Gamma(1+\alpha)}\frac{\Gamma(1+T\alpha)}{\Gamma(n_{g_tt'}+\alpha)} && (4.7)\\
&\times\prod_{t\in T}\prod_{w\in V}\frac{\Gamma(n_t+\beta)}{\Gamma(1+\beta)}\frac{\Gamma(1+V\beta)}{\Gamma(n_{tw}+\beta)} && (4.8)
\end{aligned}$$

The first factor (line 4.6) are the transition groups, whose sufficient statistics m_g are the number of sentence types drawn from transition group g. In line 4.7 we have the transitions from groups to tags, and line 4.8 is the posterior for the emissions from the tags. Note that the sentence type observations almost disappear: once they are associated with a transition group at the top level, the transition group now contains all the necessary information.

The BHMM-TG is a generalisation of the BHMM and the BHMM-T: when all sentence types share the same transition group for all tags, we recover the BHMM. Conversely, if all sentence types are in separate transition groups, we recover the BHMM-T[1].

---

[1] There is a small technical difference in the trigram model because the BHMM-TG takes s_{i-2} into account whereas BHMM-T does not; however, in practice, $s_{i-2} = s_{i-1}$, when both are in the sentence, or it is the boundary sentence type, which has its own special tag. In either case the BHMM-T and BHMM-TG result in equivalent conditioning distributions.

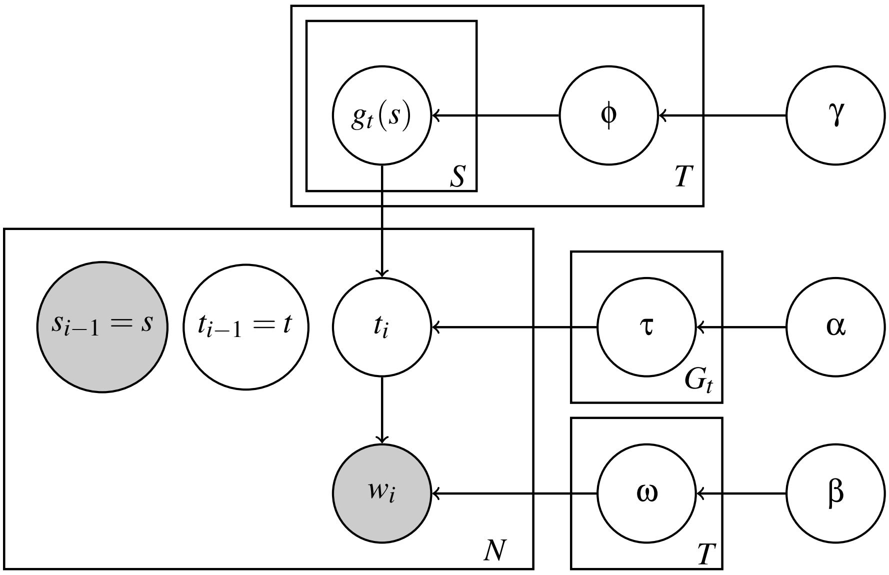

Figure 4.1: Graphical model representation of the bigram BHMM-TG. The values of the previous tag t_{i-1} and sentence type s_{i-1} (i.e., t and s, respectively) determine the transition group $g_t(s)$, which in turn determines the transition distribution $\tau_{(g)}$ from which t_i is drawn.

4.2 Inference

The BHMM-TG has two sets of latent variables that must be inferred: the part of speech tags and the transition groups. The tags are inferred using the same Gibbs sampler as in the BHMM (see Section 2.3.5), with adjustments for the transition groups. The predictive probabilities for tags now involve transition probabilities that are conditioned on the transition groups, as above in Eqs. 4.2 and 4.3. The transition group identities (and thus the transition distributions) are stable during the tag sampling stage. The transition groups are initialised at random, and are resampled after every ten iterations of tag sampling.

We implemented two different samplers for transition group sampling. The first is a Metropolis-Hastings sampler and changes the transition group of a single sentence type at a time. The second is a Gibbs sampler that samples from the space of all possible partitions of the sentence types.

4.2.1 Local Metropolis-Hastings Sampler

This sampler updates the transition group for a single sentence type at a time. In each sampling iteration, the sampler iterates through the set of tags T. At each tag (type), a sentence type s and a proposed new group g'_t is chosen uniformly at random. (Because the proposal distribution is uniform, this is technically an independence chain Metropolis-Hastings sampler.)

In order to decide whether to move s from its current group $g_t(s) = g$ to $g_t(s) = g'$, we draw from the acceptance probability distribution:

$$p(\text{accept } g'_t(s)) = min\{1, \frac{\pi(g'_t(s))}{\pi(g_t(s))}\} \tag{4.9}$$

We use Equation 4.5 to calculate $\pi(g'_t(s))$, the posterior probability of the current model with the addition of the new group assignment. $\pi(g_t(s))$ is simply the current posterior probability of the model without the change of groups.

4.2.2 Global Gibbs Partition Sampler

The local sampler makes small moves, changing only a single group at a time, and hence might suffer from mixing problems. For example, if reaching the optimal partition were to require two moves that both had low individual probabilities, the local sampler would be unlikely to make both moves in succession.

For this reason, we implemented a second sampler that samples the full group partition directly, using a Gibbs sampler over the set of possible partitions. However, this sampler is only tractable for small numbers of transition groups, as the number of possible partitions rapidly becomes very large. (The number of possible partitions of n items is given by its Bell number: $B_2 = 2, B_3 = 5, B_4 = 15, B_5 = 52$.)

The global Gibbs partition sampler iterates over each tag in T. For each tag, we generate all possible partitions $\boldsymbol{g}_t$ of the sentence types (e.g., WQD; W,DQ; WD,Q; WQ,D; W,Q,D) into transition groups and calculate their probability given the current tags:

$$P(\boldsymbol{g}_t|\boldsymbol{t}, S, \gamma, \alpha) = \prod_{g \in G} \frac{\gamma^{(m_g)}}{G\gamma^{(S)}} \prod_{t \in T} \prod_{t \in T} \frac{\alpha^{(n_{gt})}}{T\alpha^{(n_g)}} \tag{4.10}$$

where m_g designates the number of draws from transition group g (i.e., the number of sentence types assigned to that transition group) and n_g designates the number of token transitions from g. G is the number of transition groups per tag, S is the number of sentence types, and T is the number of tag types. The notation

$x^{(n)} = (x)(x+1)...(x+n-1)$ denotes the ascending factorial, used to calculate the summation of the Dirichlet-multinomial predictive posterior probabilities. The first factor calculates the posterior probabilities of the transition group multinomials, while the second factor calculates the posterior probabilities of the current tag transitions given the partition $\boldsymbol{g}_t$.

4.3 BHMM-TG Experiments

4.3.1 Data

We use the same datasets as in Chapter 3, described in Section 3.3.1: the smaller Eve dataset and larger Manchester dataset for English, and the Spanish Ornat corpus and Cantonese LWL corpus to test on other languages. We follow the same train/dev/test regime as described in Section 3.3.1 and report results on the test+training sections.

These datasets are annotated with five sentence types (see Section 3.1): *Wh*-questions (W), Questions (Q), Declaratives (D), Imperatives (I), and Short utterances (S). We test the same subsets of these sentence types as well:

QD All questions separatated from all other sentence types (catch-all ‘declaratives’).

WQD Distinct *wh*-questions, other questions, and ‘declaratives’

SWQD Short (2 words or less) utterances are separated from the other declaratives, questions remain separate.

ISWQD Imperatives are also distinguished from other declaratives.

4.3.2 Baselines and Procedure

We want to evaluate the BHMM-TG against the BHMM and the BHMM-T model from Chapter 3. In order to compare these models directly, we use the variations of the BHMM-TG that are equivalent to the BHMM and the BHMM-T. For the BHMM, this is the BHMM-TG model with only a single transition group per tag, so that all sentence types share the same transition group. The BHMM-T is a special case of the BHMM-TG in which each sentence type is assigned to a separate transition group and sharing is disallowed. In both cases transition groups are set at the beginning of sampling and thereafter not resampled.

We run inference on models trained on the Eve, Ornat and LWL datasets for 10000 iterations, as before. The larger Manchester models are run for 8000 iterations. The α and β hyperparameters are estimated using a Metropolis-Hastings sampler, while γ is set to 0.1, as initial experiments showed that changing the value of this hyperparameter has little effect. Each experimental condition is run ten times. We report average values for all evaluation measures and confidence intervals at the 95% level.

4.3.3 True Tags

In this set of experiments, we fix the tags to the gold standard tags and perform inference only over the transition groups, to see what kind of patterns the transition groups find. We cannot evaluate the transition groups quantitatively (since there is no gold standard to evaluate against), but we can inspect them to see if they match our intuitions.

As an initial experiment, we tested transition groups with randomly assigned sentence labels and gold tags. In this case, the trained models invariably shared all transition groups between all sentence types. This is to be expected, since there would be no significant differences in transitions between meaningless sentence types. This is a crucial advantage of the BHMM-TG: it has the flexibility to ignore uninformative features, unlike BHMM-T, which would be faced with a fragmented dataset.

4.3.3.1 Small dataset experiments: Eve results

We noted in the BHMM-T results in Chapter 3 that performance on the relatively small Eve dataset decreased with more sentence types. We surmised that this decrease in performance was due to data sparsity when estimating sentence-type-specific transitions. The BHMM-TG model can ameliorate the data sparsity problem by allowing the model to group transitions from sentence types without enough support.

The grouping occurs on a tag-by-tag basis, depending on the transition behaviour of each tag. We can hypothesise likely groupings based on the word order characteristics of the tags:

1. Tags such infinitival-*to* (`INF`), determiners (`DET`), prepositions (`PREP`), and adjectives (`ADJ`) should have their transition distributions shared among all sentence types, since their distributional usage constraints are determined by local context, not global context features like sentence type.

2. Tags that change position in the course of word order shifts between declaratives and questions — pronouns (`PRO`), verbs (`V`), auxiliaries (`AUX`) — should have sentence-type-specific transition distributions.

We run a bigram model ten times, using the Eve development set annotated with all five sentence types. The following table shows the transition group partition of sentence types for each tag in each of the ten runs. When all ten runs have the same partition, we omit the count, but if all ten runs did not converge on the same partition, the number of runs resulting in that partition is noted in the subscript (e.g., `AUX`$_7$ in partition D,I,Q,S,W indicates that in seven of ten runs, the transition groups found for the `AUX` tags were completely split between all sentence types).

1	D,I,Q,S,W	`ADV`, `OTH`, `PRO`, `V`, `AUX`$_7$,`N`$_2$
2	DI,Q,S,W	`AUX`$_3$
3	DQ,I,S,W	`N`$_8$
4	D,IQ,S,W	`NEG`$_5$
5	D,I,QS,W	`CONJ`$_3$
6	D,IQW,S	`NEG`$_5$, `DET`
7	D,IQS,W	`CONJ`$_7$
8	DQ,IW,S	`PART`
9	DIQ,S,W	`ADJ`$_8$
10	DQW,I,S	`PREP`$_1$
11	DIQW,S	`ADJ`$_2$, `PREP`$_9$
12	DIQW	`INF`

A large variety of partitions are found: twelve different partitions are used over all runs for thirteen tags (nine partitions if only the most frequent partition for each tag is counted). This indicates that, given stable, high-quality tags, the transition group model is able to use the extra flexibility in representation to capture meaningful patterns in the data.

Seven of thirteen tags converge to the same partition on all runs: `ADV, OTH, INF, PRO, DET, PART`, and `V`. The others alternate between two partitions that are slight variations: `NEG` is undecided whether to cluster *wh*-questions with other questions and imperatives (partitions 4 and 6), nouns (`N`) usually cluster declaratives and other questions together (in partition 3) but occasionally separate them (partition 1), and so on. Three of the partition alternations (`AUX`, `CONJ`, `PREP`) involve imperatives, the least frequent sentence type.

Note also that our posited collapsing of the five sentence types into three (DIS,W,Q) does not appear at all. This supports the desire for flexibility in sentence type separations, since evidently our intuitions about likely sentence type groupings do not correspond to the optimal partitions for the model.

Returning to our first hypothesis: tags such as adjectives (partitions 9 and 11), prepositions (partitions 10 and 11), and infinitival *to* (partition 12) are indeed shared between most sentence types. (`INF` does not appear in any short utterances in the dataset.) Short utterances, unsurprisingly, are different from the longer sentence types and are often separated out.

Likewise as hypothesised, these models infer completely split transition group partitions for pronouns, verbs, auxiliaries — the tags that are affected by word order variation between sentence types. Other tags that are split completely are adverbs and the ‘other’ category, both of which contain words that are less restricted in terms of word order (e.g., *right, just, here, yes, darling*). They are also both diverse ‘catch-all’ categories, to some extent, with the words in the categories likely to differ between sentence type, which may also influence the contexts in which they appear. For example, adverbs in *wh*-questions are likely to be *wh*-words at the beginning of the utterance, while adverbs in declaratives are more likely to be near the verb.

4.3.3.2 Larger dataset experiments: Manchester results

In the Manchester experiments with the bigram BHMM-T (Section 3.4.3) we found that the most fine-grained sentence types (ISWQD) did not perform better than the coarser three sentence type split (WQD), despite the larger amount of data to infer sentence-type-specific transitions from,

We now explore whether this indicates that distinguishing the imperatives and short utterance types is harmful in larger datasets. When given gold tags, does the bigram BHMM-TG with five sentence types recreate the W,Q,D partition? And is W,Q,D optimal — or does the three sentence type BHMM-TG (i.e., a model trained on WQD data) find other partitions?

We train both bi- and trigram BHMM-TG models with gold tags and two different sentence type annotations: one with only three sentence types, WQD, and one with the full set of five sentence types, ISWQD. In the WQD condition, where three different sentence types are observed, and which corresponds to a W,Q,DIS partition of the full five sentence types, we indeed find that the bigram models (both samplers) consistently split all sentence types fully in all tags, recovering the BHMM-T. In the ISWQD condition, with five possible transition groups, we also find that the bigram models prefer to split most tags completely. Only the `INF` tag transition (corresponding to infinitival *to*) is shared between the imperatives and either the declaratives or non-*wh* questions (i.e., either DI,S,W,Q or D,S,W,QI partitions).

The posterior probability of the nearly fully-split models found in the five sentence type condition is higher than in the three sentence type condition in which some of the sentence types are implicitly shared (since the D sentence type contains I and S as well). The transition group component of the posterior is lower, due to the larger number of clusters, but this is cancelled out by higher probability transitions. (Emission probabilities are the same, since the tags are fixed to their true values.) In contrast, the five sentence type Eve BHMM-TG models from the previous section, which share some transition groups between sentence types, have higher posteriors than fully split model run on gold tags.

Overall, the Manchester results demonstrate that, given sufficient data and gold tags, there is little pressure to group transitions together: both samplers find fully split solutions. When performing inference over the smaller Eve data set, sharing transitions can be a better strategy.

These results have been achieved with gold tags. This significantly simplifies the problem, both because the tags are of high quality, so transitions are not noisy, and also because they are stable and do not change during inference. We now turn to experiments involving inference over both tags and transition groups, to see if the same patterns hold with inferred tags as well as gold tags.

4.3.4 Inferred Tags and Transition Groups

We now test the ability of the BHMM-TG to infer tags as well as sentence type partitions. We evaluate the inferred tags using VM (see Section 2.4). If the transition groups allow for a better, more flexible integration of sentence types into the BHMM structure than this should be reflected in improved performance with regard to the BHMM and the BHMM-T.

Furthermore, the flexibility of transition groups should make the model more robust to sentence type distinctions, since transition groups allow the model to recover coarser distinctions. We thus expect to see either increasing or stable performance with increasingly fine sets of sentence types.

4.3.4.1 Eve results

Results for the Eve corpus are shown in Figure 4.2, comparing both samplers for the BHMM-TG against the BHMM and the BHMM-T.

For the most part, the global partition sampler (BHMM-TG-P) finds better solutions than the local sampler (BHMM-TG-L), although the results are comparable, despite the samplers' very different sampling strategies. This indicates that neither sampler suffers from a severe mixing or convergence issue. In general, we find that in both samplers the transition groups converge within the first few hundred iterations and thereafter do not change significantly.

The BHMM-TG models are making use of transition groups: transitions are grouped into a variety of different partitions. Due to the lack of cluster labels, it is difficult to see what these partitions are capturing. The bigram models find the fully split partition (maximum number of transition groups) more often than trigram models. The fully split partition also occurs more often in models with fewer sentence types. This indicates that data scarcity is key to driving sharing of transition groups: both trigram models and increasing numbers of sentence types result in more distributions to estimate. This can be ameliorated if tags share transition groups, since they will then have to estimate fewer transition distributions.

Trigram performance increases with the number of sentence types, whereas the bigram models find the best performance with three sentence types, as with the BHMM-T models. However, the bigram models all perform within statistical significance of one another (and the BHMM-T baseline). They also all significantly outperform the BHMM baseline.

The only models that clearly underperform the BHMM-T models are the trigram models trained on data with only two sentence types (QD). This is due to the transition groups opting to share transitions very often, in 87 of 130 cases (13 tags times 10 runs). The BHMM-TG and BHMM-T models converge to about the same posterior log probability, with the BHMM-TG finding a slightly higher probability solution on average. However, the transition component in BHMM-TG models is higher than in the BHMM-T models, while the reverse is true for the emission component. Evidently the emission component corresponds more closely to VM performance.

In summary, we do not find a consistent increase in tagging performance in the BHMM-TG models over the BHMM-T models on the Eve datasets. There is also not a consistent difference in performance between sentence type settings; in the trigram

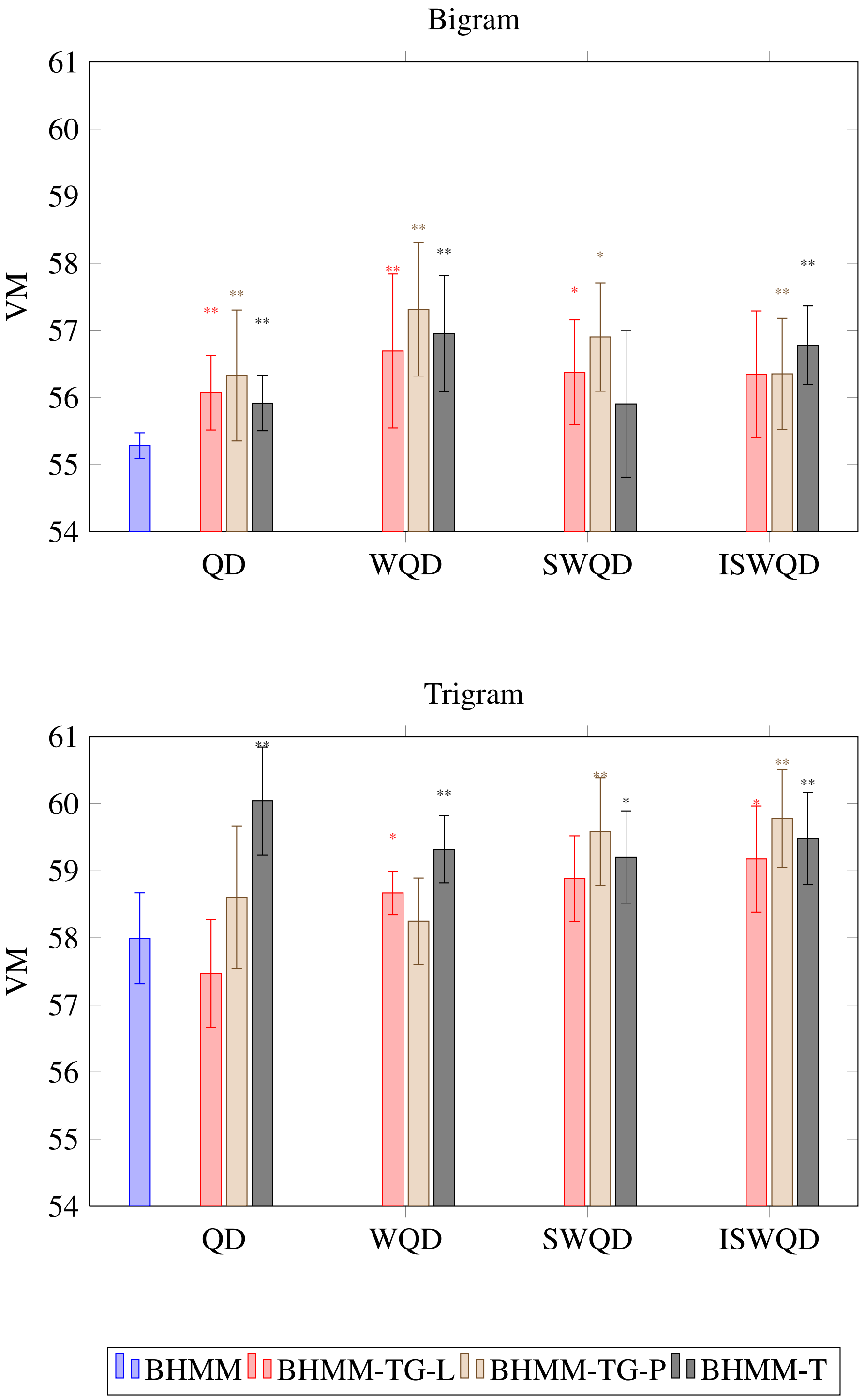

Figure 4.2: BHMM-TG performance on the Eve corpus: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

models, there is a clearer trend of increasing sentence types leading to better performance overall, but this does not hold for bigram models.

4.3.4.2 Manchester results

In the BHMM-T results on the Manchester corpus (Section 3.4.3), we found a clear example of how a larger number of sentence types could cause a decline in tagging performance. As we described in Section 4.3.3.2, the bigram BHMM-T with three sentence types (WQD) clearly outperformed the baseline BHMM, whereas the BHMM-T with five sentence types only improved slightly over the baseline. The BHMM-TG should be able to recover the better WQD distinction even when being given the full set of sentence types, stemming the decline in performance.

However, examining the bigram results in Figure 4.3, we see that this does not occur: the BHMM-TG over five sentence types matches the (lower) performance of the ISWQD BHMM-T almost exactly, rather than being able to match the WQD BHMM-T's performance. Indeed, the inferred transition groups are nearly all fully split, recovering the BHMM-T, in both the WQD and ISWQD models. In the case of the WQD sentence types, this leads to high VM performance, but not with further sentence type splits.

The posterior probabilities of the ISWQD bigram BHMM-TG models are significantly higher than those of the WQD models, a fact not reflected in VM performance. As before, this is due to the ISWQD model having higher probability transitions, while the emissions and transition group components of the posterior are of higher probability in the WQD model.

The trigram models split less completely than the bigram models. In some cases, as for the BHMM-TG local sampler with three or four sentence types (WQD and SWQD), this improves performance. However, this is not consistent, and the partition sampler does not find the same solution. In the case of the ISWQD results on BHMM-TG-L, the sharing of sentence types (only half the possible partitions were fully split) results in lower performance than the fully split BHMM-T model, but not significantly. We were unable to run the partition sampler on the five-sentence type dataset due to its prohibitive runtime. Both the BHMM-TG and BHMM-T trigram models do not show much improvement over the BHMM: as in Chapter 3, this suggests that sentence type is mainly useful in models with impoverished context or smaller amounts of data to learn from.

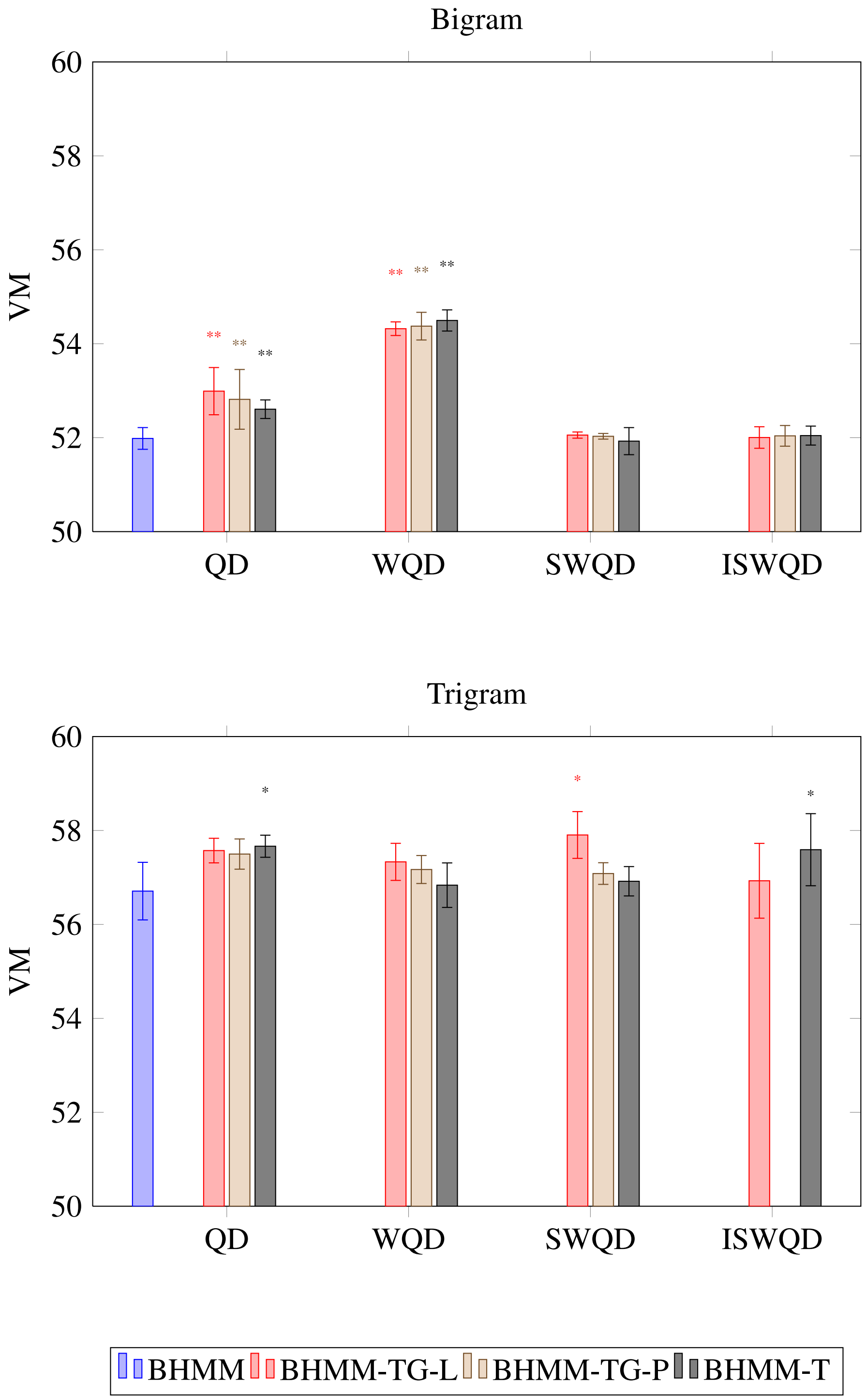

Figure 4.3: BHMM-TG performance on the Manchester corpus: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

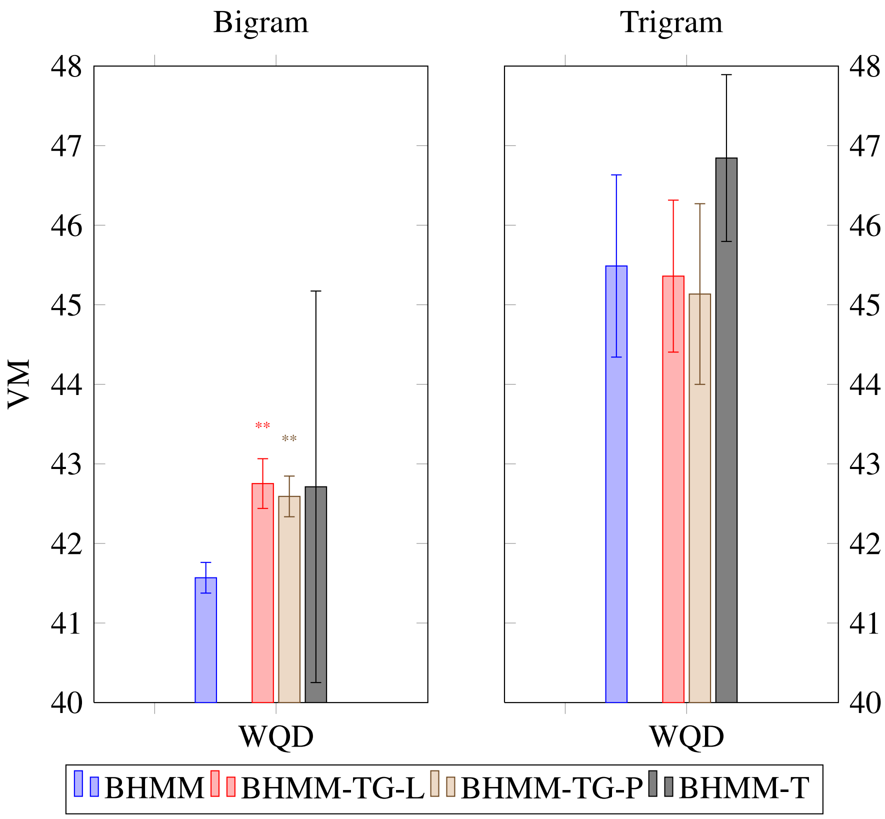

Figure 4.4: BHMM-TG performance on the Spanish (Ornat) corpus: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

4.3.4.3 Other Languages

As before, we also test the BHMM-TG on non-English corpora. For these datasets we only identify three sentence types, WQD, so we are principally interested in the performance of the BHMM-TG with regards to the BHMM-T and BHMM baselines.

The results are shown in Figure 4.4 for Spanish and Figure 4.5 for Cantonese. These figures show similar patterns: the bigram models perform similarly to the BHMM-T, but do not significantly outperform this baseline, whereas the trigram models more closely match the BHMM, resulting in underperformance with regards to the BHMM-T models.

The high variation in the bigram Ornat BHMM-T models is due to a local optimum found by a few models that entails moving some high-frequency verbs (*es*, *está*) into a cluster that primarily contains prepositions. The BHMM-TG models avoid this local

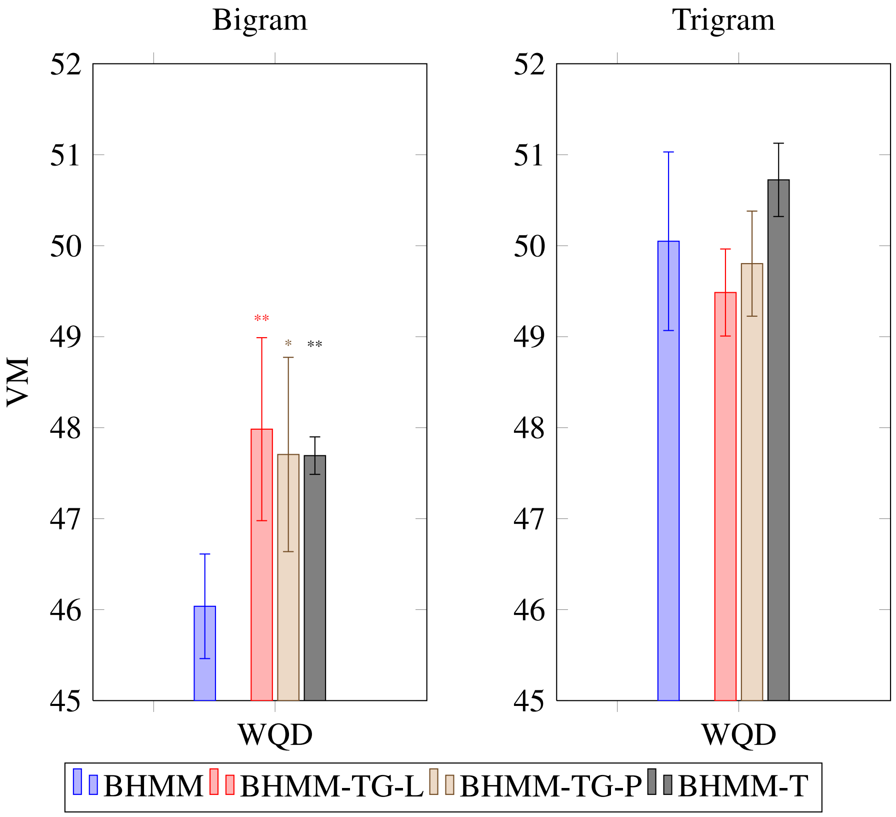

Figure 4.5: BHMM-TG performance on the Cantonese (LWL) corpus: Mean VM and 95% Confidence Intervals ($N = 10$). Values that differ significantly from baseline (BHMM) are marked with $*$ ($p < 0.05$, two-tailed) or $**$ ($p < 0.01$, two-tailed).

optimum and converge to a more consistent solution.

These performance differences between bigram and trigram models are reflected in the levels of sharing in the transition group partitions found: nearly half of the transition group partitions found by the bigram Ornat models are fully split, whereas half of the transition group partitions found in the trigram models have only a single group, with all three sentence types sharing the same transition group. The same pattern is evident in the LWL BHMM-TG transition group partitions.

4.4 Discussion

The BHMM-TG was formulated to add flexibility to the sentence type representation in the BHMM-T. The model had two aims: firstly, to be more robust against the choice of sentence types used in the model, and secondly, to allow different tags to flexibly chose the optimal sentence type partitioning. We hoped these abilities would allow it to find better tags than the BHMM-T and the BHMM-TG. Unfortunately, the BHMM-TG did not show consistent improvement over the BHMM-T. In this section we consider the possible reasons for this result and discuss potential solutions and future steps.

Are transition groups a bad idea? One possible explanation for the results we found is that the transition group structure — the enabling of transition sharing — does not reflect linguistic reality, and is thus misplaced in a model of language.

However, the results obtained with gold tags belie this explanation: here we saw that the transition groups were used to distinguish transitions in some sentence type, while clustering other transitions together. These BHMM-TG models found higher probability solutions than either the BHMM or the BHMM-T, showing that with good tags, the transition groups are beneficial.

On the other hand, the models with inferred tags had various degrees of transition group sharing, which corresponded closely to their performance on the VM task: with more sharing, their performance likened the BHMM baseline, and with less, the BHMM-T baseline. Empirically the BHMM-T seemed to be an upper bound for BHMM-TG performance when doing joint inference over both tags and transition groups, contrary to what we hypothesised.

Insufficiently strong prior over groups Increasing the amount of data, by using the Manchester dataset, greatly decreased the amount of sharing of transition groups.

Overall, given the BHMM-TG structure, splitting the transition groups led to higher posterior probabilities, but lower emission posteriors and lower VM performance. This indicates that the model is in a sense ‘overfitting’ to the sentence-type-specific transition distributions, by placing too much weight on modelling the transitions. A stronger prior might counteract this tendency; unfortunately it is unclear how to add such a stronger prior to the model. Given the ‘type-based’ nature of the transition group structure, these counts will always be small compared to the token counts in the transition distributions.

Why not a hierarchical model? An alternative model structure would include transitions as a hierarchical distribution, with bottom-level sentence-type-specific transitions, as in the BHMM-T, and a top-level general transition distribution across all sentence types in a backoff distribution. Each tag and sentence type would have an individual hyperparameter determining the amount of backoff to the general transition distribution. In this case, if both question types had similar transitions for a particular tag, whereas the declarative transitions were dissimilar, we would predict the question types to have a high degree of backoff to the shared distribution, but the declarative transitions would be based mainly on a sentence-type-specific transition with little backoff. When using only three sentence types, all partitions can be represented in the hierarchical model. However, with more than three sentence types, the hierarchical model cannot model partitions in which two sets of sentence types share distributions, such as e.g., WQ,SD.

Mismatch between posterior probability and VM tag performance On several occasions we observed a mismatch between the model with the highest overall posterior probability and the model with highest VM tagging performance. In each of these cases, the model with the highest VM also had higher posterior emission probabilities, which were countered by lower transition and transition group posteriors. This suggests that focussing on the transition structure of the BHMM is less likely to lead to improved tag performance, and accordingly, in the next chapter we refocus on extending the emission distribution of the BHMM.

4.5 Conclusion

In this chapter we presented the BHMM with transition groups (BHMM-TG), which is a generalisation over the BHMM and the BHMM-T from previous chapters. The goal of this model structure was to enable flexible use of the sentence type cue. This was effective when inferring only the transition groups with fixed tags. However, when inferring both transition groups and tags we found that the BHMM-TG models did not consistently outperform the BHMM-T model, in terms of finding better part of speech tags, despite the added descriptive power.

Chapter 5

Joint learning of morphology and syntactic categories

Children acquire morphology and word order roughly in parallel. In English, the question as to which comes first seems to be child-dependent (Clark, 2003b). While Brown's (1973) stages of language development posit the acquisition of the first multi-word combinations in Stage I, with morphology appearing only in Stage II, some particularly salient morphemes may be acquired earlier (e.g., in Mervis and Johnson (1991) a child in the one word stage used the plural morpheme reliably). The typology of the language being learned influences the time-course of acquisition: learners of morphologically rich languages become productive in morphology earlier (Xanthos et al., 2011). Experiments testing sentence comprehension in children also demonstrate cross-linguistic differences in the cues used to make sense of confusing or non-canonical sentence structures: children learning Turkish, for example, rely on morphological affixes, regardless of word order (Slobin, 1982), whereas learners of Japanese rely on the presence of grammatical particles as well as word order (Hakuta, 1982). Faced with the task of enacting nonsense sentences, Italian learners disregard word order in favour of more coherent semantics, while English learners do the opposite (Bates et al., 1984).

This interaction between morphology and word order gives rise to the hypothesis that a general model of syntactic category acquisition must take morphology into account, in order to accommodate languages with rich morphology. Vice-versa, distributional word order information must be included in an entirely from-scratch model of morphology, in order to distinguish between syntactic categories (and the associated morphological processes) in languages in which morphology alone is not sufficient to disambiguate categories.

In this chapter, we introduce a model that aims to operate on this level of typological generality, by integrating both morphology and word order. The BHMM from earlier chapters relies solely on distributional context, i.e. word order (which really specifies syntactic category order, rather than words *per se*), for syntactic categorisation. The transition distribution carries the information about the probability of a cluster (hidden state) in a given context, whereas the emission distribution in the BHMM is a simple multinomial, principally ensuring that tokens of the same word type will be more likely to be members of the same cluster. This model is effective for languages with strict word order, such as English, where, having seen a determiner such as *the*, we can safely predict the next word to be a noun, or perhaps an adjective: nearly every other syntactic category would result in an ungrammatical sentence, due to not obeying word order constraints (e.g., *the would*, *the terrify*).

Many other languages, however, do not rely as heavily on word order alone to signal syntactic category membership and other grammatical relations. Instead, morphological processes such as affixation, which apply to the words themselves, serve as signalling functions (e.g., in English the morpheme *-ness* marks the transformation of an adjective to a noun, such as *kindness*, *weightlessness*). When the syntactic markers are attached to the word using bound morphemes, syntactic information is retained even when words are rearranged. In isolating languages, the same syntactic information is conveyed either using particles, which must stay in a certain proximal relationship with the word they apply to, or via word order directly (e.g., subject is before verb), thus requiring a more rigid word order.

By adding a morphological component to the BHMM, we hope to expand the coverage of the BHMM to languages in which morphology is a richer cue than English. Conversely, since the morphology model is also unsupervised, we hope that adding surrounding context via the BHMM can improve the morphology model, by adding cues to the applicable (syntactic category-appropriate) morphological processes. The syntactic category of a word determines its morphological behaviour (to the extent that the word is regular; it cannot use morphology from a different syntactic category). Knowing which morphemes occur in a word is likely to aid syntactic categorisation: words ending in *-y* are likely to be adjectives, but if a word ending in *-y* appears in a non-adjectival context, we should not classify it as an adjective, and also not segment the *-y* ending of that word as an adjectival suffix (e.g., in *the shaggy dog story*, *shaggy* should be identified as an adjective and be segmented as *shagg-y* but *story* should not).

We add a model of morphological segmentation (Goldwater et al., 2006) to the emission distribution of the BHMM, so that each tag in the data sequence emits not a word, but a stem and a suffix which combine to form the observed word. This is a rather simple concatenative morphology but is suited to (and commonly used for) English and Spanish, the two languages we study in this chapter. Of the two, Spanish has a much richer morphology, and we find that leads to different behaviours with regards to single-task baselines.

Additionally, the model uses a representation of the input that is in line with how children hear and manipulate language: it is based on tokens, allowing ambiguous words (e.g., *he laughs* vs. *her laughs*) to be treated correctly, as distinct forms. Surprisingly few joint models have been proposed to take advantage of the interaction between morphology and syntactic categorisation, and these have included a word-type constraint (Lee et al., 2011b; Can, 2012; Sirts and Alumäe, 2012): each wordform in the corpus (i.e., *laughs*) is limited to a single part of speech and a single morphological analysis. This leads to easier inference, but ignores the true nature of language, which includes plenty of ambiguity, both in terms of words with multiple syntactic categories and in terms of words with ambiguous morphology.

In the following sections, we first discuss previous models of unsupervised learning of morphology and connections between models of morphology and parts of speech. We then describe the morphological component of our model, and discuss how to integrate it into the BHMM. Finally, we present experimental results on English and Spanish, and discuss how their typological differences leads to different patterns of learning within the model.

5.1 Related Work: Unsupervised Morphology

Learning morphology in an unsupervised fashion has been the focus of a considerable amount of research (see Hammarström and Borin (2011) for a detailed overview). Morphological structures and patterns vary considerably between languages, ranging from affixation (e.g., *-ed* for English past tense) through apophony (e.g., ablaut in irregular English verbs such as *shine/shone*) to transfixation (e.g., Arabic morphology).

For the most part, research in unsupervised morphology induction has focussed on affixative morphology, and we continue in this line. From this perspective, learning a morphology can be viewed as the task of segmenting words into reusable components, i.e. discovering the set of morphemes, the lexicon, that are sufficient to generate the words in the language.

A trivial solution is to have a morpheme lexicon that consists simply of the set of all word types; by definition this lexicon will cover the language. This is in fact the Maximum Likelihood solution (Goldwater, 2007). However, storing *open, opens, opened, close, closes, closed* as individual words is intuitively less efficient than representing these words as the combination of the stems (*open,close*) and suffixes (*-NULL, -s, -ed, -d*). Additionally, from an acquisition perspective, this latter representation is able to explain humans' ability to extend morphology to new words, e.g. to generalise *wugs* from *wug*; under the whole-word representation, each word would have to be learned separately.

Minimal Description Length (MDL) and Bayes' Rule provide two formalisations of this notion of the desirability of parsimony within the lexicon, either in terms of minimising a cost or maximising a probability. In the MDL framework, the total cost of a posited lexicon is calculated as the sum of the length of the lexicon and the total length of all pointers to the lexicon required to generate the language data. Length is measured as 'compressed length', the number of bits necessary for an optimal encoding of the lexicon in an information-theoretic sense. The most well-known morphology induction system using MDL is Linguistica (Goldsmith, 2001, 2006). One weakness of the MDL framework is that it only gives a method of assessing the cost of a proposed lexicon; it lacks a methodology for generating new proposals. Linguistica and other MDL-based models (Brent and Cartwright, 1996; de Marcken, 1996; Creutz and Lagus, 2002) depend on heuristics in their search for a good lexicon.

MDL is equivalent to using MAP to evaluate a probabilistic Bayesian model (Wallace and Freeman, 1987; MacKay, 2003); in this latter case, we are interested in the most probable lexicon $\mathcal{L}$ given the seen language data $\mathcal{D}$. We calculate the probability of a lexicon using Bayes' Rule, $P(\mathcal{L}|\mathcal{D}) \propto P(\mathcal{L})P(\mathcal{D}|\mathcal{L})$: and see that, analogously to MDL, we must take into account both the prior probability of the lexicon, $P(\mathcal{L})$, and the probability of the data generated by the lexicon $P(\mathcal{D}|\mathcal{L})$. (The Kraft-McMillan theorem gives us the equivalency between MDL's optimal code length and probability distributions.) The Morfessor system (Creutz and Lagus, 2004, 2007) and the ParaMor system (Monson et al., 2007, 2008) use MAP in a similar way as Linguistica uses MDL, to evaluate a greedy search procedure.

More fully Bayesian models use more general inference techniques, such as sampling, to explore the search space in a less constrained or *ad hoc* manner. Goldwater et al. (2006) investigate the effect of input type on morphology inference. Most morphology induction systems have used word lists as input data. This corresponds to learning from types, rather than the token input heard by children, and assumes that word frequency is irrelevant to morpheme learning. There is some evidence that morpheme type frequency is attended to in acquisition (Bybee, 1995), and Goldwater et al.'s (2006) results supported this type-based assumption as well: a model that took token frequencies into account performed worse than a model which used only type frequencies. However, this held only for data that consisted entirely of verbs. When natural corpus data (child directed speech) was used as input (Goldwater, 2007), the word type model performed poorly; instead, a model which dampened corpus frequencies (but still made use of them) performed best. This model provides the morphological component to the model presented in this chapter and will be described in more detail in section 5.3.

Other Bayesian models of morphology learning, many of which also include frequency-damping mechanisms, include Naradowsky and Goldwater (2009), which is an extension of Goldwater et al. (2006), and Johnson (2008), which uses a non-parametric probabilistic grammar formalism (adaptor grammars). In a closely related task, Snyder and Barzilay (2008) and Naradowsky and Toutanova (2011) use Bayesian models to perform morpheme alignment between languages.

The generative perspective, in which the goal is to learn a morpheme lexicon, is only one way of viewing the task of morphology induction. The earliest work in morphology learning, introduced by Zelig Harris (Harris, 1955, 1967), focussed on finding the segmentation points between morphemes, without aiming to find coherent and reusable morphemes. The key idea is that the boundary between morphemes is a place of higher entropy or variation than within a morpheme. Others have used variants of this method with different criteria for setting segmentation points (Déjean, 1998; Bordag, 2006; Bernhard, 2006; Keshava and Pitler, 2006; Demberg, 2007; Dasgupta and Ng, 2007a; Moon et al., 2009), often with a second step of filtering and clustering the found morphemes. This work has principally used word lists as input, but many have used token counts rather than type counts to measure variability.

An alternate formulation focusses on finding clusters of words that are morphologically related, without insisting on a segmentation into morphemes. This allows irregular and non-concatenative morphology to be captured, such as English strong verbs (*run/ran, sing/sang, drive/drove*). While they usually make use of edit distance or other measures of orthographic similarity, such as shared affixes (Neuvel and Fulop, 2002), these models incorporate other features as well, such as semantic relatedness

(Schone and Jurafsky, 2000; Baroni et al., 2002), local (word) contexts (Schone and Jurafsky, 2001), and frequency (Yarowsky and Wicentowski, 2000).

5.2 Related Work: Unsupervised Morphology with Part of Speech Tagging

The clear interaction between morphology and parts of speech has motivated work in one task to make use of features from the other. However, there has been very little work up until quite recently on learning both simultaneously, in a joint model. In this section, we review first models of part of speech tagging which make use of morphological features, then models of morphology which use part of speech categories. Finally we describe three joint models of morphology and POS tagging, and describe how our approach diverges from these.

5.2.1 Morphology in Unsupervised Tagging

Many part of speech categories are marked by morphemes; in English this is mainly evident at word endings, in the form of suffixes (especially in verbs). To take advantage of this fact, many unsupervised POS induction models have incorporated features representing the ending characters of words, e.g. Smith and Eisner (2005); Haghighi and Klein (2006); Berg-Kirkpatrick et al. (2010); Lee et al. (2010).

A less suffix-specific method is to incorporate character-level information over the whole word. Clark (2003a) uses a character HMM to generate each word; for infrequent words the addition of word-internal information is particularly helpful. Adding a character language model to the emission distribution in a non-parametric HMM-based tagging model (Blunsom and Cohn, 2011) leads to large gains in tagging performance in morphologically rich languages (and smaller gains in English).

A small number of part-of-speech induction systems use morphological segmentations learned by a separate morphology model as features: Dasgupta and Ng (2007b) use their own morphology model (Dasgupta and Ng, 2007a) together with context features in a SVM model. Hasan and Ng (2009) add suffixes found by Keshava and Pitler's (2006) system to Goldwater and Griffiths's (2007) tagger in a weakly supervised setting. Abend et al. (2010) use Morfessor (Creutz and Lagus, 2007) to bootstrap a POS induction model from automatically found reliable seed words. In a more orthodox unsupervised POS tagging setting, Christodoulopoulos et al. (2011) use Morfessor suffixes as features within a Bayesian mixture model.

There has been some work on joint morphology and dependency parsing, principally for morphologically complex languages such as Arabic or Latin (Cohen and Smith, 2007; Goldberg and Tsarfaty, 2008; Lee et al., 2011a). These models are supervised and hence not directly relevant to the work presented here. However, they demonstrate the need to incorporate morphology at higher levels of syntactic analysis, especially in languages with richer morphology.

5.2.2 Syntactic Categories in Morphology Models

Surprisingly few morphology models have added part of speech information as input features, perhaps because they often use word lists as input. Many models restrict the input to a few parts of speech (e.g., Goldwater et al. (2006) uses only verbs).

Hu et al. (2005) add POS information from an external (trained, supervised) tagger to Linguistica (Goldsmith, 2001). However, they evaluate only the savings on descriptive length (in an MDL framework), and do not report whether this results in improved segmentation. Can and Manandhar (2010) use categories induced by Clark's (2000) distributional method; it is unclear whether adding categories improves their morphology model. Schone and Jurafsky (2001) use local contexts as a cue to cluster morphologically related words together, on the assumption that words with similar distributional characteristics will also share morphological characteristics.

Models of full morphological paradigm induction (Chan, 2006; Dreyer and Eisner, 2011), in which the set of alternations of a given stem are learned, assume known POS tags in order to separate e.g., noun and verb occurrences of *compress*.

5.2.3 Fully Joint Models

Finally we examine three recent models which jointly learn morphological segmentations as well as syntactic categories.

Lee et al. (2011b) present a Bayesian model of morpheme segmentation that incorporates both a lexicon level, in which words are generated from latent syntactic category-specific morpheme distributions, and a token level, in which words are represented by a HMM sequence of inferred tags and words. Each word type is assigned to a single morphological segmentation and part of speech tag within the lexicon component. The number of latent categories is set to be very small (5), suggesting that only very coarse categories are learned (e.g. perhaps nouns, verb, adjective, 'other'); moreover, they do not evaluate the inferred latent categories against gold part of speech tags. Ablation results show that separating the morpheme generation into category-specific distributions is far more helpful for morphological segmentation than modelling the local syntactic category contexts using the HMM. This suggests that the limited number of categories may be inadequate to model the token tags correctly.

Can (2012) attempts a full joint model, presenting both tagging and morphology results; unfortunately, without ablation results of the single components it is not possible to discover whether joint learning results in improvement. The model description is difficult to follow, but implies that during inference, all tokens of a word type are assigned the same morphological analysis.

Sirts and Alumäe (2012) present a complete Bayesian model of both POS tag and morphological segmentation. However, this model is type-based: each word form is assigned a single segmentation and part of speech tag. This has been shown to improve POS inference performance but violates the known ambiguities of natural language. In this case, the type-based constraint is necessary to make inference tractable, since the model includes multiple levels of non-parametric processes. This model also is able to segment words into multiple (more than two) morphemes; it does not differentiate between stems and affixes. They evaluate on both parts of speech and morphological segmentation across a wide variety of languages, though not on the same datasets for both tasks. Oracle experiments in Estonian, in which either the tags or the morphological segmentation is set to the gold value, show that the joint model is able to improve over the oracle on the other task, an impressive result.

The model presented in this chapter differs from these other models in that it is consistently token-based; we allow word tokens to be assigned to different tags and different morphological analyses. This is in line with the ambiguity occurring in natural language, but could hurt performance if ambiguity within the model is overly rife. As we shall see next, the morphology component in the model includes statistical processes which aim to add type-based behaviour without curtailing ambiguity completely.

5.3 Morphology Model: Goldwater et al. (2006)

In this section we describe the morphological component of the full model, which also includes a tagging component. The morphology model is due to Goldwater et al. (2006) and further details can be found in Goldwater (2007). In the subsequent section this model is added to the BHMM as a tagging component.

The morphology model segments words into stems and (possibly null) suffixes. We first describe the generative process for generating segmentations, then describe how a non-parametric Pitman-Yor Process shapes the underlying generating distribution into one that can reflect both learning from type and token statistics.

5.3.1 Generating analyses

We draw a morphological analysis for each word from the base distribution G. An analysis l has a three components: a cluster label, a stem, and a suffix; the stem and the suffix concatenate to form the word, and together are also called a segmentation of the word. The cluster does not necessarily correspond to a part of speech tag. Within the generative model, the cluster is responsible for generating the stem and suffix; words in the same cluster will be likely to share stems and/or suffixes. As an example, the word *wugs* may be assigned to cluster `3`. The stem and the suffix would be drawn from the stem and suffix distributions associated with cluster `3`.

The probability of an analysis l, which consists of a cluster c, stem s, suffix f, tuple, factors as follows:

$$G(l = (c,s,f)|w) = P(c)P(s|c)P(f|c)[w = s \oplus f] \tag{5.1}$$

where the last term is enclosed by Iverson brackets, which evaluate to one if the statement within is true and zero otherwise (equivalent to an indicator function). This term indicates whether the word w is the concatenation (denoted by $\oplus$) of the stem s and suffix f and thus ensures that the analysis is compatible with the observed word.

Each component of the analysis is drawn from a multinomial with Dirichlet prior. This requires the number of possible stems, suffixes, and clusters to be specified in advance. This is somewhat suboptimal; a more complex model would use nonparametric processes (see for example Sirts and Alumäe (2012)), but inference in such a model would also become very difficult and require approximations (Blunsom and Cohn, 2011). We set the number of possible morpheme types (stems and suffixes) to the number of possible morphemes (unique word-beginning or -ending substrings) in the data, to minimize the amount of prior knowledge. The Dirichlet hyperpriors are accordingly set to very small values to encourage sparseness (i.e. each cluster should only use a small subset of all possible affixes).

5.3.2 Assigning analyses to tokens

One could use the distribution G above to assign analyses to word tokens directly. However, it is unlikely to perform well, due to the independence assumptions underlying the factorisation between stem and suffix. These are too strong: clearly not all stems take all suffixes with equal probability, even if they are in the same cluster (for example, *people* is much less likely to appear with a *-s* suffix than *human*, cf. also Yarowsky and Wicentowski's (2000) *sing* and *singed* example). Additionally, we know empirically that most words of the same type should receive the same morphological analysis. Having to regenerate this analysis multiple times is undesirable.

To ameliorate these problems, a Pitman-Yor process (PYP) (Pitman and Yor, 1997) is added over the base distribution. This stochastic process allows for the production of power-law distributions like those found in natural language.

The Pitman-Yor process can be described using the Chinese Restaurant metaphor (Ishwaran and James, 2003), in which customers are sequentially seated at a restaurant with an infinite number of tables, each of which serves a single dish (but the same dish may be served at multiple tables). The seating arrangement plus the dishes defines a clustering of the customers, with each set of customers at a table belonging to the cluster labelled by the dish at that table. (See (Jordan, 2005; Teh, 2006; Orbanz and Teh, 2011) for introductions to the Pitman-Yor process and the closely related Dirichlet Process.)

To extend this metaphor to the morphology model, we designate the word tokens as customers being seated at tables in the restaurant. The morphological analyses take the role of the dishes being served at the tables within the restaurant.

Token 'customers' are indexed by i; w_i is the i'th word, assigned to table k, which is 'serving' the analysis l_k. We use indicator variables z_i to track the table at which i is seated, i.e. $z_i = k$. More than one token can be assigned to the same table k, resulting in these tokens sharing the same analysis l_k (obviously these tokens have to be of the same word type, in order to share a segmentation). Note that analyses l_k and $l_{k'}$ are equivalent if they share the same cluster and segmentation.

When a word token customer enters the restaurant, it may either be seated at an occupied table serving analysis l_k, with probability proportionate to the number of customers seated at that table, or it may be assigned to a new table, which is then given an analysis drawn from the underlying base distribution G. Formally, the Pitman-Yor process is parameterised as PYP(a, b, G), where $0 \leq a < 1$ and $b > -a$. Each token is assigned in sequence to a table $z_i = k$. After i tokens have been seated, creating a set of table assignments $\mathbf{z}$, token $i+1$ is assigned a table k with probability:

$$P(z_{i+1} = k|\mathbf{z}) = \begin{cases} \frac{n_k - a}{i+b} & \text{if } 1 <= k <= K\text{, i.e. if } k \text{ is already occupied} \\ \frac{Ka+b}{i+b} & \text{if } k > K\text{, i.e. if } k \text{ is a new table} \end{cases} \tag{5.2}$$

where K is the total number of occupied tables before $i+1$ is seated and n_k is the number of customers seated at table k.

As the above equation shows, a popular table (where n_k is large) is more likely to attract more customers, especially when a is small: this results in the ‘rich get richer’ dynamics characterising the PYP and DP. Conversely, when a is large (near 1) customers are more likely to be seated at new tables, and increasingly so as K grows. A large b also increases the likelihood of drawing a new analysis, but usually has less effect than the a parameter. (Note that when $a = 0$, b is equivalent to α in a Dirichlet process: $\text{PYP}(0, b, G) = \text{DP}(b, G)$.)

Thus, depending on the parametrisation, adding a PYP to the morphology generating distribution G can achieve the effect of having most tokens of a word type reuse the same analysis, instead of drawing a new analysis from the base distribution. This leads to a ‘type’-based analysis, i.e., one in which the base distribution G is used to generate analyses for only a minority of word tokens, which are then cached and reused by the PYP. Clusters of tokens thus share the same analysis, in the same way that all tokens share the same analysis in a word type-based lexicon. However, it is important to note that the PYP may create more than one such analysis for a single word type.

Goldwater et al. (2006) applied this model to a corpus (not a word list) of verb forms and found good performance at small and zero values of a (with $b = 0$). However, Goldwater (2007) found that when learning the morphology of a phonetically transcribed child directed speech corpus, the smallest (zero) values of a and b performed poorly: they lead to oversegmentation. For this reason in our experiments we do not commit to a type-based morphology model (with $a = b = 0$, which is technically undefined) and instead experiment with a range of hyperparameters.

5.4 Full Model: Morphology + Tagging

The morphology model on its own is only aware of word-internal patterns — commonly occurring suffixes and stems. However, the presence of morphological ambiguity and syncretism (e.g. -*s* as plural noun or third person singular marker) means that these patterns are not always sufficient to distinguish between morphemes. By adding local context and modelling the syntactic categories of words, we hope to separate syncretic morphemes.

Separating words into categories has a second potential advantage: if the categories are of high quality, meaningful patterns such as final *-s* should become more prominent than noisy patterns. For example, pairs such as *won/wont, run/runt, plan/plant* suggest a *-t* suffix, but if words are separated according to syntactic category these pairs do not appear.

In order to incorporate local syntactic content into the morphology model, we combine it with the BHMM used for tagging (see Section 2.3.3). Each syntactic category is associated with a distinct morphology model, i.e. a separate PYP with a separate underlying base distribution G_t, generating only the analyses of the words belonging to that syntactic category. The tokens are generated in sequence, with the tag generated conditioned on local history (as in the BHMM) and the word token then generated by the tag-specific morpheme model.

In more detail, the generative process is thus: first a tag t is drawn based on the transition distribution from the previous two tags (line 5.4 below). Then a morphological analysis is drawn from the PYP associated with that tag, in the same manner as in the morphology model: each token is given a table assignment, $z_i = k$, and that table carries the analysis l_k (line 5.5). The concatenation of the stem and suffix of the analysis produces the observed word w_i (line 5.6).

$$\tau_{(t,t')} \sim \text{Dirichlet}(\alpha) \tag{5.3}$$

$$t_i = t | t_{i-1} = t', t_{i-2} = t'', \tau \sim \text{Mult}(\tau_{(t',t'')}) \tag{5.4}$$

$$z_i = k | t_i \sim \text{PYP}_t(a, b, G_t) \tag{5.5}$$

$$w_i = l_k.\texttt{stem} \oplus l_k.\texttt{suffix} \tag{5.6}$$

The morphology model also generates a latent morphology cluster l_k.`cluster`; we generally set the number of clusters to 1, removing it as an informative variable from the model. This results in a single stem and suffix distribution for each tag (i.e., within each G_t).

The full joint posterior distribution of a sequence of words, tags, and morpheme label assignments is:

$$\begin{aligned}
P(\boldsymbol{w},\boldsymbol{t},\boldsymbol{k},\boldsymbol{z}|\alpha_t,a,b,\boldsymbol{G}) &= P(\boldsymbol{t}|\alpha)P(\boldsymbol{w},\boldsymbol{k},\boldsymbol{z}|\boldsymbol{t},a,b,\boldsymbol{G}) && (5.7)\\
&= \prod_{i=2}^{N} P(t_i|t_{i-1},t_{i-2},\alpha)P(z_i|t_i,\boldsymbol{k},a,b,\boldsymbol{G})P(w_i|z_i)\prod_{t=1}^{T}\prod_{k=1}^{K_t} P(l_k|G_t) && (5.8)\\
&= \prod_{t,t',t''=1}^{T} \frac{\Gamma(n_{tt't''}+\alpha)}{\Gamma(1+\alpha)}\frac{\Gamma(1+T\alpha)}{\Gamma(n_{tt'}+T\alpha)} && (5.9)\\
&\times \prod_{t=1}^{T}\frac{\Gamma(1+b)}{\Gamma(n_t+b)}\prod_{k=1}^{K_t}(ka+b)\frac{\Gamma(n_k-a)}{\Gamma(1-a)}G_t(l_k)\prod_{i=2}^{N}P(w_i|z_i) && (5.10)
\end{aligned}$$

Note that $P(w_i|z_i = k) = 1$ if $w_i = s_k \oplus f_k$ and 0 otherwise. The first factor in 5.9 is the transition between tags; the remainder in 5.10 are the factors for the tag-specific Pitman-Yor assignments $\boldsymbol{z}$ to tables $\boldsymbol{k}$ with labels $\boldsymbol{l}$ (compare to Goldwater (2007), Equation 4.3). G_t is calculated using Equation 5.11 below. We add two dummy tokens at the start, end, and between sentences to pad the context history.

Note that all tag-specific morphology models share the same Pitman-Yor parameters a and b. This is not strictly necessary, but keeping a and b constant between all tags results in fewer hyperparameters to set or estimate. This constraint could be relaxed in future work.

The posterior distribution of the tag-specific base distribution G_t is exactly the same as in the original morphology model, only split between tags. It is a mixture of Dirichlet multinomial posteriors representing the cluster c, stem s, and suffix f distributions, with respective κ, σ, and ϕ hyperparameters (see Section 2.3.3):

$$G_t(l = (c,s,f)) = \frac{m_{tc}+\kappa}{m_t+C\kappa} \times \frac{m_{tcs}+\sigma}{m_{tc}+S\sigma} \times \frac{m_{tcf}+\phi}{m_{tc}+F\phi} \tag{5.11}$$

where m_t is the number of tables in the PYP associated with the tag t that share the cluster/stem/suffix in question.

5.4.1 Notes on the Model

5.4.1.1 Non-Parametrics

Although the key component of the morphology model is non-parametric, other components of the model (transitions, base distributions) use finite distributions which require specifying the size of e.g. the tagset or the number of possible stems in advance. We leave extending the model to include more non-parametric components for future work. For example, replacing e.g. the transition component with a infinite HMM

(Gael et al., 2009) would allow us to avoid specifying a fixed number of tags. However, inference, which is already non-trivial in the standard iHMM (Gael et al., 2008), would become significantly more complex when we add the PYP morphology emissions. Blunsom and Cohn (2011) present an approximation for a significantly more straightforward hierarchical PYP model with a fixed number of tags; these ideas could potentially be used in future work to create a non-parametric version of this model.

5.4.1.2 Hierarchical PYPs and Shared Stems

We use separate PYPs for each tag without a common backoff PYP, i.e. not a hierarchical model. This is justified by the tag-specificity of the suffix distribution: since suffixes apply to only a single syntactic category, there is no reason to have a hierarchical model 'top level' with a shared suffix distribution. It is unclear what, for example, having a *-ly* suffix common to all syntactic categories would achieve, other than cause confusion.

The motivation behind split tag-specific stem distributions is less clear. Arguably, derivational morphology implies that the same stem *quick-* should be used to generate the adverb *quick-ly* as well as the adjective *quick*. Preliminary experiments with a single stem distribution shared between all tags did not yield good results. A hierarchical stem distribution, with both a general and tag-specific component, might perform better; we leave this idea to future work.

5.4.1.3 Morphological Clusters as Tags

We make a distinction between the morphological clusters responsible for generating stems and suffixes and the syntactic tags used in the transition distributions.

A different possible model structure would consider the clusters in the base distribution G to be syntactic categories, and try to add local context information into the morphology model, rather than having a separate tagging component. However, it becomes very difficult to integrate transitions (or other forms of context information) into such a model. Essentially, the difficulty is in integrating token components of the model (tag sequence, table assignment sequence) with the 'type' components (word and morphological analyses derived from the table label, which may be shared by many tokens seated at the same table). This may be avoided by stipulating that the token components also follow type constraints (i.e. that each word type only have a single possible tag), in the vein of Lee et al. (2011b) and Sirts and Alumäe (2012).

However, we wish to retain an explicitly token-based model and thus do not try these variants.

5.5 Inference

We use Gibbs sampling for inference over the three sets of discrete variables: tags $\boldsymbol{t}$, table assignments $\boldsymbol{z}$, and labels/analyses $\boldsymbol{l}$. Note that two of these sets of variables, tags and table assignments, are associated with the word tokens, whereas the labels are associated with the set of tables (sat at by the tokens). The sampler has two stages: Firstly the token variables — tags and table assignments — are resampled, based on the current values of all other variables. Secondly, the labels on the tables are resampled.

5.5.1 Initialization

First, the tags are initialized uniformly at random from the set of possible tags using the same initialization process as in the BHMM.

Secondly, for each token, a segmentation and a morphological cluster is chosen, also uniformly at random. (We generally disallow segmentations with a null stem.) The combination of these gives us an analysis (segmentation and cluster). If this analysis is new within the tag's morphological PYP, a new table is created for the token. If there are existing tables with this analysis, the table assignment is sampled from amongst the existing tables and a possible new table.

An alternate initialization procedure would be to initialize using the table assignment sampling procedure: Given a tag, sample a table for each token, which then receives the label on that table. This method has the disadvantage that it will be prone to assign most subsequent words to the same initial table, meaning that most tokens of a word will be initialised with the same segmentation. This creates a local maximum that is later difficult to mix out of.

5.5.2 Tags

As in the BHMM, tags are sampled using a collapsed Gibbs sampler (see Section 2.3.5). Tags are sampled from the product of posteriors of the transition and emission distributions. The transition distribution is a Dirichlet-multinomial, identical to the BHMM. However, the emission distribution is no longer a simple Dirichlet-multinomial, since it is a tag-specific PYP. Instead, calculating the marginal probability of the word within the PYP involves summing over the probability of all the existing tables in the given PYP that emit the correct word (summing over all segmentations and clusterings), plus the probability of a new table being created, which also includes the probability of the new analysis from G_t (which itself is a sum over the probability of all possible segmentations and clusterings as labels).

More formally, tags are sampled from the following distribution:

$$\begin{aligned} p(t_i = t | w_i, \boldsymbol{t}^{\backslash i}, \boldsymbol{k}, \alpha) &= p(t | t_{i-1}, t_{i-2}, \alpha) \times p(w_i | t, \boldsymbol{k}) & (5.12) \\ &= p(t | t_{i-1}, t_{i-2}, \alpha) \times \Big(p(k_{\text{new}} | t, w) + \sum_{k \text{ s.t. } l_k = w_i} p(k | t, w)\Big) & (5.13) \end{aligned}$$

We omit for clarity the transition terms required to capture the overlap in contexts. They are the same as in the BHMM (see Equation 2.34).

5.5.3 Table Assignments

Table assignments are sampled from the Pitman-Yor distribution over all tables with labels compatible with w_i, plus a possible new table. A new assignment z_i is drawn according to:

$$p(z_i = k | t_i = t, \boldsymbol{w}, \boldsymbol{z}^{\backslash i}, a, b) = \begin{cases} \frac{n_k - a}{n_t + b} & \text{if } 1 <= k <= K_t \\ \frac{K_t a + b}{n_t + b} G_t(l_k) & \text{if } k > K_t \end{cases} \qquad (5.14)$$

where n_k is the number of words assigned to table k and K_t is the total number of tables produced by the tag t's PYP. (Both of these are sufficient statistics based on $\boldsymbol{z}^{\backslash i}$, which we omit from the equations in the interest of legibility.)

$G_t(l_k)$ is calculated by summing over the probability of all possible table labels for a new table for word w_i. If a new table is drawn ($k > K_{t_i}$) then we also sample a new label using the table label sampling distribution (see Equation 5.15 below).

5.5.4 Table Labels

After all tags and table assignments have been sampled, the label on each table, representing the morphological analysis for the words seated at that table, is resampled. It does not condition on token counts; rather, it is a mixture of three posterior Dirichlet-multinomials that track the counts of the number of tables m produced by the generator.

Table labels are sampled as follows:

$$p(l_k = (c, s, f) | l_{\backslash k}, \kappa, \sigma, \phi) = \frac{m_{t,c}^{\backslash k} + \kappa}{m_t^{\backslash k} + K\kappa} \times \frac{m_{t,c,s}^{\backslash k} + \sigma}{m_{t,c}^{\backslash k} + S\sigma} \times \frac{m_{t,c,f}^{\backslash k} + \sigma}{m_{t,c}^{\backslash k} + F\phi} \qquad (5.15)$$

where m refers to the number of tables, not tokens, that share a component (cluster, stem, suffix) of the analysis l_k (see Equation 5.11).

5.6 Evaluation

Our model infers both tags and morphological segmentations, and we evaluate its performance on both. Tags are evaluated as in the BHMM, using VM; see Section 2.4 for a complete discussion on tag evaluation and VM in particular. Note that VM is an evaluation measure for clusters; it is not specific to syntactic categories. In this chapter, we will also use it for morphological evaluation.

Despite the fact that our model's principal task in terms of morphological analysis is setting the correct segmentation point, we do not evaluate the segmentation point itself. The corpus we use, from the `CHILDES` corpora collection, includes morphological annotations that do not allow a gold segmentation point to be determined. Each word is annotated as a stem and a label encoding the affix type. For example, *running* is annotated as *run*-`PROG`. This scheme is agnostic about the segmentation point, i.e., whether it should placed so as to have a consistent *run* stem (but a rare *-ning* suffix), or whether to emphasise a consistent *-ing* gerund suffix. (See Goldsmith (2006) for a general discussion on the difficulty of setting gold segmentation points.)

We thus evaluate the model output in terms of clusters: to what extent do the sets of words that share a suffix proposed by the model correspond to the sets of words that share a gold standard suffix? When evaluating, we also take into account the word's cluster or tag: we distinguish between the *-s* suffix in tag `4` and the *-s* suffix in tag `9`. In the best case, one might correspond to the plural suffix (gold `-PL`) and the other to third person singular (`-3S`), so it is important to keep them distinct. On the other hand, if the model has created multiple noun clusters, this will hurt suffix VM performance as well as tag VM performance. We do not separate null suffixes between tags, to avoid inserting too much tag evaluation into our morphological evaluation measure.

Note that the abstract gold suffixes will create clusters of words that the segmentation model might be unable to cluster together, if an abstract suffix corresponds to a set of surface suffixes that do not share strings. For example, both *dolly* and *doggie* are annotated with the `-DIM` suffix (*doll*-`DIM` and *dog*-`DIM`), but these words cannot share a suffix within our model. Ceiling suffix VM performance for the model is thus below 100.

We also investigated evaluating the clusters created by stems. However, in the languages we ran experiments on (English and Spanish), most word tokens have a null suffix. In this case, the stem clusters become trivial, as long as the model does not exorbitantly oversegment. We did not see this behaviour — instead, stem VM measures were near ceiling and hence uninformative, and we do not report them.

As an alternate method of evaluation we use EMMA (Evaluation Metric for Morphological Analysis) (Spiegler and Monson, 2010). This metric has been designed for the Morphochallenge competition (Kurimo et al., 2010), which evaluates word lists rather than corpora data. EMMA uses an integer linear program to compute the optimal mapping between gold standard morphemes and the morphemes proposed by the model. Unlike the VM metric above, EMMA evaluates word types, instead of tokens. It allows for multiple possible analyses for each word type but does not take their frequency into account in the evaluation. It captures syncretism by separating clusters of identical morphemes by the gold standard disambiguated morphemes, but also expects allomorphs to be clustered together, which our model cannot do, since we cluster based on surface strings. When evaluating using EMMA, we also add tag or cluster labels to the stems and suffixes found by the models, as with suffix VM.

5.7 Data

For English, we use the Eve dataset from CHILDES (MacWhinney, 2000) (see Section 3.1 for details). As before, we remove all child utterances, leaving only adult utterances, which are almost uniformly CDS. These corpora have annotated POS tags; as before we collapse the full set to a coarser set of 13 tags. The coarser tags gather e.g. verbs together, rather than separating past and present tense verbs. Participles and gerunds are in a separate category from verbs. This may not correspond with optimal clustering in the morphology model, since many gerunds and verbs will share a common stem (*eating*, *eats*), leading to pressure to put them in the same tag (sharing the same morphology model); on the other hand there will be counter-pressure to separate them due to distinct transitions patterns for gerunds and verbs.

These corpora have also been annotated at the morphology level: each word is annotated with its ‘base’ form plus a code representing the affix, as described earlier (e.g., *runs* is annotated as *run*`-3S`, *plums* as *plum*`-PL`). We ignore irregular/non-affixing forms annotated with *&* (e.g. *was*, annotated as *be&PAST*) and use only hyphen-separated suffixes to evaluate. Where there are multiple suffixes concatenated together (e.g.*dog*`-DIM-PL`) we treat this as a single suffix (`-DIM-PL`) for purposes of evaluation.

The English corpus has 17 gold suffix types.

We also test on the Spanish Ornat CDS corpus from CHILDES (Ornat, 1994), first presented in Section 3.5.1, that has been annotated with POS tags and morphological analyses in the same way as the English corpora. The Spanish data has significantly richer morphology, with 83 gold suffixes. Verb suffixes in particular also have multiple surface suffixes (allomorphs) associated with each gold suffix, corresponding to the different conjugations (e.g. `-INF` has *-ar*, *-er*, and *-ir* surface forms).

5.8 Baselines and Oracle Models

We test a variety of models to examine the effect of adding local context and morphemic representations to the model.

MORPHTAG is the full model as described in Section 5.3.2.

MORPHCLUSTERS refers to the original morphology-only, single PYP model of Goldwater et al. (2006), without the BHMM. This model separates words into clusters based on morphological patterns but does not use local syntactic contexts. We set the number of clusters to be equal to the number of tags, and we measure tag performance using these cluster identities. (Note that these clusters were not intended originally to represent syntactic parts of speech; we use this as a baseline to investigate the informativeness of morphology alone.)

MORPHTAGNOTRANS is the full morphology+tagging model but without transitions between tag tokens. Tag membership is estimated using word-internal patterns only, as in the MORPHCLUSTERS. Comparing the performance of MORPHTAG against both of these baseline models thus evaluates the effect of adding local syntactic context to MORPHTAG. The MORPHTAGNOTRANS model also demonstrates the effect of splitting the original MORPHCLUSTERS model into multiple independent PYPs. We evaluate tagging performance using the tag tokens; the number of clusters available to the morphological analyses is set to 1 (as it is in all the subsequent models), making this variable uninformative.

MORPHTAGNOSEG is a model in which the only available suffix is the null suffix; thus segmentations are trivial and only tags (and table assignments) are inferred. This model is approximately equal to the BHMM but with the addition of a PYP within the emission distribution. We also evaluate against tags found by the BHMM.

MORPHTAGTRUETAGS is the full model, including transitions between tag tokens, but all tags are fixed to their gold values. This gives us oracle results for the morphology+tagging model. (Due to the annotation scheme used in CHILDES, oracle morphological segmentations are unavailable, so we were unable to test a model with gold morphology and inferred tags.)

In all models and all experiments, the variable evaluated against the gold tag (usually the tag variable except in the MORPHCLUSTERS model) is set to have the same number of possible values as the number of gold tags in the data. Except for the MORPHCLUSTERS, all models have the number of morphological clusters set to 1; in MORPHCLUSTERS the number of clusters is set to the number of gold tags.

5.9 English Experiments

Our English experiments are all conducted using the Eve corpus from CHILDES, described earlier. We train all models using the same train/development/test split described in Section 3.3.1. In this case, we use the development data to evaluate the best values for the a and b hyperparameters in the Pitman-Yor process in the morphology model. The best hyperparameter setting found for the development data is then tested on test data.

The settings for a we tested are biased towards small values, since Goldwater (2007) showed that values closer towards $a = 1$ tend to undersegment (or rather, tend to place all probability mass on the null suffix). We test $a = 0.00001, 0.1, 0.3, 0.5$ and 0.9 and $b = 0, 0.0001, 0.01, 0.1, 1, 10, 30, 100, 300$. The b hyperparameter generally has less effect on model performance. Very large values will lead to more tables in the PYP, similar to larger a values.

There are a number of other hyperparameters in the model which we set to fixed values. The transition hyperparameter α in the BHMM is set to 0.1 in all models. We set the hyperparameters within the morphology base distribution G to the following values (see Equation 5.11: cluster hyperparameter $\kappa = 0.5$, stems hyperparameter $\sigma = 0.001$, suffixes hyperparameter $\phi = 0.001$. This encourages sparsity in the affix distributions and a more uniform distribution over clusters. The number of possible stems and suffixes is given by the dataset: in the train+dev set in Eve, the number of possible stems (i.e. unique prefix strings) is 5351 and the number of possible suffixes (unique suffix strings) is 3727. In the train+dev sets these are 5339 and 3708, respectively.

Sampling is run for 5000 iterations. Inspection of the posterior log-likelihood indicates that the models converge after about 1000 iterations. We run inference over

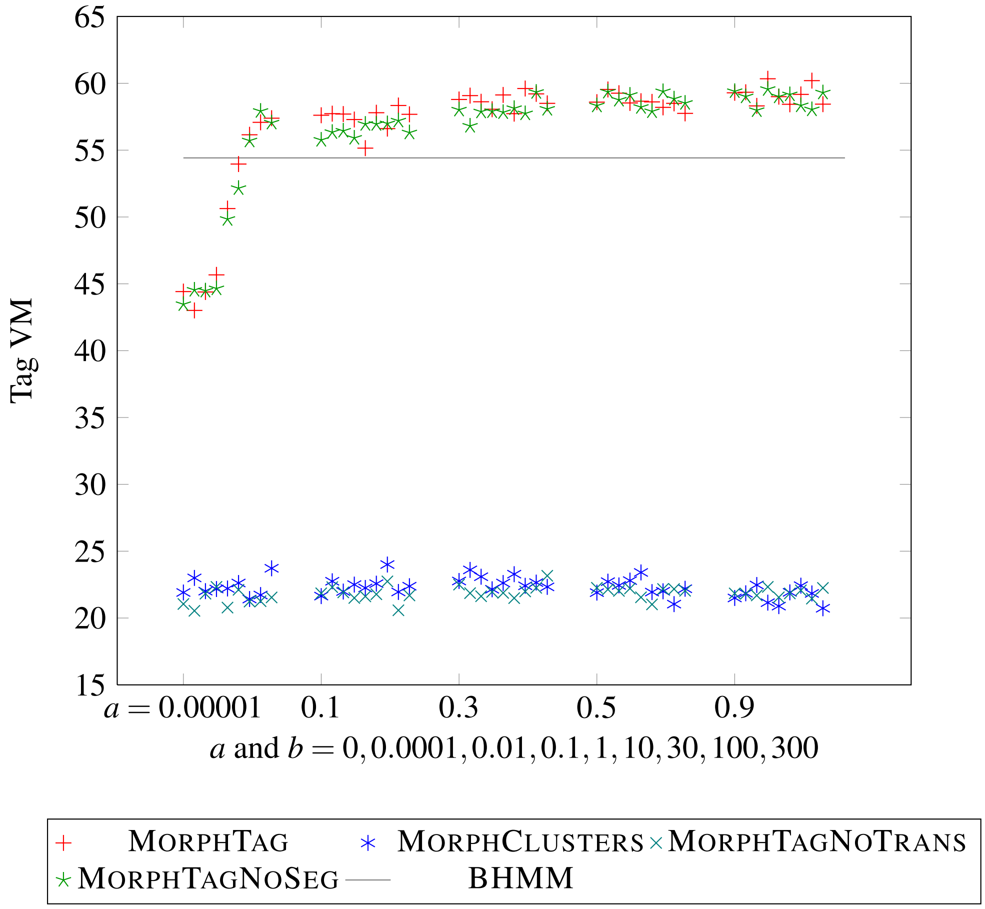

Figure 5.1: Eve Development Hyperparameter Settings: Tag VM
For each setting of a, results for all settings of increasing values of b follow.

all models ten times and report the average performance. We use the non-parametric Wilcoxon rank-sum test for significance testing and a significance level of $\rho < 0.05$.

5.9.1 Tags

Tagging results on the English development set evaluated using VM are shown in Figure 5.1 for a variety of a and b settings. We also show BHMM performance, which does not vary with a and b since it does not include a PYP.

We see a dramatic difference in the quality of tag clusters found by models with local context (MorphTag, MorphTagNoSeg, BHMM) and models clustering words using only morphological information (MorphClusters, MorphTagNoTrans). Both of these latter models lack local context in the form of transitions and result in bad clusters. There is no significant difference between the clusters found by the original morphology model, with a single PYP with multiple clusters, and the full model without transitions. (The random clusters baseline tag VM is 0.2: the morphology models are learning clusters that are significantly better than random.)

Adding the PYP to the emission distribution allows the MORPHTAG and MORPHTAGNOSEG models to improve over the BHMM. However, the morphological analysis in MORPHTAG does not seem to help, since the baseline MORPHTAGNOSEG model without (informative) segmentations performs about as well as the full model. The difference is due to the Pitman-Yor process modelling the data statistics better than the Dirichlet-multinomial used in the BHMM. For some hyperparameter settings — at $a = 0.1$ and $a = 0.3$ — the full morphology model improves over the model without segmentations, but only very slightly.

These results are consistent with the similar PYP-HMM model proposed for unsupervised part of speech tagging by Blunsom and Cohn (2011). They use a character level language model in the emission distribution to capture morphological cues (but they do not include an explicit model of morphology). They find that including this language model leads to only small gains in English (on a newswire corpus that has a greater variety of morphological patterns than CDS), but much larger gains in other languages with richer morphology, notably Arabic and Spanish.

In general, we see that tagging performance is fairly stable over most values of a and b, except when a is set to very small values (and even then, a large b can compensate in terms of smoothing; this is close to a DP, where $a = 0$). At small a settings, there is increased pressure within the morphology model to put all tokens of a wordtype at the same table; this will make it very difficult to resample a large number of tokens from one tag to a better tag, since they will prefer to remain at a popular table even in a suboptimal tag.

5.9.2 Morphology

Before turning to the evaluation measures, Figure 5.2 shows statistics about the suffixes found by the models. *Missing Suffixes* are words that have a gold non-null suffix but are assigned a null suffix by the model; *Extra Suffixes* are words that have a null gold suffix but are assigned a non-null suffix by the model. *Correct Null Suffixes* are word with gold null suffixes that have been correctly assigned a null suffix by the model. Finally, *Model Non-Null Suffixes* shows how many words have been assigned a non-null suffix by the model, whether or not these are correct. (There is no good way of deciding what is a correct non-null suffix given the gold annotations.) Note that these are token, not

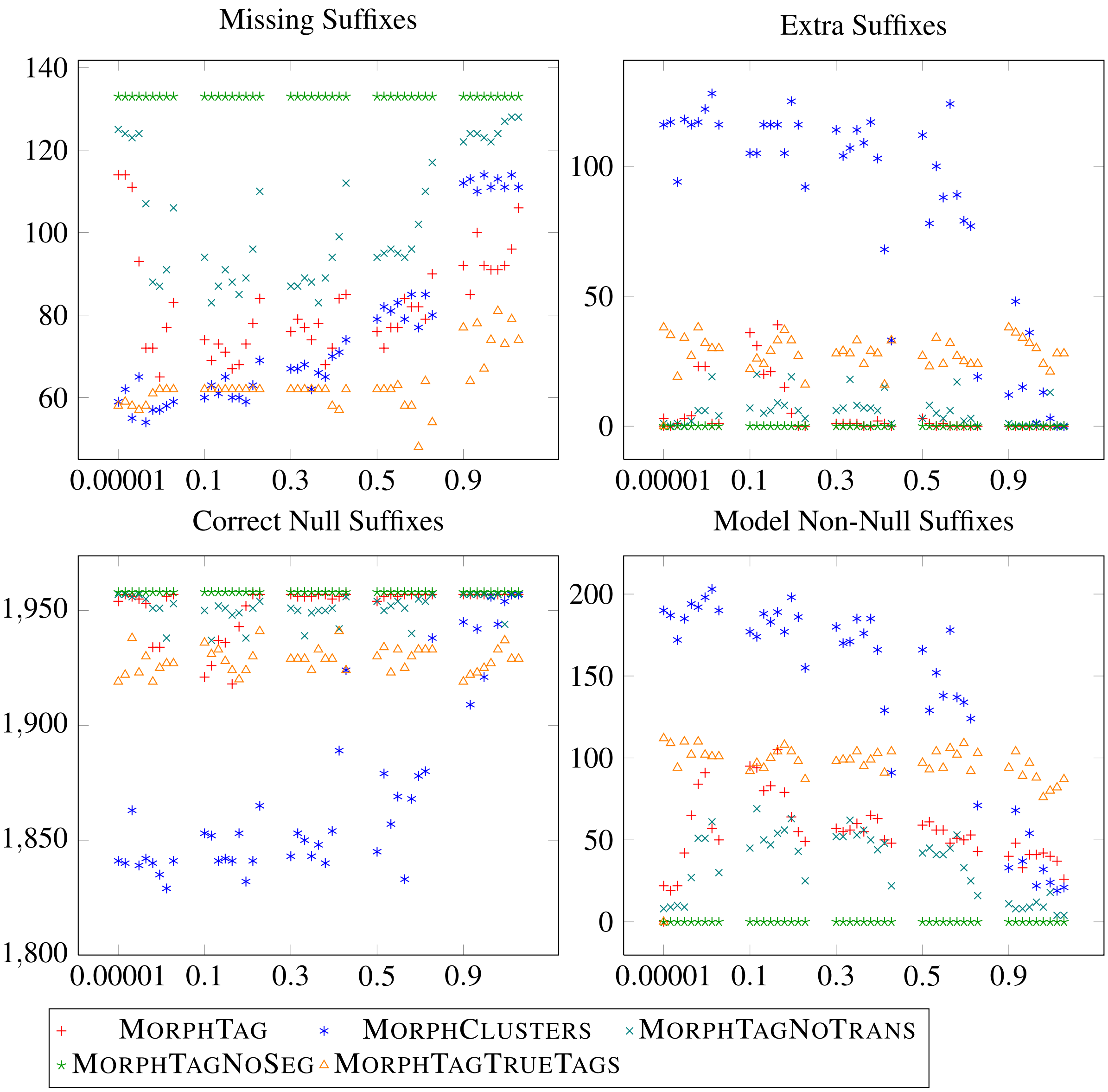

Figure 5.2: Eve Development Hyperparameter Settings: Suffix Counts

The x-axis shows all combinations of settings of the $a = 0.00001$, 0.1, 0.3, 0.5, 0.9 and $b = 0, 0.0001$, 0.01, 0.1,1, 10, 30, 100, 300 hyperparameters.

type, counts, from the development set only (remember the models are inferred using train+development data).

These plots allow us to identify some general trends. At small a, words are more likely to be reused from the cached tables, allowing the morphology generator G_t to put probability mass on suffixes that appear infrequently in terms of token counts but sufficiently often when using type counts. When a is larger, word tokens will be more likely to be directly generated from G_t. This leads to the null suffix having very high probability in the generator, since most word tokens do not share a suffix in English (95% of words in the development set have a gold null suffix).

This effect is most clearly visible for MORPHCLUSTERS. MORPHCLUSTERS proposes many (non-null) suffixes at small values of a, leading to oversegmentation; this declines at larger a, leading to greater numbers of missing suffixes. Different values of b seem no clear effect except when large b and large a lead to oversmoothing, which results in fewer non-null suffixes being proposed. (To a large extent, the oversegmentation occurring in MORPHCLUSTERS is due to the model splitting *is* to reuse a popular -*s* suffix — the sames suffix is used to generate *crayon.s, say.s, a.s*.)

Splitting the words into separate morphological processes dampens the oversegmentation problem: all other models propose far fewer non-null suffixes than MORPHCLUSTERS. The quality of the clusters has a clear effect on the number of morphemes proposed. Oracle clustering, in MORPHTAGTRUETAGS, leads to fairly consistent morphologies over all hyperparameter settings, apart from a slight decline at $a = 0.9$.

However, when the tags are unstable (as when they are being inferred), hyperparameters become more influential. The poor tag clustering found by MORPHTAGNOTRANS also leads to poor morphological segmentation, with severe undersegmentation, since within each cluster, there is not sufficient evidence for a shared sufix other than the null sufix. The fact that MORPHCLUSTERS does not suffer from this problem is due to the difference in inference: during table label resampling in the other models (see Section 5.5.4), an analysis can change clusters but not tags, since tags depend on token contexts. This means that MORPHCLUSTERS can more easily cluster together words with a shared suffix, by resampling tags as well as suffixes at this sampling step. (Its poor Tag VM scores suggests that it still clusters all other words fairly randomly.)

The full MORPHTAG models also suffer from poor tag clusters at very small a settings, leading to the same pattern of undersegmentation. However, at moderate a,

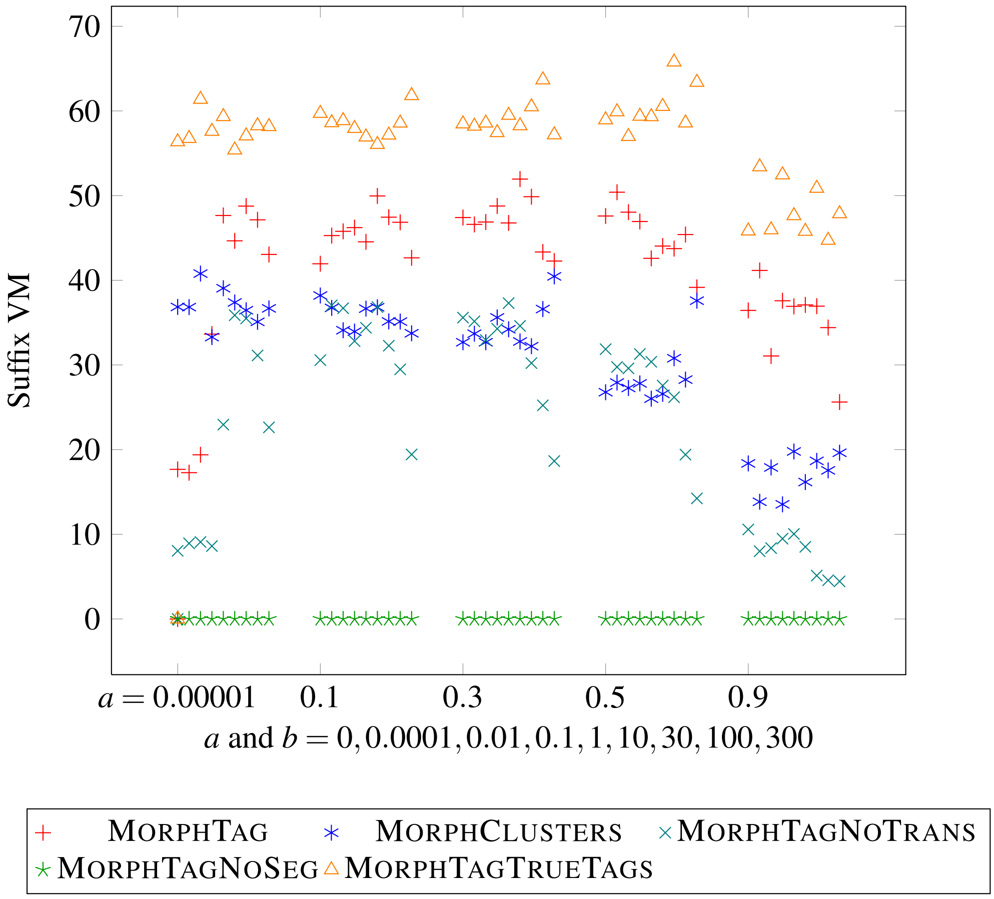

Figure 5.3: Eve Development Hyperparameter Settings: Suffix VM

MORPHTAG proposes more and better non-null suffixes than MORPHTAGNOTRANS, based on better word clusters found using local syntactic context information.

5.9.2.1 Suffix VM

We use Suffix VM to evaluate morphological segmentation performance, shown in Figure 5.3. Suffix VM varies more between hyperparameter settings than Tag VM, which is to be expected, given that the hyperparameters principally affect the morphological component of the model.

The MORPHTAGTRUETAGS models clearly outperform all other models, making use of correct tag information to find better morphological segmentations. The models with less informative word clusters, i.e. those without context information in the form of transition distributions, MORPHCLUSTERS and MORPHTAGNOTRANS, significantly underperform the other models. MORPHTAGNOSEG has a zero score due to the particularities of VM: by putting all words into the same morphological null-suffix cluster, it earns a homogenity (VM's precision analogue) score of zero, since

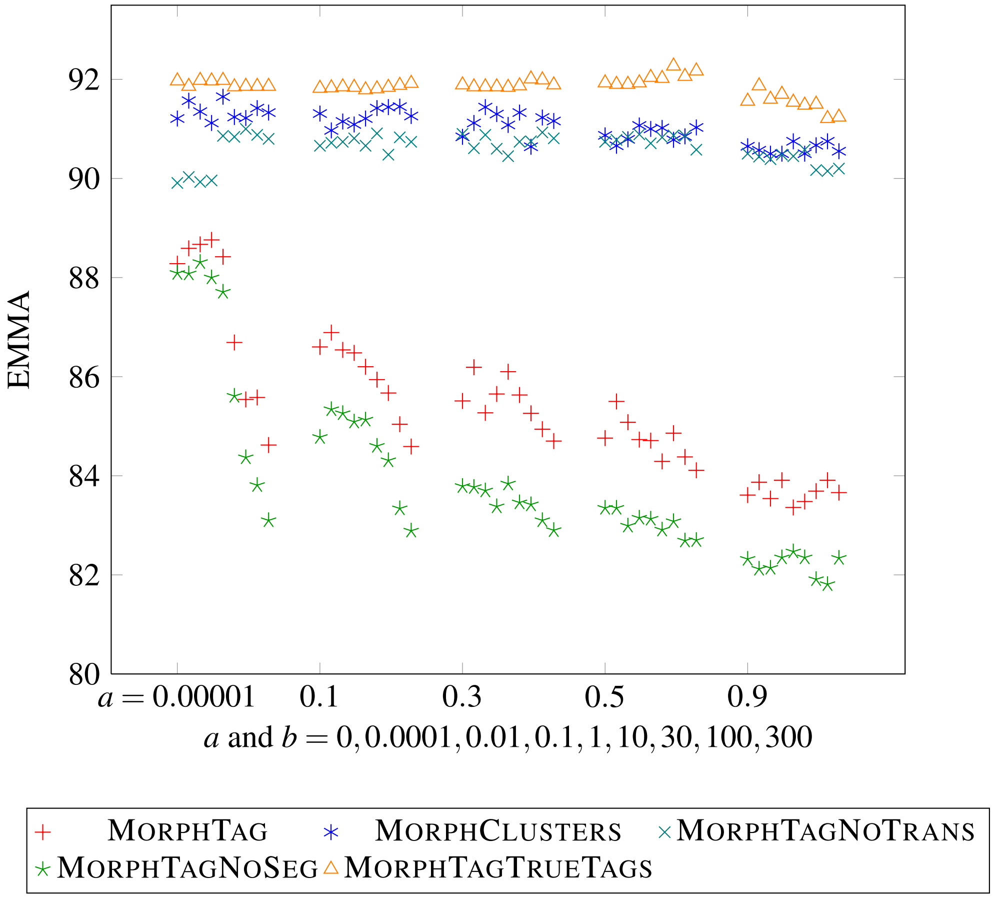

Figure 5.4: Eve Development Hyperparameter Settings: EMMA

the entropy over gold classes within that cluster matches the entropy over gold classes in general. The full MORPHTAG models clearly outperform the models without local syntactic constraints, by approximately ten VM points on average.

These results partially validate our hypothesis that modelling the interaction between syntactic clusters based on local contexts and morphological cues would benefit both tasks mutually. Given the tagging results, in English this interaction seems to mainly benefit the morphology segmentation task rather than tagging.

We note that the variation in suffix VM over multiple runs is relatively high (the standard deviation over ten runs is on the order of 5-10 points, whereas 0-2 points is more common for tag VM). Examining the log posteriors over iterations indicates that these models are each individually converging, suggesting that the variation is due to a search space with multiple local maxima, resulting in different suffixes being found.

5.9.2.2 EMMA

Figure 5.4 shows results for all models using the EMMA evaluation measure.

We first note the extremely small dynamic range: the MorphTagNoSeg baseline already performs very near ceiling (100) for a baseline without any morphology. This is due to the relative infrequency of words with non-null suffixes (roughly 15% of word types in the development set have a non-null suffix). EMMA performs a one-to-one matching of morpheme types in the gold and prediction lexicons when the matching is unambiguous, and adds partial weights in ambiguous cases. In the case of MorphTagNoSeg, the one-to-one matching will correctly match the unsegmented words with all gold unsegmented words, which then leads to high baseline performance. In the best case, when both *a* and *b* are small, this is at 88.

What causes the decrease in performance of MorphTagNoSeg at larger *a* and *b*? Recall that we annotate suffixes with tag or cluster membership. At small *a* and *b*, tagging performance for the models with transitions (MorphTagNoSeg and MorphTag) is poor. This is due to poor mixing — new tables are unlikely to be created, so all tokens get stuck at a same table, whether or not it is in an appropriate tag restaurant. This also leads to little tag ambiguity: most, if not all, word tokens of a word type are assigned to the same tag. When *a* and *b* become larger, tokens mix more, and often there are tokens of a word type within multiple tags; usually the majority are in one tag, with only a few outlier tokens. Since EMMA treats words in different tags as different types, and does not take frequencies into account, a single token in a second tag will mean that e.g. *tea* becomes ambiguous between *tea*-`TAG9` and *tea*-`TAG7`, and will cause a half point precision penalty score.

In aggregate, this tag ambiguity is the cause of the low scores for all models with transition distributions (MorphTag and MorphTagNoSeg). The transition distributions add pressure to put tokens in multiple tags, to accommodate the variation of contexts. Models without context (or with stable tags, in MorphTagTrueTags) do not suffer from this and consequently have extremely high performance (even when not segmenting particularly well according to Suffix VM).

If we do not add tag membership annotations, the ambiguity is removed and MorphTag scores are similar to MorphClusters. But MorphTagNoSeg performs just as well, simply by matching stems with very high precision. This leads us to discount EMMA as a metric able to capture the characteristics of model performance that we are interested in.

We note that the EMMA figures reported in Spiegler and Monson (2010) are significantly lower; these were results from a variety of morphological induction systems trained on data from the MorphoChallenge competition. The word lists used are

	Tag VM (s.d.)	Suffix VM (s.d.)	EMMA (s.d.)
MORPHTAG	59.44 (2.07)	50.39 (7.95)	85.50 (0.50)
MORPHCLUSTERS	23.93 (1.37)	29.24 (11.47)	91.78 (0.33)
MORPHTAGNOTRANS	23.27 (1.67)	37.26 (5.80)	91.77 (0.32)
MORPHTAGNOSEG	59.53 (1.59)	0.00 (0.00)	84.31 (0.41)
MORPHTAGTRUETAGS	100.00 (0.00)	46.50 (6.50)	90.75 (0.04)
BHMM	56.08 (2.20)	-	-

Table 5.1: Eve Test Results

	Tag VM (s.d.)	Suffix VM (s.d.)
MORPHTAG	59.06 (1.93)	41.93 (10.04)
MORPHCLUSTERS	22.44 (1.04)	27.97 (11.89)
MORPHTAGNOTRANS	21.24 (1.49)	27.23 (3.96)
MORPHTAGNOSEG	59.43 (1.68)	0.00 (0.00)
MORPHTAGTRUETAGS	100.00 (0.00)	42.52 (5.20)
BHMM	56.17 (2.30)	-

Table 5.2: Eve Train+Test Results

significantly more complex than CDS and contain far fewer null-suffixed words. In that setting, EMMA may be a viable metric (and is convincingly better than the MorphoChallenge metric), but it falls short when used on simpler data, like CDS tokens.

5.9.3 Test Results

To evaluate the test set, we chose the best hyperparameter settings from our development evaluation, as measured by Suffix VM. We use Suffix VM because it has the greatest dynamic range over the different hyperparameter settings. The best Suffix VM performance on the development set was obtained using a hyperparameter setting of $a = 0.3$ and $b = 10$.

Table 5.2 shows results on the combined train and test set using these settings. (See Table 5.1 for results over the test set alone.) We were not able to run the EMMA evaluation script over the full train and test set, due to out-of-memory errors.

In general, performance patterns follow those found on the development data. Tag

VM performance with inferred morphology in the full MorphTag models is not statistically significant from performance without morphology (MorphTagNoSeg); there does not seem to be much of a gain in English to adding morphology to a part of speech tagger, at least not with CDS data and this model structure. However, both MorphTag and MorphTagNoSeg significantly outperform the BHMM baseline.

On Suffix VM, we find that the full MorphTag models match the performance of MorphTagTrueTags. (The difference is statistically insignificant in both test and train+test evaluations.) As in the results from the development set, MorphTag outperforms the models without tags informed by local context by a large margin.

Test performance on EMMA is similar to dev performance, except in this case MorphClusters and MorphTagNoTrans manage to outperform even MorphTagTrueTags. MorphTag and MorphTagNoSeg are still heavily penalised for high levels of tag ambiguity.

5.10 Spanish Experiments

Experiments in Spanish are run on the Ornat corpus with the same train/dev/test split as in Chapter 3. We experiment with hyperparameters values as in the English experiments.

The Ornat corpus has fewer words than the Eve corpus (60084 vs. 89058 in train+dev), but more word types (3044 vs. 1964). In the train+dev data, there are 8683 possible stems and 6617 possible suffixes; for train+test those figures are 8649 and 6598. Note that these numbers are nearly twice as large as for English, especially in terms of word ending variability.

We experimented with the same values for the a hyperparameter as in the English experiments and used fewer values of b: 0, 0.0001, 0.01, 0.1, 1, 10, 100.

5.10.1 Tags

Figure 5.5 shows Tag VM results on the development set for a variety of hyperparameters. Unlike in English, there is a clear difference between tagging performance in the MorphTag and MorphTagNoSeg models: adding morphology improves tagging over the baseline without morphology. This difference is largest at moderate a values ($a = 0.1$ and $a = 0.5$); at larger a values, when the morphology model proposes more null suffixes, the difference disappears.

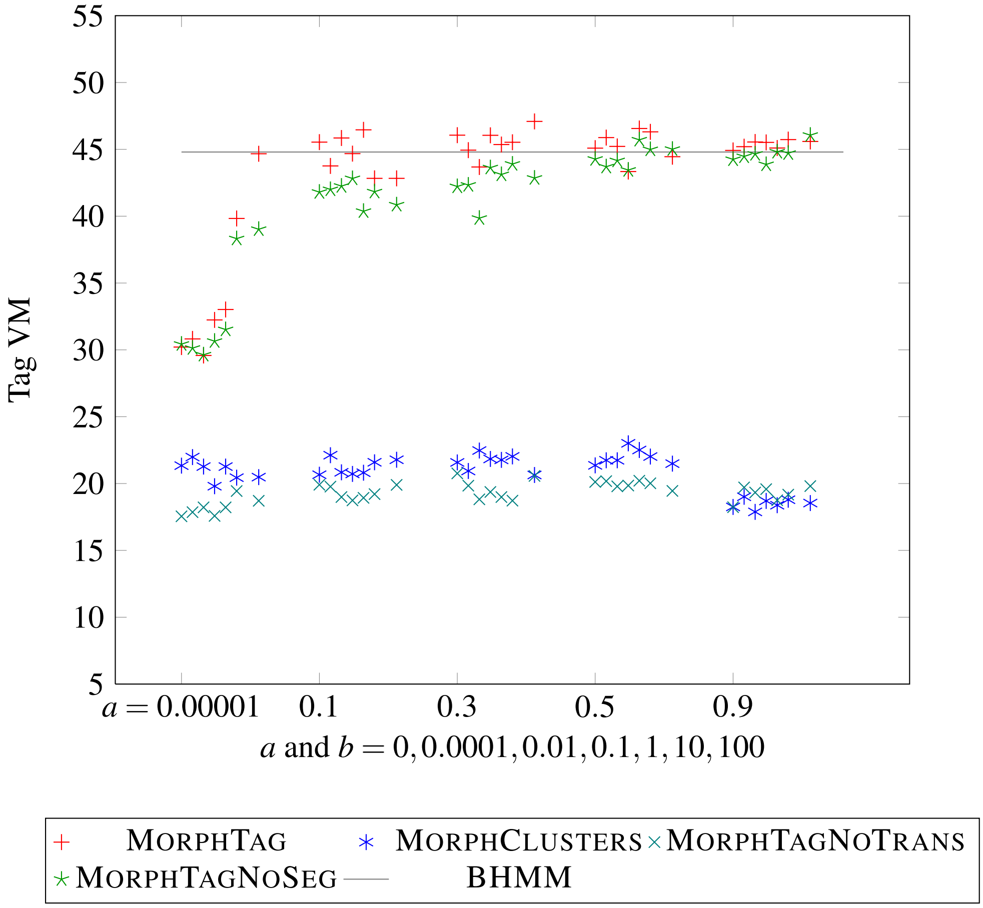

Figure 5.5: Ornat Development Hyperparameter Settings: Tag VM

As in English, including local contexts in the form of transition distributions is essential for accurate clusters; both MORPHCLUSTERS and MORPHTAGNOTRANS perform poorly. The more informative and frequent suffix patterns found in the Spanish data seem to give MORPHCLUSTERS an advantage over MORPHTAGNOTRANS that was not visible in English.

Interestingly, the BHMM baseline outperforms the MORPHTAGNOSEG model with PYP, except at large a where they are nearly equivalent models. At this setting, almost all words are generated from the base distribution's stem Dirichlet-multinomial, rather than from cached values. The difference between BHMM and MORPHTAGNOSEG performance in English and Spanish is probably due to fixed hyperparameter settings (i.e, $\alpha, \beta, \kappa, \sigma, \phi$); we did not have time to test these settings extensively.

5.10.2 Morphology

Plots showing counts of the suffixes found by the models are shown in Figure 5.6. There are far more words with gold non-null suffixes in this dataset, but nevertheless we see the same general patterns as in English: MORPHCLUSTERS proposes more non-null suffixes than MORPHTAG, except at $a = 0.9$. Fewer of these suffixes are incorrect: in Spanish, the patterns that MORPHCLUSTERS finds are more likely to correspond to true suffixes rather than coincidental patterns. Also in this vein, MORPHTAGTRUETAGS finds far more non-null suffixes than any other model, and most of these suffixes are correct. As in English, correct tag information leads to more reliable discovery of morphemes.

A note about the annotation of morphology in Spanish: The gold standard is constructed in such a way as to make many gold morpheme clusters impossible to find through segmentation. Verbs with little suffixation (third person singular present) are annotated as having a `-3S&PRES` suffix, even when the surface form is the same as the stem (e.g. *gusta* is annotated as *gusta*-`3S&PRES`). Unless the model makes the questionable, non-gold decision to represent the stem as *gust-*, this will lead to large numbers of missing suffixes, which is indeed what Fig. 5.6 shows.

Another potential problem is that noun affixes (such as plural) are labelled with gender (`-PL&MASC`, `-PL&FEM`). Gender is often signalled with *-a* or *-o* endings, but if the model finds only the *-s* suffix common to all nouns, these will be somewhat arbitrarily grouped between two gold categories.

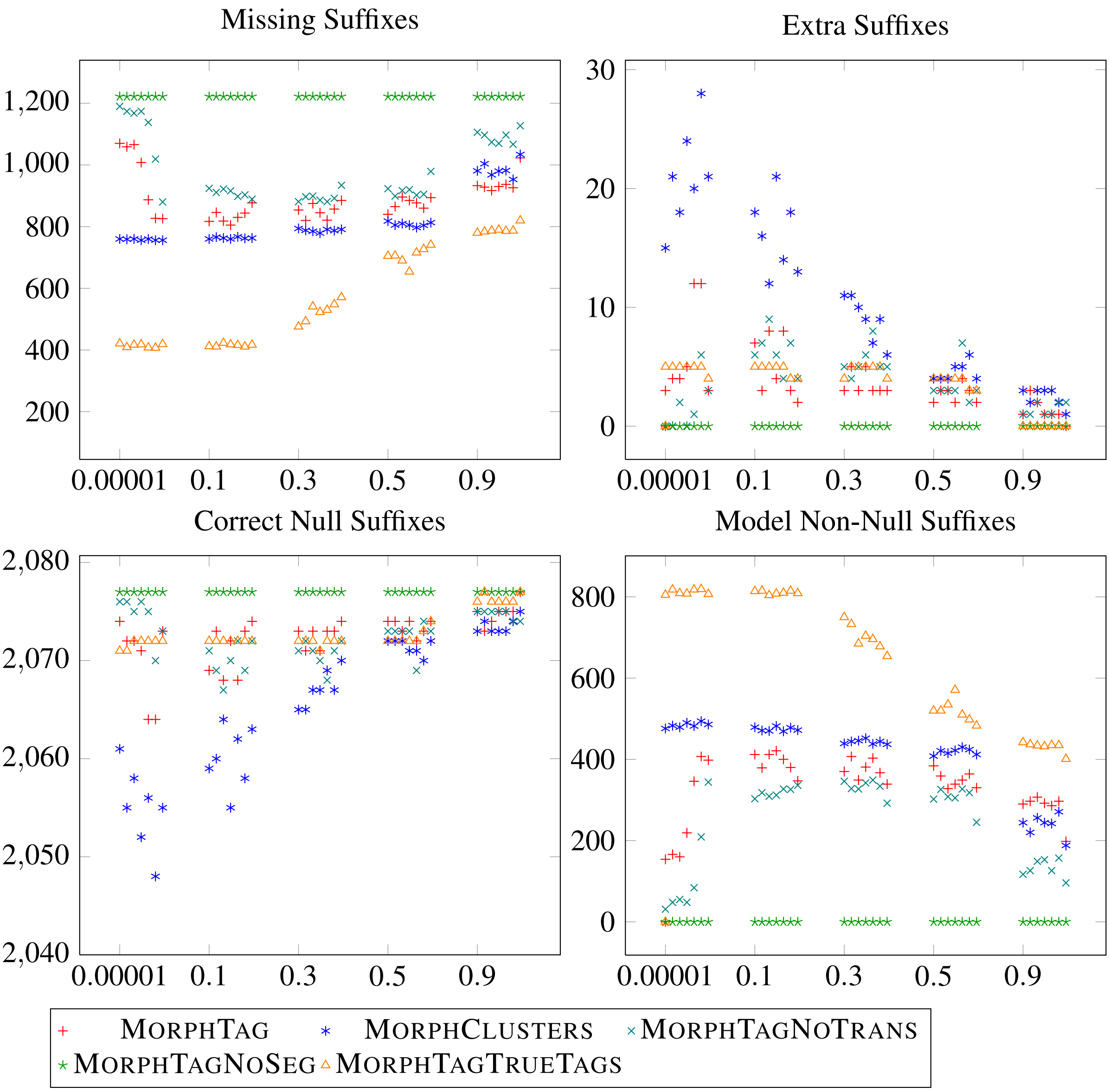

Figure 5.6: Ornat Development Hyperparameter Settings: Suffix Counts

The x-axis shows all combinations of settings of the $a = 0.00001$, 0.1, 0.3, 0.5, 0.9 and $b = 0, 0.0001$, 0.01, 0.1,1, 10, 100 hyperparameters.

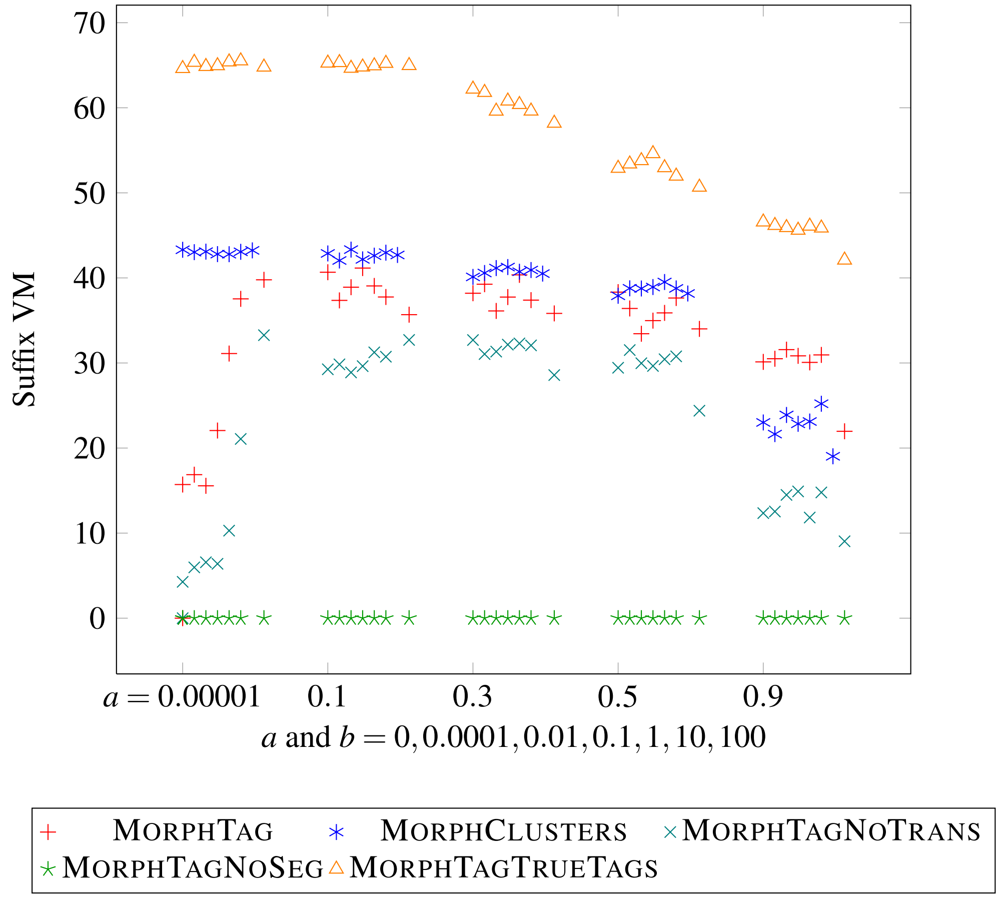

Figure 5.7: Ornat Development Hyperparameter Settings: Suffix VM

5.10.2.1 Suffix VM

Suffix VM performance is shown in Figure 5.7. Unlike English, there is no clear advantage for the MorphTag models over MorphClusters. Despite the fact that MorphTag finds better tag clusters, it slightly underperforms in comparison to MorphClusters. At small a, MorphTag suffers from poor tag clusters leading to undersegmentation; at $a = 0.9$, MorphClusters suffers from undersegmentation due to an overly high probability of null-suffixes.

MorphTagTrueTags performance drops as a becomes large, due to fewer suffixes being proposed, particularly in the verb category. As more tokens are generated from the base distribution directly, more weight is placed on the null suffix and infrequent suffixes are missed.

The models without transitions, MorphClusters and MorphTagNoTrans find more stable segmentations in Spanish than in English: Suffix VM performance varies minimally with b, and is surprisingly stable over a large range of a. A nearly

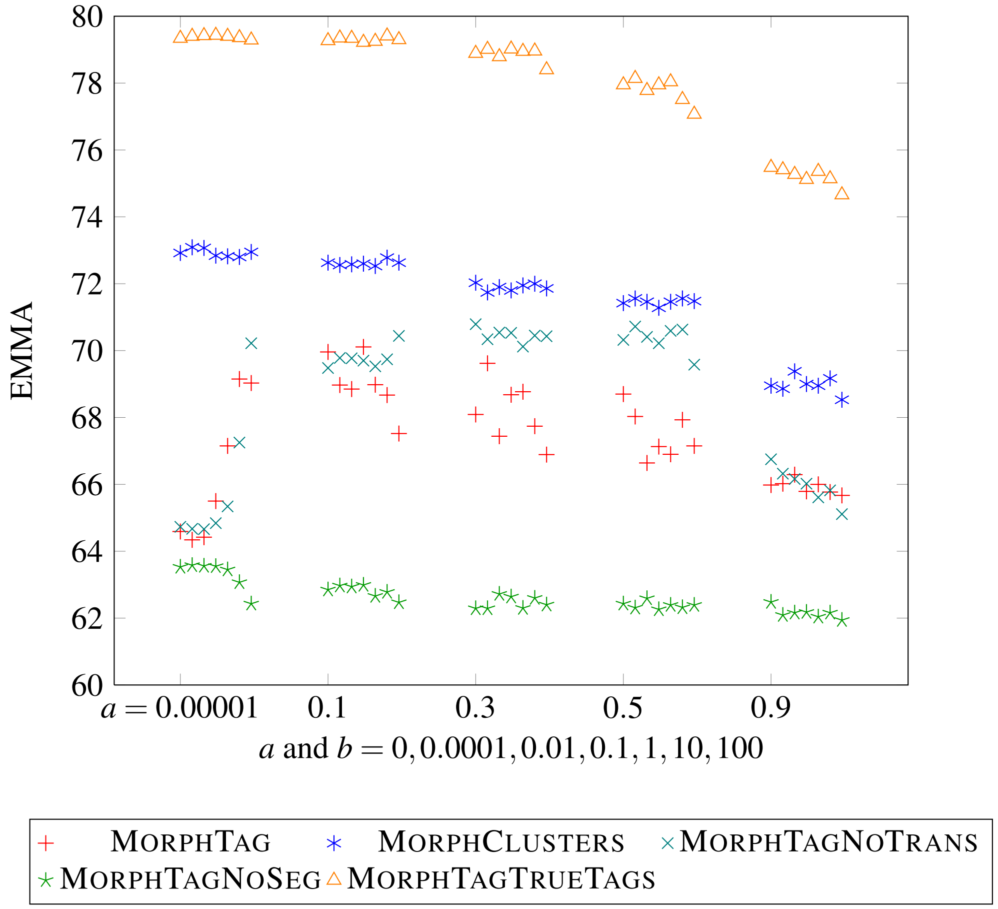

Figure 5.8: Ornat Development Hyperparameter Settings: EMMA

identical (incomplete) set of verb suffixes always found; the only other reliable suffix found is *-s*, which appears in many different clusters. The suffixes proposed by MorphTag are very similar. Evidently word-internal patterns are sufficient for identification of morphemes in Spanish.

5.10.2.2 EMMA

EMMA scores on Spanish, shown in Figure 5.8, are consistently significantly lower than in English, reflecting the higher amounts of suffixation, as well as the difficulty of capturing the annotation, discussed above. The MorphTagNoSeg baseline still scores fairly highly for a supposedly non-competitive baseline. As in English, we see the models with the least amount of tag ambiguity score highest, with MorphClusters still clearly outperforming the other models without gold tags.

	Tag VM (s.d.)	Suffix VM (s.d.)	EMMA (s.d.)
MorphTag	46.71 (2.66)	36.96 (2.93)	68.56 (1.79)
MorphClusters	22.77 (3.07)	41.16 (1.10)	72.93 (0.34)
MorphTagNoTrans	20.12 (1.77)	30.54 (2.31)	69.90 (0.87)
MorphTagNoSeg	42.27 (3.72)	0.00 (0.00)	61.84 (0.40)
MorphTrueTags	100.00 (0.00)	65.26 (0.25)	80.22 (0.37)
BHMM	43.10 (2.23)	-	-

Table 5.3: Ornat Test Results

	Tag VM (s.d.)	Suffix VM (s.d.)
MorphTag	43.35 (2.63)	34.43 (3.15)
MorphClusters	20.27 (2.51)	36.92 (1.26)
MorphTagNoTrans	16.80 (1.71)	25.52 (2.77)
MorphTagNoSeg	39.55 (3.71)	0.00 (0.00)
MorphTrueTags	100.00 (0.00)	61.39 (0.27)
BHMM	40.76 (1.95)	-

Table 5.4: Ornat Train+Test Results

5.10.3 Test Results

We found the best development Suffix VM scores with hyperparameter settings of $a = 0.1$ and $b = 0.1$; we use these settings to evaluate the models on the test data.

On the test evaluation, we find that the MorphTag model outperforms both the MorphTagNoSeg and the BHMM baseline models on the tagging task. These differences are significant both on the test set alone and on the train+test set (at $p < 0.05$).

In terms of Suffix VM, MorphTag and MorphClusters perform equally well (they are not statistically significantly different on the full test+train set). The difference in performance between MorphTagTrueTags and the models with inferred tags is much larger, and is in line with the much larger number of suffixes, particularly for verbs, proposed by this model: in a language with many morphemes, having stable and correct categories allows these morpheme patterns to become more visible.

However, the much higher quality tags in MorphTag are not sufficient to garner any improvement in Suffix VM over MorphClusters. The tagging improvement is mainly due to better clustering of function words such as determiners and conjunctions, which do not contain morphology. The set of suffixes found by MORPHTAG and MORPHCLUSTERS are quite similar and contain only the most frequent suffixes (e.g. *-s*, *-mos*, *-r*, *-n*).

The EMMA results are also similar to the development set results, with MORPHCLUSTERS outperforming all other models with inferred tags.

5.11 Conclusion

In this chapter, we have presented a model of joint syntactic category and morphology induction. This model is token-based, and thus allows for syntactic and morphemic ambiguity. We tested this model on two languages with different morphological characteristics. On English, a language with relatively little morphology, especially in CDS, we found that better categorisation of words yielded much better morphology in terms of suffixes learned. For the most part, this was due to the models with context information proposing fewer spurious suffixes, rather than more correct suffixes.

In Spanish, we saw less difference on the morphology task between models with categories inferred solely from morphemic patterns and models that also used local syntactic context for categorisation. The Spanish data included far more suffixes, and the difficulty for the models was primarily in proposing sufficient suffixes (i.e., not undersegmenting), rather than in proposing too few (oversegmenting).

However, in Spanish we saw an improvement in the tagging task when morphology information was included. This suggests that English (where there was no improvement when using morphology for tagging) and Spanish make different word-order and morphology trade-offs. In English, local context provides much of the same information (or more) as morphology in terms of determining the correct syntactic category, but knowing a good estimate of the correct syntactic category is useful for determining a word's morphology. In Spanish, a word's morphology can much more easily be determined simply by looking at frequent suffixes within a purely morphological system. On the other hand, word order is freer, so taking into account morphological information can add to local context in a productive way and improve tagging.

Chapter 6

Conclusion

In this thesis we have investigated the problem of unsupervised syntactic category acquisition from a number of angles. We now conclude the thesis by summarising our contributions to this problem, and indicate possible directions for future work.

6.1 Contributions

In Chapter 3, we extended a model of part of speech induction with a novel feature, sentence type. Previous work on part of speech induction has relied exclusively on very local contexts (a small number of surrounding words) to model distributional similarity. We argued that this ignores many long-range syntactic processes that have effects on local contexts, and proposed using sentence type as a feature signalling some of these effects. We examined the range and frequency of sentence types in child directed speech. Experimental results have demonstrated infants' awareness of sentence type signalled via prosody, further motivating the need to acknowledge the effects of sentence types on local contexts in models of acquisition.

The model structure we used, the BHMM, allowed us to experiment with a number of ways of adding sentence type to the model. We found that the model in which sentence type conditioned on transitions, and could thus reflect differences in local context, outperformed the BHMM without sentence type information.

The model proposed that included sentence types in transitions (BHMM-T) assumed all transitions would differ between sentence types. In Chapter 4 we relaxed this assumption by adding transition groups, allowing tags to share transition distributions between sentence types. When tags were fixed to their gold values, transition groups (in small corpora) were useful, and led to higher probability models than either fully shared or fully split variants. When inferring both tags and transition groups, this model recreated the fully split model when given sufficient amounts of data, reinforcing the usefulness of the sentence type cue for modelling transitions.

Having examined the effects of high-level syntactic features on syntactic categorisation, we then turned to word-internal morphological features in Chapter 5. Taking advantage of the morphological attributes of a word has long been known to help syntactic categorisation; likewise, morphological segmentation is facilitated by separating words into syntactic categories. Our model learns both tasks jointly, and is the first to do so without a type-constraint, enabling it to model natural language ambiguity in a natural way. We found that the joint model either met or exceeded the performance of single task models. Differences in patterns of performance between languages were in accordance with typological variation, with morphology being a more helpful cue for syntactic categorisation in languages with richer morphology and vice-versa. These results reinforce the importance of joint learning for language-independent models, as single task models will not be able to make use of the cues available from the other task.

The models in this thesis explored, in turn, the effects of increased complexity and realism in the transition and emission distributions of a BHMM-structured model for unsupervised part of speech tagging. Our overall results indicate that, while there are small but significant improvements to be had by a more informative transition distribution, such as transition distributions with sentence type information, larger improvements on tagging performance are obtainable with better emission distributions. In English, changing the emission distribution from a multinomial to a non-parametric distribution improved tagging performance and in Spanish we found even larger gains when the emission distribution included a model of morphology.

6.2 Future Work

6.2.1 Combined Models

A next step would be to add the two successful models together, by adding sentence type conditioned transitions from the BHMM-T to the joint morphology and tagging model. This would be straightforward to do, and potentially the increase in descriptiveness of the transitions might translate into better performance on the morphology task, via the improved categories.

6.2.2 Hyperparameter and Structure Estimation

The models in this thesis make a number of assumptions, such as hyperparameter settings, that could be learned by more complex hierarchical models incorporating 'over-hypotheses', i.e., hypotheses about the structure of likely hypotheses (Kemp et al., 2007).

In many cases, this will take the form of hyperparameter estimation. In the morphology-tagging model, we had a fixed hyperparameter (ϕ) governing the sparsity of suffixes in all tags. Learning the values of this hyperparameter would allow the model to accommodate different levels of morphological productivity between syntactic categories: function words, for example, typically have little morphology while content words have more.

Estimating a hyperparameter requires setting a prior over possible values of that hyperparameter. This prior could incorporate typological information about a particular language, such as its morphological richness, or be set on the basis of indications such as word ending variability. Alternatively, this prior could itself be learned, embodying the hypothesis that children learn higher order typological constraints, such as the relative informativeness of word order and morphology, as they learn the linguistic items themselves (cf. the Competition Model of Bates and MacWhinney (1987)). Linking the prior over suffixes, which governs morphological productivity, to the transition prior α in the BHMM would allow the model to incorporate intuitions about an anti-correlation between word order and morphology (Fedzechkina et al., 2011; Hengeveld et al., 2004; Moscoso del Prado, 2011).

Inferring the optimal values of the Pitman-Yor process parameters would be informative about the nature of the statistics involved in learning. The hyperparameters we found to be best confirmed the need for models to take both type and token frequencies into account.

Another kind of over-hypothesis involves learning structure (Kemp and Tenenbaum, 2008). For example, such a model of morphology could infer not only the morphemes of the language, but also discover the morphological structure of the language, i.e., whether it was prefixing or suffixing, agglutinative, with multiple suffixes per word, or isolating, with hardly any suffixes at all.

6.2.3 Experimental Implications

In Chapter 3 the BHMM-T incorporated the assumption that language learners can distinguish between sentence types and use this information to learn sentence-type-specific word orders. We found that this model outperformed a model without these assumptions, on some languages. Computational modelling thus demonstrated that an ideal learner would make use of the sentence type cue; however, behavioural experiments are required before making claims about whether human learners actually do so. In Section 3.7 we outlined an artificial language learning experiment that could test the assumptions made by the model on human learners.

A further strand of experimentation could investigate whether the model's pattern of results across languages (showing no improvement when incorporating sentence type on Spanish, for example) matched crosslinguistic behavioural results.

6.2.4 Process Models

As discussed in Chapters 1 and 2, the models in this thesis are on Marr's (1982) computational level and aim demonstrate what can be learned from the available input, using the representations and interactions posited by the model.

Modelling the learning process itself would require different inference methods, since the Gibbs sampling methods used here depend on multiple passes over the data. Depending on the complexity of the model, it may also require larger datasets, in line with the thousands of utterances an infant hears every day (Cameron-Faulkner et al., 2003).

A model of the process of acquisition would also strongly suggest a non-parametric model of categorisation, that is, a model in which the number of categories is not fixed in advance (unlike the BHMM, in which the number of states is a fixed parameter). This is not an absolute requirement for a process model, but a model that gradually converges to a set of categories, the size of which is not known in advance, is clearly more desirable from the standpoint of cognitive plausibility.

A process model would also open up the possibility of evaluating the model's learning process itself against the acquisition process as observed in children, rather than against adult gold standard categories.

Models such as Parisien et al. (2008) and Chrupala and Alishahi (2010) are examples of incremental models of unbounded category acquisition that could be used as starting points. The infinite HMM (Gael et al., 2009) would be the natural extension for the non-parametric versions of the BHMM variants presented here. However, the question of how to do inference over this model in an online fashion is an open question. Particle filters (Doucet and Johansen, 2011) are an applicable method, but ensuring their viability and tractability continues to be difficult (Börschinger and Johnson, 2012).

Bibliography

Abend, O., Reichart, R., and Rappoport, A. (2010). Improved unsupervised POS induction through prototype discovery. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Akhtar, N. and Tomasello, M. (1997). Young children's productivity with word order and verb morphology. *Developmental Psychology*, 33(6):952–965.

Alishahi, A. and Chrupala, G. (2012). Concurrent acquisition of word meaning and lexical categories.

Anderson, J. R. (1990). *The adaptive character of thought*. Lawrence Erlbaum Associates.

Andrieu, C., de Freitas, N., Doucet, A., and Jordan, M. I. (2003). An introduction to MCMC for machine learning. *Machine Learning*, 50:5–43.

Balog, H. L. and Brentari, D. (2008). The relationship between early gestures and intonation. *First Language*, 28(2):141–163.

Banko, M. and Moore, R. C. (2004). Part-of-speech tagging in context. In *Proceedings of the International Conference on Computational Linguistics (Coling)*.

Baroni, M., Matiasek, J., and Trost, H. (2002). Unsupervised discovery of morphologically related words based on orthographic and semantic similarity. In *Proceedings of the Workshop on Morphological and Phonological Learning of ACL/SIGPHON-2002*.

Bates, E. and MacWhinney, B. (1987). Competition, variation, and language learning. In MacWhinney, B., editor, *Mechanisms of language acquisition*, pages 157–194. Lawrence Erlbaum Associates.

Bates, E., MacWhinney, B., Caselli, C., Devescovi, A., Natale, F., and Venza, V. (1984). A cross-linguistic study of the development of sentence interpretation strategies. *Child Development*, 55(2):341–354.

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171.

Beal, M. J. (2003). *Variational Algorithms for Approximate Bayesian Inference*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London.

Berg-Kirkpatrick, T., Bouchard-Cote, A., DeNero, J., and Klein, D. (2010). Painless unsupervised learning with features. In *Proceedings of the North American Association for Computational Linguistics*.

Bernal, S., Dehaene-Lambertz, G., Millotte, S., and Christophe, A. (2010). Two-year-olds compute syntactic structure on-line. *Developmental Science*, 13(1):69–76.

Bernhard, D. (2006). Unsupervised morphological segmentation based on segment predictability and word segments alignment. In *Proceedings of the PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*.

Besag, J. (2004). Markov Chain Monte Carlo methods for statistical inference. Technical report, Department of Statistics University of Washington, USA.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer-Verlag.

Blunsom, P. and Cohn, T. (2011). A hierarchical Pitman-Yor process HMM for unsupervised part of speech induction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Booth, A. E. and Waxman, S. R. (2003). Mapping words to the world in infancy: Infants' expectations for count nouns and adjectives. *Journal of Cognition and Development*, 4(3):357–381.

Bordag, S. (2006). Two-step approach to unsupervised morpheme segmentation. In *In Proceedings of 2nd Pascal Challenges Workshop*, pages 25–29.

Börschinger, B. and Johnson, M. (2012). Using rejuvenation to improve particle filtering for Bayesian word segmentation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Brent, M. (1993). From grammar to lexicon: Unsupervised learning of lexical syntax. *Computational Linguistics*, 19(2):243–262.

Brent, M. R. and Cartwright, T. A. (1996). Lexical categorization: fitting template grammars by incremental MDL optimization. In *Proceedings of the 3rd International Colloquium on Grammatical Inference: Learning Syntax from Sentences*.

Brighton, H., Smith, K., and Kirby, S. (2005). Language as an evolutionary system. *Physics of Life Reviews*, 2(3):177–226.

Brown, P. F., Pietra, V. J. D., deSouza, P. V., Lai, J. C., and Mercer, R. L. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Brown, R. (1973). *A first language: The early stages*. Harvard University Press.

Bybee, J. (1995). Regular morphology and the lexicon. *Language and Cognitive Processes*, 10:425–455.

Cameron-Faulkner, T., Lieven, E., and Tomasello, M. (2003). A construction based analysis of child directed speech. *Cognitive Science*, 27(6):843–873.

Can, B. (2012). *Statistical Models for Unsupervised Learning of Morphology and POS tagging*. PhD thesis, The University of York.

Can, B. and Manandhar, S. (2010). Clustering morphological paradigms using syntactic categories. In *Multilingual Information Access Evaluation Vol. I, 10th Workshop of the Cross-Language Evaluation Forum, CLEF 2009*.

Cartwright, T. A. and Brent, M. R. (1997). Syntactic categorization in early language acquisition: formalizing the role of distributional analysis. *Cognition*, 63(2):121–170.

Celeux, G., Hurn, M., and Robert, C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association*, 95(451):957–970.

Chan, E. (2006). Learning probabilistic paradigms for morphology in a latent class model. In *Proceedings of the Eighth Meeting of the ACL Special Interest Group on Computational Phonology and Morphology*, SIGPHON '06, pages 69–78.

Chemla, E., Mintz, T. H., Bernal, S., and Christophe, A. (2009). Categorizing words using ”frequent frames”: What cross-linguistic analyses reveal about distributional acquisition strategies. *Developmental Science*, 12(3):396–406.

Chomsky, N. (1957). *Syntactic Structures*. Mouton.

Christodoulopoulos, C., Goldwater, S., and Steedman, M. (2010). Two decades of unsupervised POS induction: How far have we come? In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 575–584, Cambridge, MA. Association for Computational Linguistics.

Christodoulopoulos, C., Goldwater, S., and Steedman, M. (2011). A Bayesian mixture model for part-of-speech induction using multiple features. In *Proceedings of the 16th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Chrupala, G. and Alishahi, A. (2010). Online entropy-based model of lexical category acquisition. In *Proccedings of the 14th Conference on Natural Language Learning*.

Clark, A. (2000). Inducing syntactic categories by context distribution clustering. In *Proceedings of the 2nd workshop on Learning Language in Logic and the 4th conference on Computational Natural Language Learning*.

Clark, A. (2003a). Combining distributional and morphological information for part of speech induction. In *Proceedings of the 10th annual Meeting of the European Association for Computational Linguistics*.

Clark, E. V. (2003b). *First Language Acquisition*. Cambridge University Press.

Cohen, S. B. and Smith, N. A. (2007). Joint morphological and syntactic disambiguation. In *Proceedings of the 12th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Creutz, M. and Lagus, K. (2002). Unsupervised discovery of morphemes. In *Morphological and Phonological Learning: Proceedings of the 6th Workshop of the ACL Special Interest Group in Computational Phonology (SIGPHON)*.

Creutz, M. and Lagus, K. (2004). Induction of a simple morphology for highly-inflecting languages. In *Proceedings of the 7th Meeting of the ACL Special Interest Group in Computational Phonology: Workshop on Current Themes in Computational Phonology and Morphology*.

Creutz, M. and Lagus, K. (2007). Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing*, 4(1):1–34.

Croft, W. (2001). *Radical Construction Grammar*. Oxford University Press.

Dasgupta, S. and Ng, V. (2007a). High-performance, language-independent morphological segmentation. In *NAACL HLT 2007: Proceedings of the Main Conference*, pages 155–163.

Dasgupta, S. and Ng, V. (2007b). Unsupervised part-of-speech acquisition for resource-scarce languages. In *Proceedings of the 12th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

de Marcken, C. (1996). *Unsupervised Language Acquisition*. PhD thesis, Massachusetts Institute of Technology.

Déjean, H. (1998). Morphemes as necessary concept for structures discovery from untagged corpora. In *NeMLaP3/CoNLL98 Workshop on Paradigms and Grounding in Language Learning*.

Demberg, V. (2007). A language-independent unsupervised model for morphological segmentation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38.

Doucet, A. and Johansen, A. M. (2011). A tutorial on particle filtering and smoothing: fifteen years later. In Crisan, D. and Rozovsky, B., editors, *The Oxford Handbook of Nonlinear Filtering*. Oxford University Press.

Dreyer, M. and Eisner, J. (2011). Discovering morphological paradigms from plain text using a Dirichlet process mixture model. In *Proceedings of the 16th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Fedzechkina, M., Jaeger, F. T., and Newport, E. L. (2011). Functional biases in language learning: Evidence from word order and case-marking interaction. In *Proceedings of the 33rd Annual Conference of the Cognitive Science Society (CogSci)*.

Fernald, A. (1989). Intonation and communicative intent in mothers' speech to infants: Is the melody the message? *Child Development*, 60(6):1497–1510.

Fernald, A. and Kuhl, P. (1987). Acoustic determinants of infant preference for motherese speech. *Infant Behavior and Development*, 10(3):279–293.

Fernald, A. and Mazzie, C. (1991). Prosody and focus in speech to infants and adults. *Developmental Psychology*, 27(2):209–221.

Forney, G. (1973). The Viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278.

Frank, M. C., Goodman, N. D., Tenenbaum, J. B., and Fernald, A. (2009). Continuity of discourse provides information for word learning. In *Proceedings of the 31st Annual Conference of the Cognitive Science Society (CogSci)*.

Gael, J. V., Saatci, Y., Teh, Y. W., and Ghahramani, Z. (2008). Beam sampling for the infinite hidden Markov model. In *International Conference of Machine Learning*.

Gael, J. V., Vlachos, A., and Ghahramani, Z. (2009). The infinite HMM for unsupervised POS tagging. In *Proceedings of the 14th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Galligan, R. (1987). Intonation with single words: purposive and grammatical use. *Journal of Child Language*, 14:pp 1–21.

Gao, J. and Johnson, M. (2008). A comparison of Bayesian estimators for unsupervised hidden Markov model POS taggers. In *Proceedings of the 13th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741.

Gerken, L., Wilson, R., and Lewis, W. (2005). Infants can use distributional cues to form syntactic categories. *Journal of Child Language*, 32:249–268.

Gervain, J., Nespor, M., Mazuka, R., Horie, R., and Mehler, J. (2008). Bootstrapping word order in prelexical infants: A Japanese-Italian cross-linguistic study. *Cognitive Psychology*, 57:56–74.

Ghahramani, Z. (2001). An introduction to hidden Markov models and Bayesian networks. *International Journal of Pattern Recognition and Artificial Intelligence*, 15(1):9–42.

Gleitman, L. R., Newport, E. L., and Gleitman, H. (1984). The current status of the motherese hypothesis. *Journal of Child Language*, 11:43–79.

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). Switchboard: Telephone speech corpus for research and development. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Goldberg, Y. and Tsarfaty, R. (2008). A single generative model for joint morphological segmentation and syntactic parsing. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Goldsmith, J. (2001). Unsupervised learning of the morphology of a natural language. *Computational Linguistics*, 27:153–198.

Goldsmith, J. (2006). An algorithm for the unsupervised learning of morphology. *Natural Language Engineering*, 12(4):353–371.

Goldwater, S. (2007). *Nonparametric Bayesian Modes of Lexical Acquisition*. PhD thesis, Brown University.

Goldwater, S. and Griffiths, T. L. (2007). A fully Bayesian approach to unsupervised part-of-speech tagging. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Goldwater, S., Griffiths, T. L., and Johnson, M. (2006). Interpolating between types and tokens by estimating power-law generators. In *Advances in Neural Information Processing Systems 18*.

Gómez, R. L. and Gerken, L. (1999). Artificial grammar learning by 1-year-olds leads to specific and abstract knowledge. *Cognition*, 70:109135.

Gómez, R. L. and Lakusta, L. (2004). A first step in form-based category abstraction by 12-month-old infants. *Developmental Science*, 7(5):567–580.

Graca, J., Ganchev, K., Coheur, L., Pereira, F., and Taskar, B. (2011). Controlling complexity in part-of-speech induction. *Journal of Artificial Intelligence Research*, 41:527–551.

Griffiths, T. L., Canini, K. R., Sanborn, A. N., and Navarro, D. J. (2007). Unifying rational models of categorization via the hierarchical Dirichlet process. In *Proceedings of the 29th Annual Conference of the Cognitive Science Society (CogSci)*.

Haghighi, A. and Klein, D. (2006). Prototype-driven grammar induction. In *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Hakuta, K. (1982). Interaction between particles and word order in the comprehension and production of simple sentences in Japanese children. *Developmental Psychology*, 18:62–76.

Hammarström, H. and Borin, L. (2011). Unsupervised learning of morphology. *Computational Linguistics*, 37(2):309–350.

Harris, Z. (1955). From phoneme to morpheme. *Language*, 31(2):190–222.

Harris, Z. (1967). Morpheme boundaries within words: report on a computer test. In *Transformations and Discourse Analysis Papers No. 73*. Philadelphia: University of Pennsylvania.

Hasan, K. S. and Ng, V. (2009). Weakly supervised part-of-speech tagging for morphologically-rich, resource-scarce languages. In *Proceedings of the Twelfth Conference of the European Chapter of the Association for Computational Linguistics*.

Hengeveld, K., Rijkhoff, J., and Siewierska, A. (2004). Parts of speech systems and word order. *Journal of Linguistics and Philosopy*, 40:527–570.

Hirst, D. and Cristo, A. D., editors (1998). *Intonation systems: a survey of twenty languages*. Cambridge University Press.

Homae, F., Watanabe, H., Nakano, T., Asakawa, K., and Taga, G. (2006). The right hemisphere of sleeping infant perceives sentential prosody. *Neuroscience Research*, 54(4):276 – 280.

Hu, Y., Matveeva, I., Goldsmith, J., and Sprague, C. (2005). Using morphology and syntax together in unsupervised learning. In *Papers from the Workshop on Psychocomputational Models of Human Language Acquisition*.

Huddleston, R. D. and Pullum, G. K. (2002). *The Cambridge Grammar of the English Language*. Cambridge University Press.

Ishwaran, H. and James, L. F. (2003). Generalized weighted Chinese restaurant processes for species sampling mixture models. *Statistica Sinica*, 13:1211–1235.

Jasra, A., Holmes, C. C., and Stephens, D. A. (2005). Markov Chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statistical Science*, 20(1):50–67.

Johnson, M. (2007). Why doesnt EM find good HMM POS-taggers? In *Proceedings of the 12th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Johnson, M. (2008). Using Adaptor Grammars to identify synergies in the unsupervised acquisition of linguistic structure. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Jordan, M. I. (2005). Dirichlet processes, Chinese restaurant processes and all that. Technical report, Tutorial presentation at the NIPS Conference.

Kelly, M. H. (1992). Using sound to solve syntactic problems: the role of phonology in grammatical category assignments. *Psychological Review*, 99(2):349–364.

Kemp, C., Perfors, A., and Tenenbaum, J. B. (2007). Learning overhypotheses with hierarchical Bayesian models. *Developmental Science*, 10(3):307–321.

Kemp, C. and Tenenbaum, J. B. (2008). The discovery of structural form. *Proceedings of the National Academy of Sciences*, 105(31).

Kemp, C., Tenenbaum, J. B., Niyogi, S., and Griffiths, T. L. (2010). A probabilistic model of theory formation. *Cognition*, page 165196.

Keshava, S. and Pitler, E. (2006). A simpler, intuitive approach to morpheme induction. In *PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*.

Kirby, S., Dowman, M., and Griffiths, T. (2007). Innateness and culture in the evolution of language. *Proceedings of the National Academy of Sciences*, 104(12):5241–5245.

Klein, D. and Manning, C. (2004). Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of the 9th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Kurimo, M., Virpioja, S., and Turunen, V. T. (2010). Proceedings of the MorphoChallenge 2010 workshop. Technical Report TKK-ICS-R37, Aalto University School of Science and Technology, Espoo, Finland.

Kwiatkowski, T., Goldwater, S., Zettelmoyer, L., and Steedman, M. (2012). A probabilistic model of syntactic and semantic acquisition from child-directed utterances and their meanings. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*.

Langacker, R. (1987). *Foundations of Cognitive Grammar, Volume I, Theoretical Prerequisites*. Stanford University Press.

Lee, J., Naradowsky, J., and Smith, D. (2011a). A discriminative model for joint morphological disambiguation and dependency parsing. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Lee, T. H., Wong, C. H., Leung, S., Man, P., Cheung, A., Szeto, K., and Wong, C. S. P. (1994). The development of grammatical competence in Cantonese-speaking children. Technical report, The Chinese University of Hong Kong.

Lee, Y. K., Haghighi, A., and Barzilay, R. (2010). Simple type-level unsupervised POS tagging. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Lee, Y. K., Haghighi, A., and Barzilay, R. (2011b). Modeling syntactic context improves morphological segmentation. In *Proceedings of Fifteenth Conference on Computational Natural Language Learning*.

Liang, P., Jordan, M., and Klein, D. (2010). Type-based MCMC. In *Proceedings of the 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

MacKay, D. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

MacKay, D. J. C. (1997). Ensemble learning for hidden Markov models. "`http://www.inference.phy.cam.ac.uk/mackay/abstracts/ensemblePaper.html`".

Macnamara, J. (1978). How do babies learn grammatical categories? In Sankoff, D., editor, *Linguistic variation: Models and methods*. New York: Academic Press.

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

Mampe, B., Friederici, A. D., Christophe, A., and Wermke, K. (2009). Newborns' cry melody is shaped by their native language. *Current Biology*, 19(23):1994 – 1997.

Mandel, D. R., Jusczyka, P. W., and Kemler Nelson, D. G. (1994). Does sentential prosody help infants organize and remember speech information? *Cognition*, 53:155–180.

Mandel, D. R., Kemler Nelson, D. G., and Jusczyk, P. W. (1996). Infants remember the order of words in a spoken sentence. *Congnitive Development*, 11:181–196.

Manning, C. D. and Schütze, H. (1999). *Foundations of Natural Language Processing*. The MIT Press, Cambridge, USA.

Maratsos, M. P. and Chalkley, M. A. (1980). The internal language of children's syntax: The ontogenesis and representation of syntactic categories. *Children's language*, 2:127–214.

Marcus, M. P., Santorini, B., Marcinkiewicz, M. A., and Taylor, A. (1999). Treebank-3. Linguistic Data Consortium.

Marr, D. (1982). *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W. H. Freeman and Co.

Matthews, D., Lieven, E., Theakston, A., and Tomasello, M. (2007). French children's use and correction of weird word orders: A constructivist account. *Journal of Child Language*, 34:381–409.

Mehler, J., Jusczyk, P., Lambertz, G., Halsted, N., Bertoncini, J., and Amiel-Tison, C. (1988). A precursor of language acquisition in young infants. *Cognition*, 29(2):143–178.

Meila, M. (2007). Comparing clusterings — an information-based distance. *Journal of Multivariate Analysis*, 98:873–895.

Merialdo, B. (1994). Tagging English text with a probabilistic model. *Computational Linguistics*, 20(2):155–172.

Mervis, C. B. and Johnson, K. E. (1991). Acquisition of the plural morpheme: A case study. *Developmental Psychology*, 27(2):222–235.

Minka, T. (2001). *A family of algorithms for approximate Bayesian inference*. PhD thesis, Massachusetts Institute of Technology.

Mintz, T. H. (2003). Frequent frames as a cue for grammatical categories in child directed speech. *Cognition*, 90:91–117.

Mintz, T. H. (2006). Finding the verbs: distributional cues to categories available to young learners. In Hirsh-Pasek, K. and Golinkoff, R. M., editors, *Action Meets Word: How Children Learn Verbs*, pages 31–63. Oxford University Press.

Mintz, T. H., Newport, E. L., and Bever, T. G. (2002). The distributional structure of grammatical categories in speech to young children. *Cognitive Science*, 26:393–424.

Monaghan, P., Chater, N., and Christiansen, M. H. (2005). The differential contribution of phonological and distributional cues in grammatical categorisation. *Cognition*, 96(2):143–182.

Monaghan, P. and Christiansen, M. H. (2004). What distributional information is useful and usable in language acquisition? In *Proceedings of the 27th Annual Conference of the Cognitive Science Society (CogSci)*.

Monaghan, P., Christiansen, M. H., and Chater, N. (2007). The phonological distributional coherence hypothesis: Cross-linguistic evidence in language acquisition. *Cognitive Psychology*, 55:259–305.

Monson, C., Carbonell, J., Lavie, A., and Levin, L. (2007). Paramor: Minimally supervised induction of paradigm structure and morphological analysis. In *Proceedings of Ninth Meeting of the ACL Special Interest Group in Computational Morphology and Phonology*.

Monson, C., Lavie, A., Carbonell, J., and Levin, L. (2008). Evaluating an agglutinative segmentation model for paramor. Technical Report 285, Carnegie Mellon University.

Moon, T., Erk, K., and Baldridge, J. (2009). Unsupervised morphological segmentation and clustering with document boundaries. In *Proceedings of the 14th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Moscoso del Prado, F. (2011). The mirage of morphological complexity. In *Proceedings of the 33rd Annual Conference of the Cognitive Science Society (CogSci)*.

Murray, L. and Trevarthen, C. (1986). The infant's role in mother-infant communications. *Journal of Child Language*, 13:15–29.

Naradowsky, J. and Goldwater, S. (2009). Improving morphology induction by learning spelling rules. In *Proceedings of the 21st International Joint Conference on Artifical intelligence*.

Naradowsky, J. and Toutanova, K. (2011). Unsupervised bilingual morpheme segmentation and alignment with context-rich hidden semi-Markov models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Neal, R. (1993). Probabilistic inference using Markov Chain Monte Carlo methods. Technical Report Technical Report CRG-TR-93-1, Dept. of Computer Science, University of Toronto.

Neuvel, S. and Fulop, S. (2002). Unsupervised learning of morphology without morphemes. In *Proceedings of the Workshop on Morphological and Phonological Learning*.

Newport, E. L., Gleitman, H., and Gleitman, L. R. (1977). Mother, I'd rather do it myself: Some effects and non-effects of maternal speech style. In Snow, C. E. and Ferguson, C. A., editors, *Talking to Children: Language input and acquisition*, pages 109–149. Cambridge University Press, Cambridge, UK.

Och, F. (1999). An efficient method for determining bilingual word classes. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Olguin, R. and Tomasello, M. (1993). Twenty-five-month-old children do not have a grammatical category of verb. *Cognitive Development*, 8:245–272.

Onnis, L. and Christiansen, M. H. (2005). New beginnings and happy endings: Psychological plausibility in computational models of language acquisition. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society (CogSci)*.

Onnis, L., Waterfall, H. R., and Edelmanc, S. (2008). Learn locally, act globally: Learning language from variation set cues. *Cognition*, 109(3):423–430.

Orbanz, P. and Teh, Y. W. (2011). Modern Bayesian nonparametrics. Technical report, Tutorial presentation at the NIPS Conference.

Ornat, S. L. (1994). *La adquisicion de la lengua espagnola*. Siglo XXI, Madrid.

Parisien, C., Fazly, A., and Stevenson, S. (2008). An incremental Bayesian model for learning syntactic categories. In *Proceedings of the 12th Conference on Computational Natural Language Learning*.

Perfors, A. (2008). *Learnability, representation, and language: A Bayesian approach*. PhD thesis, Massachusetts Institute of Technology.

Perfors, A., Tenenbaum, J. B., and Regier, T. (2011). The learnability of abstract syntactic principles. *Cognition*, 118(3):306 – 338.

Pitman, J. and Yor, M. (1997). The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25(2):855–900.

Redington, M., Chater, N., and Finch, S. (1993). Distributional information and the acquisition of linguistic categories: A statistical approach. In *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society*.

Redington, M., Chater, N., and Finch, S. (1998). Distributional information: a powerful cue for acquiring syntactic categories. *Cognitive Science*, 22:425 –469.

Rosenberg, A. and Hirschberg, J. (2007). V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 12th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Saffran, J. R., Aslin, R. N., and Newport, E. L. (1996). Statistical learning by 8-month-old infants. *Science*, 274(5294):1926–1928.

Saul, L. and Pereira, F. (1997). Aggregate and mixed-order Markov models for statistical language processing. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 81–89.

Schone, P. and Jurafsky, D. (2000). Knowledge-free induction of morphology using latent semantic analysis. In *Proceedings of the Conference on Natural Language Learning*.

Schone, P. and Jurafsky, D. (2001). Knowledge-free induction of inflectional morphologies. In *Proceedings of the North American Association for Computational Linguistics*.

Schütze, H. (1995). Distributional part-of-speech tagging. In *Proceedings of the 5th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Schütze, H. and Walsh, M. (2008). A graph-theoretic model of lexical syntactic acquisition. In *Proceedings of the 13th Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Scott, R. M. and Fisher, C. (2009). Two-year-olds use distributional cues to interpret transitivity-alternating verbs. *Language and Cognitive Processes*, 24(6):777–803.

Sirts, K. and Alumäe, T. (2012). A hierarchical Dirichlet process model for joint part-of-speech and morphology induction. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*.

Slobin, D. (1982). Universal and particular in the acquisition of language. In Wanner, E. and Gleitman, L. R., editors, *Language acquisition: the state of the art*, pages 128–170. Cambridge University Press.

Smith, N. A. and Eisner, J. (2005). Contrastive estimation: Training log-linear models on unlabeled data. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 354–362, Ann Arbor, MI.

Snow, D. and Balog, H. (2002). Do children produce the melody before the words? A review of developmental intonation research. *Lingua*, 112:1025–1058.

Snyder, B. and Barzilay, R. (2008). Unsupervised multilingual learning for morphological segmentation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Spiegler, S. and Monson, C. (2010). EMMA: A novel evaluation metric for morphological analysis. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling)*.

Srinivasan, R. J. and Massaro, D. W. (2003). Perceiving prosody from the face and voice: Distinguishing statements from echoic questions in English. *Language and Speech*, 46:1–22.

St Clair, M., Monaghan, P., and Christiansen, M. (2010). Learning grammatical categories from distributional cues: Flexible frames for language acquisition. *Cognition*, 116(3):341–360.

Stephens, M. (2000). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(4):795–809.

Stern, D. N., Spieker, S., Barnett, R. K., and MacKain, K. (1983). The prosody of maternal speech: infant age and context related changes. *Journal of Child Language*, 10:1–15.

Stern, D. N., Spieker, S., and MacKain, K. (1982). Intonation contours as signals in maternal speech to prelinguistic infants. *Developmental Psychology*, 18(5):727–735.

Taylor, P. A., King, S., Isard, S. D., and Wright, H. (1998). Intonation and dialogue context as constraints for speech recognition. *Language and Speech*, 41(3):493–512.

Teh, Y. W. (2006). A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 985 – 992, Sydney.

Teichert, A. R. and Daumé III, H. (2009). Unsupervised part of speech tagging without a lexicon. In *NIPS Workshop on Grammar Induction, Representation of Language and Language Learning (GIRLLL)*.

Tenenbaum, J. B., Griffiths, T. L., and Kemp, C. (2006). Theory-based Bayesian models of inductive learning and reasoning. *Trends in Cognitive Science*, 10:309–318.

Theakston, A., Lieven, E., Pine, J. M., and Rowland, C. F. (2001). The role of performance limitations in the acquisition of verb-argument structure: an alternative account. *Journal of Child Language*, 28:127–152.

Thothathiri, M., Snedeker, J., and Hannon, E. (2011). The effect of prosody on distributional learning in 12- to 13-month-old infants. *Infant and Child Development*.

Tomasello, M. and Farrar, M. J. (1986). Joint attention and early language. *Child Development*, 57(6):1454–1463.

Tomasello, M. and Olguin, R. (1993). Twenty-three-month-old children have a grammatical category of noun. *Cognitive Development*, 8(4):451–64.

Toutanova, K. and Johnson, M. (2007). A Bayesian LDA-based model for semi-supervised part-of-speech tagging. In *Advances in Neural Information Processing Systems 20*.

Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of HLT-NAACL 2003*.

Valian, V. (1986). Syntactic categories in the speech of young children. *Developmental Psychology*, 22:562–579.

Viterbi, A. J. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269.

Vlachos, A., Korhonen, A., and Ghahramani, Z. (2009). Unsupervised and constrained Dirichlet process mixture models for verb clustering. In *4th Workshop on Statistical Machine Translation, EACL' 09*.

Wallace, C. S. and Freeman, P. R. (1987). Estimation and inference by compact coding. *Journal of the Royal Statistical Society. Series B (Methodological)*, 49(3):240–265.

Welch, L. R. (2003). The Shannon lecture: Hidden Markov models and the Baum-Welch algorithm. *IEEE Information Theory Society Newsletter*.

Xanthos, A., Laaha, S., Gillis, S., Stephany, U., Aksu-Koç, A., Christofidou, A., Gagarina, N., Hrzica, G., Ketrez, F. N., Kilani-Schoch, M., Korecky-Kröll, K., Kovačević, M., Laalo, K., Palmović, M., Pfeiler, B., Voeikova, M. D., and Dressler, W. U. (2011). On the role of morphological richness in the early development of noun and verb inflection. *First Language*, 31(4):461–479.

Yarowsky, D. and Wicentowski, R. (2000). Minimally supervised morphological analysis by multimodal alignment.

Zhou, P., Crain, S., and Zhan, L. (2012). Sometimes children are as good as adults: The pragmatic use of prosody in children's on-line sentence processing. *Journal of Memory and Language*, 67:149–164.