

UNSUPERVISED JOINT PART-OF-SPEECH TAGGING AND STEMMING FOR AGGLUTINATIVE LANGUAGES

SONDAN EKLEMELİ DİLLERDE GÖZETİMSİZ EŞZAMANLI SÖZCÜK TÜRÜ İŞARETLEME VE GÖVDELEME

Necva BÖLÜCÜ

Asst. Prof. Dr. Burcu CAN BUĞLALILAR

Supervisor

Submitted to Graduate School of Science and Engineering of
Hacettepe University
as a Partial Fulfillment to the Requirements
for the Award of the Degree of Master of Science
in Computer Engineering

2017

This work named "**Unsupervised Joint Part-of-Speech Tagging and Stemming for Agglutinative Languages**" by Neeva BÖLÜCÜ has been approved as a thesis for the Degree of **Master of Science IN COMPUTER ENGINEERING** by the below mentioned Examining Committee Members.

Prof. Dr. Deniz ZEYREK

Head

Doç. Dr. Harun ARTUNER

Member

Asst. Prof. Dr. Burcu CAN BUĞLALILAR

Supervisor

Asst. Prof. Dr. Gönenç ERCAN

Member

Asst. Prof. Dr. Özkan KILIÇ

Member

This thesis has been approved as a thesis for the Degree of **MASTER OF SCIENCE IN COMPUTER ENGINEERING** by Board of Directors of the Institute for Graduate School of Science and Engineering.

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU

Director of the Institute of
Graduate School of Science and Engineering

YAYINLAMA VE FÍKRÝ MÜLKÝYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin/raporumun tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe Üniversitesine verdigimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğunu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanılması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

Tezimin/Raporumun tamamı dünya çapında erişime açılabilir ve bir kısmı veya tamamının fotokopisi alınabilir.

(Bu seçenekle teziniz arama motorlarında indekslenebilecek, daha sonra tezinizin erişim statüsünün değiştirilmesini talep etseniz ve kütüphane bu talebinizi yerine getirse bile, tezinin arama motorlarının önbelleklerinde kalmaya devam edebilecektir.)

Tezimin/Raporumun tarihine kadar erişime açılmasını ve fotokopi alınmasını (İç Kapak, Özeti, İçindekiler ve Kaynakça hariç) istemiyorum.

(Bu sürenin sonunda uzatma için başvuruda bulunmadığım taktirde, tezimin/raporumun tamamı her yerden erişime açılabilir, kaynak gösterilmek şartıyla bir kısmı ve ya tamamının fotokopisi alınabilir)

Tezimin/Raporumun tarihine kadar erişime açılmasını istemiyorum, ancak kaynak gösterilmek şartıyla bir kısmı veya tamamının fotokopisinin alınmasını onaylıyorum.

Serbest Seçenek/Yazarın Seçimi

23 / 06 / 2017


Necva Bölcü

ETHICS

In this thesis study, prepared in accordance with the spelling rules of Institute of Graduate School of Science and Engineering of Hacettepe University,

I declare that

- all the information and documents have been obtained in the base of the academic rules.
- all audio-visual and written information and results have been presented according to the rules of scientific ethics
- in case of using others works, related studies have been cited in accordance with the scientific standards
- all cited studies have been fully referenced
- I did not do any distortion in the data set
- and any part of this thesis has not been presented as another thesis study at this or any other university.

23/06/2017


Necva BÖLÜCÜ

ABSTRACT

Unsupervised Joint Part-of-Speech Tagging and Stemming For Agglutinative Languages

Necva BÖLÜCÜ

Master of Science, Computer Engineering Department

Supervisor: Asst. Prof. Dr. Burcu CAN BUĞLALILAR

June 2017, 108 pages

Part of Speech (PoS) tagging is the task of assigning each word an appropriate part of speech tag in a given sentence regarding its syntactic role such as verb, noun, adjective etc. Various approaches have already been proposed for this task. However, the number of word forms in morphologically rich and productive agglutinative languages is theoretically infinite. This variety in word forms causes sparsity problem in the tagging task for agglutinative languages. In this thesis, we aim to deal with this problem in agglutinative languages by performing PoS tagging and stemming simultaneously. Stemming is the process of finding the stem of a word by removing its suffixes. Joint PoS tagging and stemming reduces sparsity by using stems and suffixes instead of words. Furthermore, we incorporate semantic features to capture similarity between stems and their derived forms by using neural word embeddings.

In this thesis, we present a fully unsupervised Bayesian model using Hidden Markov Model (HMM) for joint PoS tagging and stemming for agglutinative languages. The results indicate that using stems and suffixes rather than full words outperforms a simple word-based Bayesian HMM model for especially agglutinative languages. Combining semantic features yields a significant improvement in stemming.

Anahtar Kelimeler: unsupervised learning, part-of-speech (PoS) tagging, stemming, Bayesian learning, Hidden Markov model (HMM), semantic, neural word embeddings

ÖZET

Sondan Eklemeli Dillerde Gözetimsiz Eşzamanlı Sözcük Türü İşaretleme ve Gövdeleme

Necva BÖLÜCÜ

Yüksek Lisans,Bilgisayar Mühendisliği

Danışman: Yrd. Doç. Dr. Burcu CAN BUĞLALILAR

Haziran 2017, 108 sayfa

Sözcük türü işaretleme, cümledeki fil, isim, sıfat v.b. sözdizimsel rolüne bakarak her bir sözcüğe uygun etiketin atanmasıdır. Bu işlem için çeşitli yöntemler önerilmiştir. Morfolojik olarak zengin ve üretken sondan eklemeli dillerde sözcük formlarının sayısı teorik olarak sonsuzdur. Sözcük formlarındaki bu çeşitlilik, sondan eklemeli dillerde etiketleme işleminde seyreklik problemi yaratmaktadır. Bu tezde sözcük türü işaretleme ve gövdeleme işlemlerini eşzamanlı gerçekleştirerek sondan eklemeli dillerde bu problemin üstesinden gelmeyi amaçlamaktayız. Gövdeleme, bir sözcüğü eklerinden ayırarak gövdeyi bulma işlemidir. Birleşik sözcük türü işaretleme ve gövdeleme, sözcükler yerine gövde ve ekler kullanarak seyreklik problemini azaltmaktadır. Ayrıca, gövde ve gövdeden türetilmiş sözcük arasındaki benzerliği yakalamak için anlamsal özelliklerden yararlanmaktadır.

Bu tezde, sondan eklemeli dillerde birleşik sözcük türü işaretleme ve gövdeleme işlemi gerçekleştirmek için tamamen gözetimsiz Bayesian Saklı Markov modeli sunulmuştur. Sonuçlar, özellikle sondan eklemeli diller için sözcükler yerine gövdeler ve eklerinin kullanılmasının sözcük tabanlı Bayesian HMM modelinden daha iyi olduğunu göstermektedir. Anlamsal özelliklerin eklenmesi ise gövdelemede belirgin bir iyileşme göstermektedir.

Keywords: gözetimsiz öğrenme,sözcük türü işaretleyici, gövdeleme, Bayesian öğrenme, saklı Markov model

ACKNOWLEDGEMENTS

First and foremost, I would like to wholeheartedly thank to my excellent supervisor Asst. Prof. Dr. Burcu Can Buğlalılar for her endless patience, valuable advice, encouragements and immeasurable amount of guidance in this thesis. At every stage of this thesis, she supported me with her knowledge. I can say for sure that I have always felt fortunate to work under her inspiring supervision.

Besides I would like to thank my thesis committee members for insightful comments for this thesis.

In addition, I would like to thank everybody who supported and contributed to this study. Especially, I would like to thank my office mate Selma Dilek; she was not only an office mate but also a sincere friend. I am also obliged to my reading group friends for their friendship and support.

Finally, I thank my beloved family for their continual support throughout my educational life. They have always believed in me and encouraged me with their best wishes.

This work is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) with the project number EEEAG-115E464.

CONTENTS

	<u>Page</u>
ABSTRACT	i
ÖZET	iii
ACKNOWLEDGMENTS	v
CONTENTS	vi
FIGURES	viii
TABLES	x
ABBREVIATIONS	xi
1. INTRODUCTION.....	1
1.1. Overview	1
1.2. Motivation	2
1.3. Research Questions	3
2. BACKGROUND	5
2.1. Linguistic Background.....	5
2.2. Machine Learning Background.....	9
2.3. Inference.....	13
2.4. Conclusion.....	14
3. RELATED WORK	15
3.1. Introduction	15
3.2. Literature Review on Unsupervised Part of Speech Tagging	15
3.3. Literature Review of Cooperative Learning of Part of Speech Tagging	21
3.4. Literature Review on Stemming.....	22
3.5. Conclusion.....	28
4. MODEL.....	29
4.1. Introduction	29
4.2. Baseline Bayesian HMM Model	29
4.3. Joint Models for PoS Tagging and Stemming	31

5. EXPERIMENTS AND RESULTS	42
5.1. Datasets	42
5.2. Evaluation Metrics	43
5.3. Experiments	46
5.4. Conclusion.....	63
6. CONCLUSION.....	65
6.1. Conclusion.....	65
6.2. Future Research Directions	66
A APPENDIX : PoS TAGSET REDUCTION	67
B APPENDIX : Word2vec DATA.....	70
C APPENDIX : RESULTS FOR 12K DATASETS	71
REFERENCES	80

FIGURES

	<u>Page</u>
2.1. Structure of a typical word in an agglutinative language	7
2.2. Stem of word <i>geçmiş</i> according to different PoS	7
2.3. A Bayesian network specifying conditional independence relations for a hidden Markov model.	9
2.4. An illustration of the CRP	13
3.1. The binary tree obtained from Brown clustering	16
3.2. Bigram HMM	17
3.3. Trigram HMM.....	17
3.4. Contextualised HMM Tagger.....	18
3.5. Infinite HMM Tagger	20
3.6. Joint PoS tagging and segmentation proposed by Sirts and Tanel [1]	21
4.1. The plate diagram of the Bayesian HMM with symmetric Dirichlet priors.	30
4.2. The plate diagram of the stem based Bayesian HMM.	32
4.3. The plate diagram of the stem and suffix-based Bayesian HMM.	34
4.4. Dependency of suffixes in an example Turkish sentence.	37
4.5. The plate diagram of the affix transition-based Bayesian HMM.	38
4.6. Stem and Transition based Bayesian HMM.....	40
5.1. Example sentence with its specific and corresponding universal POS tags.	43
5.2. Sensitivity of hyperparameter sets for PoS tagging performance in Turkish....	47
5.3. Sensitivity of parameter set for stemming performance of Turkish	50
5.4. Features summary of proposed model for Turkish	50
5.5. Sensitivity of dataset for PoS performance of Hungarian	51

TABLES

2.1. PoS tag list proposed by Petrol et al. [2]	8
5.1. Datasets used in the experiments	42
5.2. Turkish PoS tagging results for different hyperparameter sets	48
5.3. Turkish stemming results for different hyperparameter sets	49
5.4. Hungarian24k PoS tagging results for different hyperparameter sets	52
5.5. Hungarian24k stemming results for different hyperparameter sets	54
5.6. Finnish24k PoS tagging results for different hyperparameter sets	55
5.7. Finnish24k stemming results for different hyperparameter sets	56
5.8. Basque24k PoS tagging results for different hyperparameter sets	57
5.9. Basque24k stemming results for different hyperparameter sets	58
5.13. Examples to correct and incorrect stems of Turkish.....	59
5.14. Examples to correct and incorrect stems of Hungarian	59
5.10. Penn 24K PoS tagging results for different hyperparameter sets	60
5.11. udEnglish 24K PoS tagging results for different hyperparameter sets	61
5.12. udEnglish 24K stemming results for different hyperparameter sets	62
5.15. Examples to correct and incorrect stems of Finnish	63
5.16. Examples to correct and incorrect stems of Basque	63
5.17. Examples to correct and incorrect stems of English.....	63
1.1. The mapping of the Universal tagset to the Penn Treebank tagset.....	67
1.2. The mapping of the Universal tagset to the FinnTreeBank tagset	67
1.3. The mapping of the Universal tagset to UD Basque TreeBank tagset	68
1.4. The mapping of the Universal tagset to UD Hungarian TreeBank tagset	68
1.5. The mapping of the Universal tagset to UD English TreeBank tagset.....	69
1.6. The mapping of the Universal tagset to the Metu-Sabancı Turkish Treebank tagset.....	69
3.1. Hungarian12K PoS tagging results for different hyperparameter sets.....	71

3.2.	Hungarian12k stemming results for different hyperparameter sets	72
3.3.	Finnish 12K PoS tagging results for different hyperparameter sets	73
3.4.	Finnish 12K stemming results for different hyperparameter sets	74
3.5.	Basque 12K PoS tagging results for different hyperparameter sets.....	75
3.6.	Basque 12K stemming results for different hyperparameter sets	76
3.7.	Penn 12K PoS tagging results for different hyperparameter sets	77
3.8.	udEnglish 12K PoS tagging results for different hyperparameter sets	78
3.9.	udEnglish 12K stemming results for different hyperparameter sets	79

ABBREVIATIONS

CRF	Conditional Random Fields
CRP	Chinese Restaurant Process
CW	Chinese Whispers
ddCRP	distance independent CRP
DP	Dirichlet Process
EM	Expectation Maximization
FSM	Frakes and Fox Similarity Metric
GRAS	GRaph-based Stemmer
HDP	Hierarchical Dirichlet Process
HMM	Hidden Markov Model
HPS	High Precision Stemmer
ICF	Index Compression Factor
iHMM	infinite HMM
IR	Information Retrieval
KL	Kullback-Leibler
LSA	Latent Semantic Analysis
MAP	Maximum a Posteriori
MCMC	Markov Chain Monte Carlo
MCRS	Mean Number of Characters and Removed in forming Stems
MDL	Minimum Description Length
MEM	Maximum Entropy Model
MHD	Mean and Median Modified Hamming Distance
ML	Maximum Likelihood

MLE	Maximum Likelihood Estimation
MMI	Maximum Mutual Information
MWC	Mean Number of Words per Conflation Class
NLP	Natural Language Processing
NMI	Normalized Mutual Information
NWSF	Number of Words and Stems diFfer
OOV	out-of-Vocabulary
PMF	Probability Mass Function
PoS	Part of Speech
RF	Relative Frequency
SVD	Singular Value Decomposition
TTS	Text to Speech
VI	Variation of Information
WSJ	Wall Street Journal
YASS	Yet Another Suffix Striper

1. INTRODUCTION

1.1. Overview

Parts of speech play a crucial role in defining the structure and meaning of a sentence in any language. Words can be labeled with different parts of speech depending their syntactic roles in the sentence. Assigning each word a part of speech such as noun, verb, adjective, etc. is called **Part of Speech (PoS) tagging** task in Natural Language Processing (NLP). It is one of the early tasks in NLP. PoS taggers take a sentence as input and generate a list of tuples (word/tag) as output, where each word is assigned to related tag.

Example The sentence

Bunu zaten biliyordum. (I have already known that.) is tagged as:

Bunu/Pron zaten/Adv biliyordum/Verb ./Punc

This task determines the syntactic features of the words such as gender, tense, etc. [3].

Stemming is the process of removing inflectional affixes from a word. The aim of stemming is to reduce the morphological variants to a linguistically correct stem from which all different word forms are derived.

Example : *kitaplar* (*books*), *kitapta* (*in the book*), *kitaplarum* (*my books*) have the same stem *kitap* (*book*)

PoS tagging and stemming have been playing significant roles in several NLP applications. Thus, small improvements on these tasks have the potential to yield larger improvements in many NLP tasks like Information Retrieval (IR), Linguistic Research, Text to Speech (TTS), Information Extraction and Shallow Parsing.

One of the challenges of PoS tagging is *ambiguity*. Many words can take several parts of speech. For example *booking* can be a noun (e.g. We made the booking three months ago.) or a verb (e.g. She is booking a table for four at their favorite restaurant.). Such a problem is common in many languages. The other challenge is *out-of-vocabulary (OOV)* problem. There will be many words which have not been seen in training.

1.2. Motivation

Agglutinative languages like Turkish are morphologically rich and productive. Turkish has nearly 23,000 stems and words formed by gluing suffixes to stems. Therefore, infinite number of words can be formed theoretically [4]. Due to rich morphology, these languages raises several challenges in PoS tagging and stemming.

There is a strong mutual relation between stemming and PoS tagging. Modeling joint PoS tagging and stemming helps to solve these challenges. Joint PoS tagging and stemming helps tackle the OOV problem by reducing the lexicon size. For instance, the words *kitaplarda* (*in books*), *kitaplar* (*books*), *kitap* (*book*), *kitapta* (*in the book*), *kitaplarım* (*my books*), *kitaptan* (*from the book*), *kitaplari* (*(their) books*), *kitapla* (*with the book*), *kitaplara* (*to the books*) are inflected from the stem *kitap* (*book*). By mapping the different word forms to the same stem, we can reduce the word forms to a single stem by also reducing the dictionary size and increasing the frequency of occurrence of the words. Joint PoS tagging and stemming also helps to determine how to split a word as a stem and a affix. For example, the words *koyun* can be split as *koy+un* (*put*) or *koyun+♯* (*sheep*) depending on its tag. PoS tag of the word helps to choose the correct stem.

Pipeline approaches solve tasks in order, for example stemming after then PoS tagging. One drawback of pipeline approaches is the error propagation where the errors accumulate in all stages. Joint models can avoid this kind of problem and achieve a better performance on both sub-tasks.

This is why a joint model would be more effective to handle PoS tagging and stemming instead of a pipeline approach [5].

Although supervised PoS tagging and stemming models perform better than unsupervised models, supervised models are applicable only to a set of well-studied languages that have labeled corpora available. However, more than 99% of the languages in the world are still considered less-studied and resource scarce [6]. Therefore, it indicates that developing unsupervised models is crucially needed for these languages.

In this thesis, we extend the fully unsupervised Bayesian PoS tagging model [7] for agglutinative languages. Instead of using words, we enhance the model by using stems, affixes and

semantic features. We primarily focus on Turkish as an agglutinative language. However, the models will be applicable to all languages.

1.3. Research Questions

These are the research questions that are aimed to be answered in this thesis:

- Can unsupervised PoS tagging be improved by integrating the stemming task jointly to the same learning mechanism? Does joint model help to reduce the sparsity problem in PoS tagging?
- Can we enhance stemming and PoS tagging results by integrating semantic features to the joint model?

1.3.1. Thesis Structure

The structure of the thesis is as follows:

Chapter 2 details essential background knowledge to understand the thesis. It starts with linguistic background, describes agglutinative languages and challenges of these languages. Then, we explain machine learning methods that we used in this thesis.

Chapter 3 provides an overview of the previous studies on PoS tagging and stemming. We focus on HMM for PoS tagging and unsupervised methods on stemming. We also discuss evaluation algorithms for PoS tagging and stemming. This chapter also presents studies on Turkish PoS tagging and stemming.

Chapter 4 describes a novel joint model in which PoS tagging and stemming are learned cooperatively and simultaneously. First, we present the baseline model that constructed on. Finally, the inference algorithm is described.

Chapter 5 reports our experimental results and compare PoS tagging and stemming results with other approaches in the literature for agglutinative languages and morphologically poor languages. We end this chapter with the analysis of parameters.

Finally, **Chapter 6** concludes this thesis with a brief summary of our work with contributions made to the fields of PoS tagging and stemming and proposes future topics to be studied based on the study in both fields.

2. BACKGROUND

In this chapter, we review background information to follow the approaches presented in this thesis. We start by explaining the linguistic background in Section 2.1.. Then, we focus on the machine learning background in Section 2.2..

2.1. Linguistic Background

“There are close on 7,000 languages in the world, and half of them have fewer than 7,000 Speakers each, less than a village. What is more, 80% of the world’s languages have fewer than 100,000 speakers.”(Ostler 2008)

The spoken languages in the world can be classified as follows: Inflective languages, agglutinative languages, isolating languages, and incorporating languages.

Inflective languages consist of stems with variable terminations or suffixes which were once independent words like Latin. Agglutinative languages consist of more than one, and possibly many morphemes. Examples of agglutinative languages are Turkish and Hungarian. Isolating language is a language in which meaning is created by supplemental words. Thus, almost every word consists of a single morpheme in the language. Latin, Spanish, English, Chinese, and Mandarin are examples of isolating languages. Incorporating languages are referred as polysynthetic languages. A single - though extensively long - word may represent an entire phrase, or even a sentence, including a verb, an adjective and even an object in incorporating. This language is often used to refer to Native American languages such as Alabama, Dakota.

In this chapter, we provide a brief description of the morphological structure of Turkish as an agglutinative language to ease the understanding of this thesis.

2.1.1. Morphology

Morphology is about the internal structure of words and operates with the subword units called *morphemes*. It is also an interface between phonology and syntax, where morphological forms as constituents carry both syntactic and phonetic information. For example, word

kitapçılar (*booksellers*) is composed from root *kitap* (*book*), and two bound morphemes *-çı* and *-lar*.

Agglutinative languages are morphologically productive languages that contain a set of rules for morphological composition that generate a considerable amount of word forms by the concatenation of morphemes [8].

Morphemes can be either *roots* or *affixes*. Affixes can be either *inflectional* or *derivational*. Roots can take derivational and inflectional affixes; therefore, a root can be seen in a large number of different word forms. Various suffixes and their combinations make a complex problem to find stems in agglutinative languages.

Example Some of the word forms that are built from the root *başar(-mak)* ((*to*) *succeed*) are as follows:

başar(-mak) - ((*to*) *succeed*)

başarı - (*success*)

başarisız - (*unsuccessful*)

başarisızlaş(-mak) - ((*to*) *become unsuccessful*)

başarisızlaştır(-mak) - ((*to*) *make one unsuccessful*)

başarisızlaştırıcı - (*maker of unsuccessful ones*)

başarisızlaştırıcılaş(-mak) - ((*to*) *become a maker of unsuccessful ones*)

başarisızlaştırıcılaştır(-mak) - ((*to*) *make one a maker of unsuccessful ones*)

Inflectional suffixes add appropriate syntactic features such as gender, tense, etc. [3] to the word whereas derivational suffixes change the meaning of the word. For instance, the suffix *-ler* in the word *kalemler* (*pencils*) is inflectional because it marks the plurality of the word *kalem* (*pencil*) and *kalem* (*pencil*) and *kalemler* (*pencils*) share the same meaning. The suffix *-gi* in the word *silgi* (*eraser*) is derivational because it changes the meaning of the word from an action to a tool. Here, the derivational suffix also changes the PoS tag of the word.

A stem is the base of an inflected word. The stem of a word does not necessarily have to be indivisible and can consist of a root that has derivational suffixes attached to it.

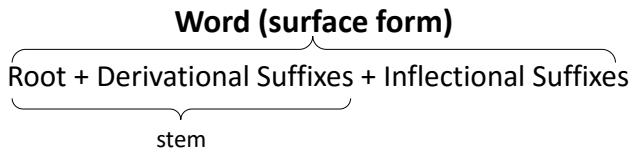


Figure 2.1. Structure of a typical word in an agglutinative language

Figure 2.1. shows how a word is generated through inflection and derivation. Roots are transformed into stems with derivational suffixes.

For example, the word *kitapçı* (*bookseller*) is a stem and it can be used to derive the plural form *kitapçılar* (*booksellers*) by adding the inflectional suffix *-lar*.

One of the challenging problem of agglutinative languages is that a word may have multiple meanings according to the stem and its PoS tag. For example *geçmiş* in Turkish may mean *past* as *adjective* or **passed** as *verb* depending on the context. In the *adjective* case, the stem is **geçmiş** whereas, in the *verb* case the stem is **geç** (see Figure 2.2.).

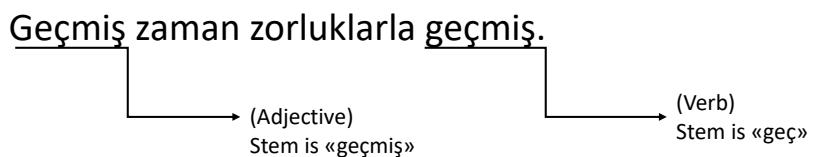


Figure 2.2. Stem of word *geçmiş* according to different PoS

2.1.2. Syntax

Syntax is a set of rules, principles and processes that govern the structure of sentences in a given language. According to the theory of universal grammar that originates from the work of Chomsky (1965) [9], “Every language has its own syntax, however languages share a common set of properties which are limited in the human brain, and that makes them universal”.

Under syntactic rules, part of speech categories such as *noun*, *verb* or *preposition* designate a group of words with certain morphosyntactic properties. These can be divided into two

Table 2.1. PoS tag list proposed by Petrol et al. [2]

Tag	Definition	Example
VERB	Verbs (all tenses and modes)	gitmiş, gelecek, yüzüyor
NOUN	Nouns (Common and proper)	kitap, Ahmet, gözlük
PRON	Pronouns	Ben, onlar
ADJ	Adjectives	sıcak, genç, küçük
ADV	Adverbs	İçeri, hızlıca
ADP	Adpositions (prepositions and postpositions)	gibi, değil, üzere
CONJ	Conjunctions	fakat, oysaki, üstelik
DET	Determiners	bir, bu
NUM	Cardinal numbers	onbeş, iki
PRT	particles or other function words	göre, kadar
X	Other (foreign words, types, abbreviations)	TDK, THY
.	Punctuation	?, !, :

categories: **closed class** types and **open class** types. Closed classes have fixed number of members, whereas open classes may accept many members, thereby they can infinite number of members.

There are four main open classes; **noun**, **verb**, **adjective**, **adverb**.

Noun class includes the words that mostly correspond to people, places, or other things.

The **verb** class includes the words referring to actions e.g. *git(mek)* ((to) go), *bil(mek)* ((to) know), *konusmak* ((to) talk).

The **adverb** class describes and gives information about a verb, adjective, adverb or phrase. For instance, in sentence “*Hızlı konuşurum.*” (“I speak fast”), the adverb *hızlı* modifies the verb *konusurum*.

The **adjective** class modifies nouns and pronouns by describing a particular quality of the word. For example, in noun phrase *çalışkan öğrenci* (*hardworking student*), **çalışkan** modifies student.

Closed classes differ from language to language differently from open classes. Major closed classes are **prepositions**, **determiners**, **pronouns**, **conjunctions**, **participles**, **numerals**.

Petrov et al. (2011) [2] propose a Universal PoS tag set that defines 12 universal categories.

2.2. Machine Learning Background

2.2.1. Hidden Markov Models

A Hidden Markov Model (HMM) is a method for representing probability distributions over sequences of observations. A sequence of hidden states (S_1, S_2, \dots) is generated according to a Markov process. Conditioned on the hidden states, we observe (Y_1, Y_2, \dots) where it is assumed that the Y_i is conditionally independent of everything else given the S_i and the S_{i+1} is conditionally independent of everything else given the S_i .

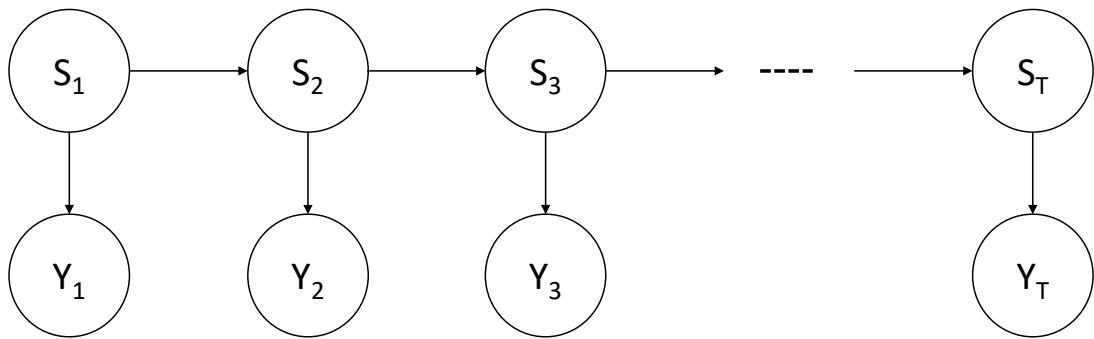


Figure 2.3. A Bayesian network specifying conditional independence relations for a hidden Markov model.

2.2.2. Bayesian Modeling

Bayesian modeling defines the probability of an instance with respect to value of parameters, latent variables or hypotheses. A Bayesian model can be parametric or non-parametric. The Bayesian parametric models have predefined number of parameters. The Bayesian non-parametric models have countably infinite parameters that grows with data. Bayesian modeling derives from Bayes' theorem:

$$p(\theta|S) = \frac{p(S|\theta)p(\theta)}{p(S)} \quad (1)$$

where $p(\theta|S)$ is *posterior distribution* of the parameters θ , $p(S|\theta)$ is the *likelihood* and $P(\theta)$ is the prior probability. The normalization constant is given as follows:

$$p(S) = \int p(S, \theta)p(\theta) \quad (2)$$

It is also called the *marginal likelihood*.

2.2.2.1. Conjugate Priors

Given a likelihood, the conjugate prior is the prior distribution such that the prior and posterior are in the same family of distributions. For example, given a likelihood $p(x|\theta)$, we choose a family of prior distributions such that

$$p(x) = \int p(X|\theta)p(\theta)d(\theta) \quad (3)$$

where θ is a set of parameters that are integrated out without being estimated. Additionally, we choose prior to posterior updating yields a posterior which is in this family.

Conjugate priors reduce Bayesian updating by modifying the parameters of prior distribution rather than computing integrals. Thus, they are widely used in practice. Dirichlet distribution is the conjugate prior for Multinomial distributions.

2.2.2.2. Dirichlet-Multinomial

The conjugation of a Multinomial distribution with a Dirichlet prior results in a posterior distribution with a Dirichlet distribution form. Defining a Multinomial distribution on $\{1, \dots, N\}$ possible outcomes and setting θ helps us to define hyperparameters. Here hyperparameters are parameters of the prior distribution when we assume that θ is following some prior distribution. For the Dirichlet distribution prior, we can say that β is a hyperparameter.

$$\begin{aligned} x_i &\propto \text{Multinomial}(\theta) \\ \theta &\propto \text{Dirichlet}(\beta) \end{aligned} \quad (4)$$

where x_i is drawn from a Multinomial distribution with parameter θ and parameter θ is drawn from a Dirichlet distribution with hyperparameter β .

2.2.2.3. Multinomial Distribution

Multinomial distribution is the probability distribution of the outcomes in a Multinomial experiment. The Multinomial distribution arises when each datum in one of K possible outcomes with a set of probabilities $\{x_1 \dots x_k\}$. Multinomial models the distribution that indicates how many times each outcome is observed over N total number of data points:

$$p(x|\theta) = \frac{N!}{\sum_{k=1}^K n_k!} \prod_{k=1}^K \theta_k^{x_k} \quad (5)$$

Here parameters θ_k are the probabilities of each data point k , and n_k is the number of occurrences of data point x_k and:

$$N = \sum k = 1K n_k \quad (6)$$

2.2.2.4. Dirichlet Distribution

Dirichlet distribution is a way to model random Probability Mass Function (PMF) for finite sets. It is often used as the prior distribution in Bayesian inference and it is the conjugate of the Categorical distribution and Multinomial distribution. Dirichlet distribution follows the form:

$$p(\theta|\beta) = \frac{1}{B(\beta)} \prod_{k=1}^K \theta_k^{\beta_k - 1} \quad (7)$$

where $\beta = (\beta_1, \beta_2, \dots, \beta_K)$ denotes the concatenation parameters, $K \geq 2$ denotes the number of categories, and $B(\beta)$ is a normalizing constant in a Beta function form:

$$B(\beta) = \frac{\prod_{k=1}^K \Gamma(\beta_k)}{\Gamma(\sum_{k=1}^K \beta_k)} \quad (8)$$

where Γ is the generalization of the factorial function defined as $\Gamma(t) = (t - 1)!$ for positive integers.

2.2.2.5. Bayesian Posterior Distribution

In a conjugate Bayesian analysis, we have a Multinomial likelihood with the Dirichlet prior. The posterior distribution of parameters is given in formula 9. This leads to a Bayesian posterior $Dirichlet(n_k + \beta_k - 1)$.

$$\begin{aligned} p(\theta|x, \beta) &\propto p(x|\theta)p(\theta|\beta) \\ &= \frac{N!}{\prod_{k=1}^K n_k!} \frac{\prod_{k=1}^K \Gamma(\beta_i)}{\Gamma(\sum_{k=1}^K \beta_i)} \prod_{k=1}^K \theta_k^{n_k + \beta_k - 1} \\ &\propto Dirichlet(n_k + \beta_k - 1) \end{aligned} \tag{9}$$

2.2.2.6. Predictive Distribution for Dirichlet-Multinomial

The predictive distribution is the distribution of observation x_{N+1} given the observations $X = (x_1, \dots, x_n)$:

$$\begin{aligned} p(x_{N+1} = j|X, \beta) &= \int (x_{N+1} = j|x, \theta)(\theta|\beta)d\theta \\ &= \frac{n_j + \beta_j}{N + \sum_{k=1}^K \beta_k} \end{aligned} \tag{10}$$

This shows a rich-get-richer behavior, where if the frequency of the previous observations in a given category are higher, then the next observation x_{N+1} has a higher probability of being in the same category.

2.2.2.7. Chinese restaurant process (CRP)

Chinese Restaurant Process (CRP) is distribution over partitions. It is a random process where there is a Chinese restaurant with infinite number of tables. Each table has a menu to serve. The first customer sits at the first table. The second customer decides either to sit with the first customer or by herself at a new table. In general, n^{th} customer sits at an occupied table k with probability that is proportional to the number n_k of customers who are already sitting at the table or sits at a new table with probability proportional to α . While this process continues, tables with preferable menus will acquire a higher number of customers. Thus, the rich-get-richer principle shapes the structure of the tables.

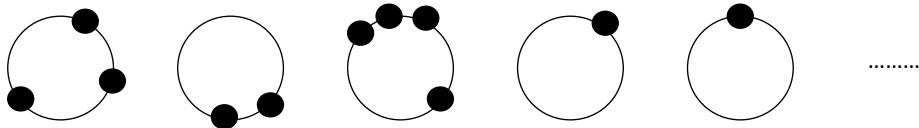


Figure 2.4. An illustration of the CRP

2.3. Inference

In machine learning, inference of parameters is an essential part of the learning mechanism. There are various approaches such as Maximum Likelihood (ML) or the Maximum a Posterior (MAP) to perform a point estimation of the parameters. Bayesian inference gives an estimation of distribution over the possible values of the parameters instead of a point estimation. Sampling by drawing random samples from a distribution is one of the approaches in estimating parameters. We use Markov Chain Monte Carlo (MCMC) for the estimation. Following section gives a brief overview about this method.

2.3.1. Markov Chain Monte Carlo (MCMC)

A Markov Chain is a mathematical system that experiences transitions from one state to another according to certain probabilistic rules. MCMC is an estimation technique that simulates a Markov Chain to generate samples from a probability distribution in a high dimensional space. This stochastic process is described in terms of a conditional probability:

$$P(X_{n+1} = x | X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) = P(X_{n+1} = x | X_n = x_n) \quad (11)$$

The possible values of X_i are drawn from a countable set S , which is the state space of the chain.

Metropolis-Hastings and Gibbs sampling are two well-known examples of the set of MCMC algorithms.

2.3.1.1. Gibbs Sampling

Gibbs sampling is a simple and widely used method for generating random samples from a joint distribution when this distribution is not known or is difficult to calculate. Let $X = (x_1, x_2, \dots, x_k)$ is a set of parameters and D is a set of observed data. In each iteration of Gibbs sampling, x_k sampled from the conditional distribution given x_{-k} (the set of all variables except x_k for $k = 1, 2, \dots, K$).

$$x_k \sim P(x_k | x_{-k}, D) \text{ for } k = 1, \dots, K \quad (12)$$

This process continues until *convergence* (the sample values have the same distribution as if they are sampled from the true posterior distribution).

2.4. Conclusion

In this chapter, essential background knowledge is presented to be referred throughout the thesis. As the thesis mainly focuses on morphology and syntax for agglutinative languages, a general overview of the two fields is given from the linguistic perspective based on basic terms and their definitions. Additionally, we present some statistical machine learning methods used for PoS tagging and stemming frequently.

3. RELATED WORK

3.1. Introduction

This chapter presents earlier work on unsupervised stemming and PoS tagging.

3.2. Literature Review on Unsupervised Part of Speech Tagging

PoS tagging is the task of assigning a syntactic category, e.g. noun, adjective for each word in a sentence. There are PoS tagging approaches such as Hidden Markov Model [10], Maximum Entropy Model [11], Decision Trees [12], Log Linear Models [13], clustering [14].

Learning in PoS tagging can be defined by, supervised, unsupervised, or hybrid learning. In this section, we concentrate on unsupervised approaches since the scope of this thesis consists of only unsupervised learning.

3.2.1. Clustering

This approach takes the advantage of distributional properties of words (similar words occur in similar contexts) by computing a context vector for each word to cluster into syntactic categories ([14], [15], [16], [17])

Brown et al. (1992) [14] present an approximate greedy hierarchical clustering algorithm that uses a bigram model to assign each word a latent class. Algorithm initializes each word type in separate cluster. Then a cluster pair is merged iteratively that cause a increase in the likelihood of the corpus according to a HMM. The probability of the corpus $w_1 \dots w_n$ is computed as follows:

$$P(w_1|c_1) \prod_{i=2}^n P(w_i|c_i) P(c_i|c_{i-1}) \quad (13)$$

where c_i is the class of w_i . The algorithm ends if no cluster pair is merged. At the end of the algorithm, a hierarchy of word types is obtained that can be presented as a binary tree as in Figure 3.1.

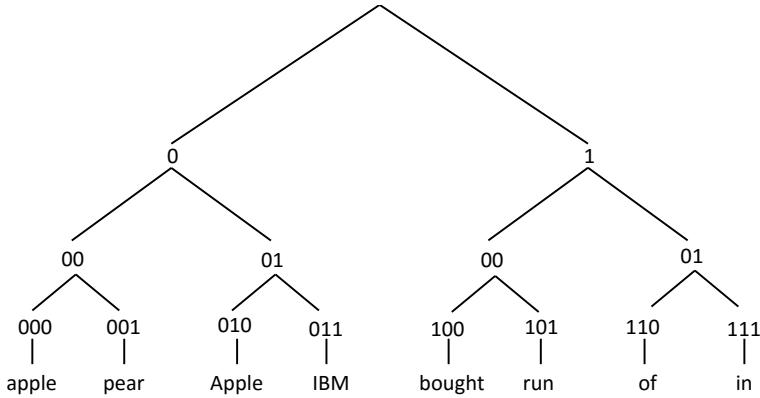


Figure 3.1. The binary tree obtained from Brown clustering

Finch and Chater (1992) [15] widen the idea of word clustering and collect global context vectors; i.e. the two preceding and the two following words of target words that are the 150 words with the highest frequency. Hierarchical clustering algorithm is applied on these vectors to acquire syntactic classes by using Spearman Rank Correlation Coefficient to measure linguistic similarity.

Schütze (1993) [18] uses two left and right words as context vectors. After obtaining words vectors, Singular Value Decomposition (SVD) is performed to reduce the dimension of the context matrix and then Buckshot clustering algorithm [10] is applied to build the clusters.

Schütze (1995) [19] applies Latent Semantic Analysis (LSA) with SVD based dimensionality reduction.

Clark (2000) [16] uses the distribution of the context in a flat clustering algorithm. Kullback-Leibler (KL) divergence is used to measure the divergence between clusters to decide whether to merge two clusters.

Biemann (2006) [17] uses Chinese Whispers (CW) graph clustering algorithm, based on the similarity in context. Unlike the other systems, this model doesn't need a clustering number as a parameter. Graph is constructed by the most frequent 10.000 words using their context statistics that are extracted from 150-250 feature words that appear immediately on the left or right of a target word.

3.2.2. Hidden Markov Models

One of the widely used approaches in PoS tagging is the HMMs([20]).

HMM assumes that there are K states $T = t_1, \dots, t_k$. These tags are hidden during the observation and they generate the word sequence $W = w_1, \dots, w_n$ observed in the corpus and the probability of the sentence is computed as follows with a first order assumption:

$$P(W, T) = P(w_1|t_1) \prod_{i=2}^n P(w_i|t_i)P(t_i|t_{i-1}) \quad (14)$$

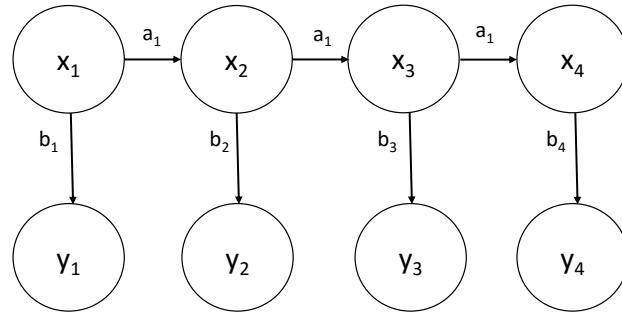


Figure 3.2. Bigram HMM

In the second order HMM, each tag is assumed to be dependent on the previous two tags in the history.

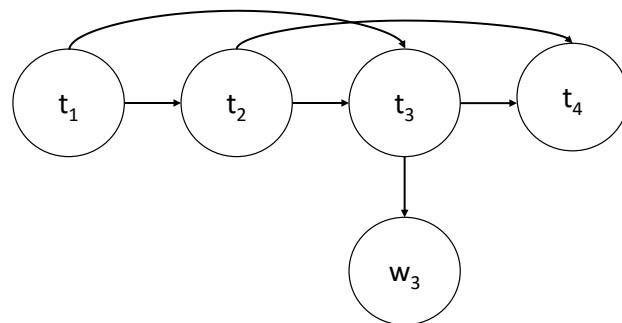


Figure 3.3. Trigram HMM

Merialdo (1994) [21] attempts to improve the trigram HMM PoS tagging by using Expectation Maximization (EM). The model uses a dictionary of possible tags for each word. Two different training a pro-supervised (Relative Frequency (RF)) and pro-unsupervised (ML / Forward Backward training) are applied. Two strategies are used for tagging:

- *Viterbi* computes the most probable tag sequence in a sentence
- *EM* computes the most probable tag for each word in a sentence

The paper concludes that ML training performs better on a small amount of labeled data , while RF gives more accurate results on a larger set of labeled data.

Banko and Moore (2004) [22] present a Contextualized HMM tagger and also do a comparative performance analysis on pre-existing strategies on the same data. The goal of contextualized HMM tagger is to include more context into tagging to estimate the probability of a word based on the tags immediately preceding and following it.

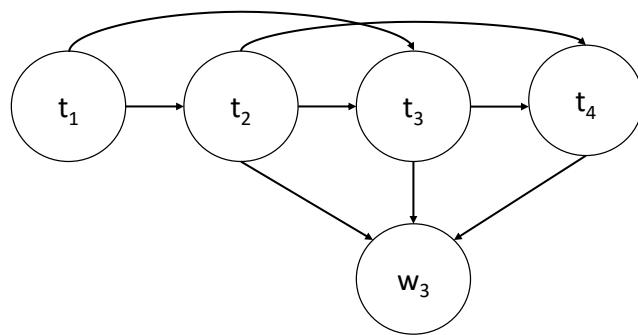


Figure 3.4. Contextualised HMM Tagger

3.2.3. Bayesian

Johnson (2007) [23] criticizes the standard HMM-EM approaches because of their poor performance on the unsupervised POS tagging due to their tendency to emit from each hidden state equal number of words. He adopts a Bayesian learning in an HMM model and compares the estimators used in HMM PoS taggers with the Bayesian estimator. The study shows the

drawbacks of EM [24] compared to Gibbs sampling [25] and Variational Bayes [26] estimators. The results show that training with EM gives poor results because of the distribution of hidden states.

Goldwater and Griffiths (2007) [7] propose a Bayesian approach adopted in a second order HMM with symmetric Dirichlet priors over transition and emission distributions:

$$\begin{aligned} t_i | t_{i-1} = t, \tau^{(t,t')} &\propto \text{Mult}(\tau^{(t,t')}) \\ w_i | t_i = t, \omega^{(t)} &\propto \text{Mult}(\omega^{(t)}) \\ \tau^{(t,t')} | \alpha &\propto \text{Dirichlet}(\alpha) \\ \omega^{(t)} | \beta &\propto \text{Dirichlet}(\beta) \end{aligned} \quad (15)$$

Gibbs sampling is used to estimate the parameters. Two sets of experiments are performed with fixed values of hyperparameters, and with the hyperparameter inference. The results show that Bayesian HMM increase the accuracy by up to 14% over Maximum Likelihood Estimation (MLE).

Remark: We adopt the PoS tagging algorithm of [7] for joint PoS tagging and stemming. Description of the algorithm is given in the Chapter 4.

Gao and Johnson (2008) [27] compare different estimators used in HMM PoS taggers and show that while Gibbs sampler performs better on small datasets with few tags, whereas Variational Bayesian performs better on large data sets.

Gael et al. (2009) [28] use the infinite HMM (iHMM) version of the non parametric HMM that also learns the number of hidden states. Dirichlet and Pitman-Yor processes are used on experiments. Shallow parsing task is used as an extrinsic evaluation of PoS tagging.

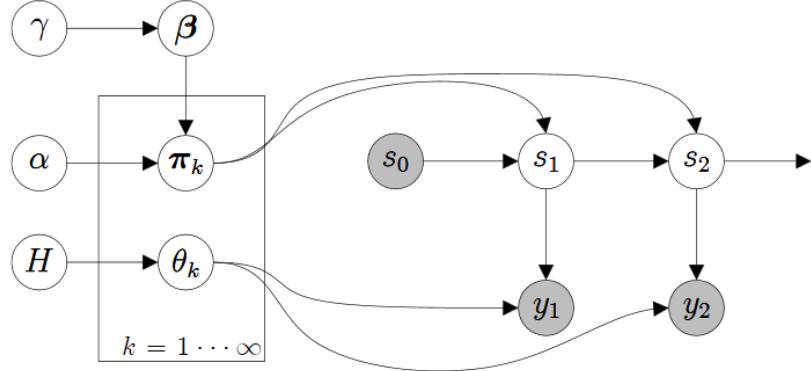


Figure 3.5. Infinite HMM Tagger

Stratos et al. (2016) [29] assume that each hidden state is linked with an observation state (anchor state). For instance, word “the” can appear only as a determiner tag. For this reason, this HMM model is called as anchor HMM.

3.2.4. Other Approaches

Eisner and Smith (2005) [13] use Conditional Random Fields (CRF) with contrastive estimation. They present a diluted dictionary, where infrequent words may have any tag. This method outperforms the EM and Bayesian HMM models.

Christodoulopoulos et al. (2010) [30] compare older systems and show that former one-tag-per word models tended to improve system performance by reducing model flexibility. They use prototype based features based on [31] with automatically induced prototypes.

Berg-Kirkpatrick et al. (2010) [32] use a log-linear model for PoS tagging. the authors use the morphology as a parameter in the sequence model to induce words that share the same tag to have same morphological features.

3.3. Literature Review of Cooperative Learning of Part of Speech Tagging

Qiu et al. (2012) [33] present a joint model that integrates two Markov chains for segmentation and PoS tagging . One of the chains is used for segmentation and the other one is used for PoS tagging. Results show that joint model outperforms traditional methods on Chinese segmentation and PoS tagging.

Sirts and Tanel (2012) [1] present a fully unsupervised non-parametric Bayesian model for joint PoS tagging and morphological segmentation. Model generates each word type with its tag and morphological segmentation and then proceed to generate HMM parameters by HDP. Standard HMM procedure is applied to generate the word itself, its tag, its segmentation.

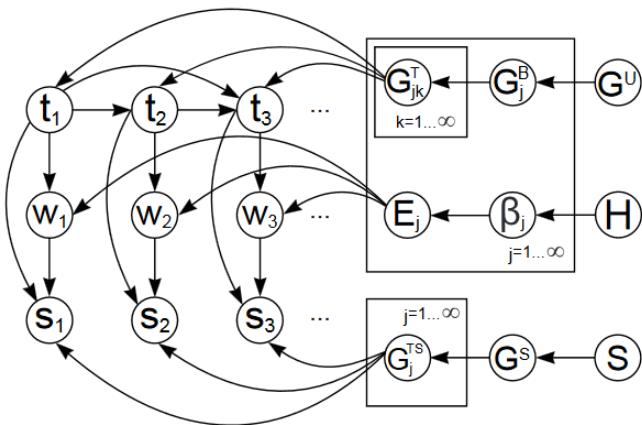


Figure 3.6. Joint PoS tagging and segmentation proposed by Sirts and Tanel [1]

Gibbs sampling is used for tagging and Metropolis-Hastings sampling is used for segmentation.

Sirts et al. (2014) [34] present a new approach that is a joint non-parametric Bayesian model combining morphological and distributional information based on distance independent Chinese Restaurant Process (ddCRP). ddCRP is an extension of CRP and defines a distribution over partitions of data points. In CRP, each customer chooses a table based on a probability proportional to the number of customers who are already sitting at that table, whereas in ddCRP, a customer follows another customer and sits at the same table with that customer.

Prior is given as:

$$P(c_i = j) \propto \begin{cases} f(d_{ij}) & \text{if } i \neq j \\ \alpha & \text{if } i = j \end{cases} \quad (16)$$

where c_i is the index of the customer followed by customer i , f is a decay function, d_{ij} is the distance between i and j .

Word embeddings are used for distributional features to assess the similarity between words.

3.3.1. PoS Tagging of Turkish

Oflazer and Kuruöz (1994) [35] and **Oflazer and Tür (1997)** [36] propose a rule based approach for Turkish PoS tagging.

Hakkani-Tür et al.(2000) [37] introduce a statistical approach for morphological disambiguation.

Altinyurt et al. (2006) [38] combine rule based and statistical approaches to build a PoS tagger. This tagger uses word frequencies and n-gram statistics.

Dinçer et al. (2008) [39] propose a stochastic PoS tagger for Turkish for information retrieval task. They define seven different lengths of word endings are used in their model. The best accuracy is obtained with 5 letters by 90.2%

Kentool [40] presents a PoS tagger for Turkish based on a full scale two-level morphological specification of Turkish.

3.4. Literature Review on Stemming

Stemming is a linguistic process based on removing affixes from a word to produce a common form of the word. For example, the words *playing*, *plays*, *played* might be stemmed to the base form *play*. Stemming algorithms have been studied since the 1960s. We can categorize stemmers in **three** classes.

1. Rule-based

2. Statistical

3. Hybrid

We mainly focus on statistical approaches since the scope of this thesis is limited to unsupervised learning.

3.4.1. Rule-based Stemmers

Rule-based stemmers rely on specific rules on a given language. This type of stemmers generally remove suffixes from word endings based on manually defined transformation rules.

Some of the well-known rule-based stemmers are by Lovins [41], Dawson [42], Porter [43], Paice/Husk [44], and Krovetz [3]. complicated stemmer due to linguistic morphology.

3.4.2. Statistical Stemmers

The recent stemmers are mostly based on statistical methods. The advantage of these stemmers is that they can obviate the language specific knowledge. Therefore, they are usually language independent. A number of studies [45], [46], [47], [48] and [49] have shown that statistical stemmers are good substitutes to language-specific stemmers, especially for languages where linguistic resources are not sufficient.

The statistical stemmers use different methods like HMM, Maximum Entropy Model (MEM), Graph-based methods, Minimum Description Length principal (MDL).

The successor variety approach has been used firstly by **Harris(1955)** [50] to determine the suffixes without any prior knowledge of the language. The method calculates the number of distinct letters following a successor letter in a word to find the break-point where the successor variety increases sharply. The main idea behind this is that the letter at any position is dependent on the letters preceding it and dependency increases as we move towards the stem. Once the count of the successor and predecessor letters are available , different features are used to find the stem, such as peak and plateau, successor/predecessor entropies.

Xu et. al. (1998) [51] analyze the cooccurrence statistics of words to cope with the drawbacks of the Porter stemmer [43]. For instance, in the Porter stemmer the words *policy* and

police are conflated although they have different meanings but the words *index* and *indices*' are not conflated although they have the same root.

Goldsmith (2001) [52] proposes an information theoretic morphological segmentation system based on MDL. The best segmentation of the word is the one that minimizes the total compressed length of the corpus. For example, *laughing*, *laughs*, *walked*, *walking*, *walks*, *jumped*, *jumping*, *jumps* are grouped as $\{laugh, walk, jump\}$ and suffixes are grouped as $\{ed, ing, s\}$ that is called a signature. This method is implemented as Linguistica [53].

Bacchin et al. (2002) [54] propose a graph-based algorithm for stemming. In the first step, the method splits the words at every possible split points to form a set of substrings. Then, the sets of substrings are used to build a directed graph to determine the prefix and suffix scores based on frequencies of substrings. The best split point is determined by the maximum probability of a suffix-prefix pair.

Melucci and Orio (2003) [55] present an HMM based stemmer. The letters are represented as the states in the HMM. The states correspond to either prefixes or suffixes. Rules are defined for transitions. The parameters are estimated by EM algorithm. Once the parameters are estimated, the path that has the maximum probability generates a segmentation of a word, where the first part is considered as the stem.

McNamee and Mayfield (2004) [56] propose an alternative stemming algorithm that uses letter n -grams. Digrams or trigrams are generated for each word. For example, following bigrams and trigrams are generated from the word *kalemler*:

k*, *ka*, *al*, *le*, *em*, *ml*, *le*, *er*, *r
k*, **ka*, *kal*, *ale*, *lem*, *eml*, *mle*, *ler*, *er**, *r

The basic intuition of this approach is that similar words share common n -grams, and n -gram frequencies of an inflected form of a word are less than its stem. In other words, similar words will share a high proportion of n -grams.

Bacchin et al. (2005) [57] extend the graph-based stemmer introduced in [54] to discover stems and derivations using mutual reinforcement relationship between stems and suffixes. Initially, a set of probable substrings are generated by splitting each word at all positions. Then, a directed graph is built where nodes represent substrings and a directed edge is inserted between node x and node y if there is a word z such that $z = xy$. The estimation of affix scores are calculated by HITS algorithm [58]. Once the prefix and suffix scores are

estimated, the algorithm finds the most probable split point by maximizing the likelihood of prefix and suffix pairs of each word in the word list.

Peng et al. (2007) [59] suggest context sensitive stemming using distributional similarity of words for the information retrieval task. Each query is expanded with the morphological variants of the query term. Additionally, bigrams are used for contextual features. For example, when stemming is applied on *developing*, *developed*, *develops*, *development*, they are all reduced to *develop*. Using bigrams may lead to selecting *develops*.

Majumder et al. (2007) [49] develop a statistical stemmer called YASS (Yet Another Suffix Striper) that adopts a complete linkage clustering algorithm by using a string distance measure. After the calculation of string similarity based on the string distance measure, the clusters (presumably morphologically related) are created using a graph-based complete linkage clustering algorithm.

Paik and Parui (2011) [60] present an unsupervised algorithm that collects the potential suffixes based on their cooccurrence frequency and then groups each word based on common prefix based on given length. Strength of the common prefix of each class is measured by integrating the potential suffix information. If strength measure is good enough, then it is considered as the root of the class. Otherwise, another root from the class is found iteratively.

Paik et al. (2011) [47] introduce GRAph-based Stemmer (GRAS) that is a statistical stemmer that groups words to find suffix pairs. The algorithm searches common prefixes among word pairs. For example, let two words $W1 = P + S1$ and $W2 = P + S2$ where p is the longest common prefix between $w1$ and $w2$. The suffix pair $s1$ and $s2$ is a valid suffix pair if there is a common prefix followed by these suffixes in other word pairs. A weighted graph $G = (V, E)$ is built by using these suffix pairs. Each vertex of G represents a word in the lexicon and each weighted edge $w(u, v)$ represents the frequency of the suffix pair between the vertices u and v . Then the graph is decomposed to generate classes of related words.

Paik et al. (2011) [46] propose a stemming algorithm that is also based on cooccurrence statistics of words in the corpus. A graph is built where the word variants are vertices and two word variants forms and edge weighted by frequency of word variant pairs. Thus, this is a neighbor-based algorithm that can to find morphologically related words.

Paik (2013) [48] presents another stemming algorithm. Morphologically related words are clustered by using cooccurrence information that enables query independent search in the information retrieval task.

Brychcin and Konopik (2015) [45] present High Precision Stemmer (HPS) that is a statistical approach that uses orthographic and semantic information. This method works in two steps. In the first step, Maximum Mutual Information (MMI) clustering is used to cluster orthographically and semantically similar words. The word similarity is based on the longest common prefix. The second step uses a maximum entropy classifier on the clusters obtained from the first step. The classifier uses orthographic and semantic features of words to split word into their stems and suffixes. Brychchin and Konopik evaluate the performance of their stemmer on different size of data size and report that better results could be achieved with only 50.000 words. HPS, as reported in the paper, outperforms YASS [49], GRASS [47], and Linguistica [52]. Moreover, the authors train HPS in four major language families and six languages (i.e. Spanish, Polish, Hungarian, Czech, and Slovak). The results show that, HPS performs both well on seen and unseen data. The weakness of the HPS is the computational complexity especially on large datasets.

3.4.3. Hybrid Stemmers

Hybrid stemmers combine the rule-based and statistical approaches. This combination generally helps in increasing the performance of the stemmer.

Some of the hybrid stemmers are [61], [62], [63], [64], [65], [66].

3.4.4. Previous Work on Turkish Stemming

In this section, we summarize the stemming methods proposed for the Turkish language.

Köksal (1979) [67] proposes an early stemming algorithm that takes a fixed length of the initial part of the word as the stem. 5-6 letters gives the best results. However, a fixed length performance well in information retrieval task, whereas another length performs better on a different task. This shows that there is no common fixed length for different tasks. It is a simple approach but the results show that taking a fixed length improves the IR performance for Turkish.

Oflazer's (1994) morphological analyzer [68] uses a stem list and structural analysis to yield all possible analyses a given word.

Solak et al. (1994) [69] present AF algorithm. It is an adaptation of the morphological analysis system developed by Oflazer [68].

FindStem is another stemming algorithm developed by **Sever and Bitirim** (2003) [70]. The algorithm consists of three steps: identifying the root, doing morphological analysis and identifying the stem. The method relies on a lexicon that contains the morphological and PoS features of words, and syntactic rules.

Dinçer and Karaoglan (2003) [71] introduce a probabilistic stemmer for a Turkish information retrieval system.

Eryigit and Adalı (2004) [72] propose a rule-based suffix stripping algorithm for Turkish similar to Porter stemmer.

Akın and Akın (2007) [73] introduce zemberek as a morphological analyzer and Çilden [74] introduces Snowball as a stemmer.

Özgür et al. [75] analyze the effects of stemming based on fixed-length word truncation and morphological analysis for multi-document summarization on Turkish. LexRank [76] summarization algorithm is used for the comparison. Results show that fixed-length word truncation methods improve the summarization scores, whereas morphological analysis does not improve summarization.

Özgür et al. [77] presents a language independent unsupervised stemmer for agglutinative languages. In the presence of a large enough training set, the algorithm performs stemming for an unseen word without a rule set or a separate lexicon.

Kışla and Karaoglan (2016) [66] present a hybrid method that is based on a simple idea that nouns and verbs have different suffix patterns. A statistical method is used to strip off the suffixes and based on the suffix pattern PoS tagging is determined which then enables the decision for the stem boundary.

3.5. Conclusion

In this chapter, we reviewed the previous work on unsupervised PoS tagging and stemming. We also presented the PoS tagging and stemming methods applied to Turkish as an agglutinative language. This background will serve as reference point for developing a joint PoS tagging and stemming presented in the next chapter.

4. MODEL

This chapter presents the proposed joint unsupervised PoS tagging and stemming models in this thesis.

4.1. Introduction

PoS tagging and stemming are closely interconnected tasks, which is already addressed in Chapter 1.. There have been many studies that perform the two tasks in an unsupervised framework. Most of these previous works have either presented pipeline approaches or hybrid approaches. We propose joint learning of PoS tagging and stemming in this thesis.

In this chapter, we describe our joint PoS tagging and stemming models. In order to learn both stems and PoS tags, we adopt the Bayesian HMM model of Goldwater and Griffiths [7], which is accepted as the baseline model. After the description of the baseline Bayesian PoS tagging model in Section 4.2., we will explain our models in Section 4.3..

4.2. Baseline Bayesian HMM Model

The Baseline Bayesian HMM model by Goldwater and Griffiths [7] extends the standard HMM model by adding prior distributions to the model parameters (i.e. transition and emission probability distributions). In this approach, for the prior distributions conjugate symmetric Dirichlet priors over Multinomial parameters are placed. The plate diagram of the model is given in Figure 4.1..

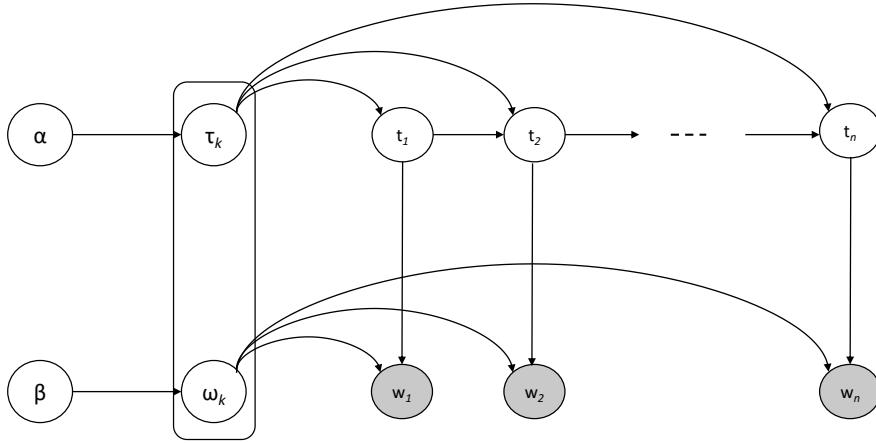


Figure 4.1. The plate diagram of the Bayesian HMM with symmetric Dirichlet priors.

The mathematical model is given as follows:

$$\begin{aligned}
 t_i | t_{i-1}, t_{i-2} = t', \tau^{(t,t')} &\propto \text{Mult}(\tau^{(t,t')}) \\
 w_i | t_i = t, \omega^{(t)} &\propto \text{Mult}(\omega^{(t)}) \\
 \tau^{(t,t')} | \alpha &\propto \text{Dirichlet}(\alpha) \\
 \omega^{(t)} | \beta &\propto \text{Dirichlet}(\beta)
 \end{aligned} \tag{17}$$

where w_i denotes the i th word and t_i is its tag. $\text{Mult}(\omega^t)$ is the emission distribution in the form of a Multinomial distribution with parameters $\omega^{(t)}$ that is generated by $\text{Dirichlet}(\beta)$ with hyperparameter β . Analogously, $\text{Mult}(\tau^{(t,t')})$ is the transition distribution with parameters $\tau^{(t,t')}$ that is generated by $\text{Dirichlet}(\alpha)$ with hyperparameter α .

Based on the mathematical model, the conditional probability of a tag and a word are defined as follows:

$$P(t_i | \mathbf{t}_{-i}, \alpha) = \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \tag{18}$$

$$P(w_i | \mathbf{t}_{-i}, \mathbf{w}_{-i}, \beta) = \frac{n_{(t_i, w_i)} + \beta}{n_{(t_i)} + W_{t_i}\beta} \tag{19}$$

where \mathbf{t}_{-i} is the current values of all tags except t_i , \mathbf{w}_{-i} represents the complete word list excluding w_i , W_{t_i} is the number of word types in the corpus, T is the size of the tag set, n_{t_i} is the number of words tagged with t_i , $n_{(t_i, w_i)}$ is the number of tag-word pair (t_i, w_i) , $n_{(t_{i-2}, t_{i-1})}$

is the frequency of the tag bigram $\langle t_{i-2}, t_{i-1} \rangle$ and $n_{(t_{i-2}, t_{i-1}, t_i)}$ is the frequency of the tag trigram $\langle t_{i-2}, t_{i-1}, t_i \rangle$.

Goldwater and Griffiths [7] use Gibbs sampling [25] to perform the inference. The inference involves estimating the posterior distribution:

$$P(\mathbf{t}|\mathbf{w}, \alpha, \beta) \propto P(\mathbf{w}|\mathbf{t}, \beta)P(\mathbf{t}|\alpha) \quad (20)$$

The sampling distribution of t_i under this model is:

$$\begin{aligned} P(t_i|\mathbf{t}_{-i}, \mathbf{w}_{-i}, \alpha, \beta) &= \frac{n_{(t_i, w_i)} + \beta}{n_{t_i} + W_{t_i}\beta} \cdot \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \\ &\cdot \frac{n_{(t_{i-1}, t_i, t_{i+1})} + I(t_{i-2}=t_{i-1}=t_i=t_{i+1}) + \alpha}{n_{(t_{i-1}, t_i)} + I(t_{i-2}=t_{i-1}=t_i) + T\alpha} \\ &\cdot \frac{n_{(t_i, t_{i+1}, t_{i+2})} + I(t_{i-2}=t_i=t_{i+2}, t_{i-1}=t_{i+1}) + I(t_{i-1}=t_i=t_{i+1}=t_{i+2}) + \alpha}{n_{(t_i, t_{i+1})} + I(t_{i-2}=t_i, t_{i-1}=t_{i+1}) + I(t_{i-1}=t_i=t_{i+1}) + T\alpha} \end{aligned} \quad (21)$$

where $n_{(t_{i-1}, t_i)}$ is the frequency of the tag bigram $\langle t_{i-1}, t_i \rangle$, $n_{(t_i, t_{i+1})}$ is the frequency of the tag bigram $\langle t_i, t_{i+1} \rangle$, $n_{(t_{i-1}, t_i, t_{i+1})}$ is the frequency of the tag trigrams $\langle t_{i-1}, t_i, t_{i+1} \rangle$, $n_{(t_i, t_{i+1}, t_{i+2})}$ is the frequency of tag trigram $\langle t_i, t_{i+1}, t_{i+2} \rangle$ and $I(\cdot)$ is an identity function that gives 1 if its argument is true, and otherwise 0. Sampling a tag affects three trigrams. Therefore, those changes are taken into account with the identity functions.

All tags are randomly initialized at the beginning of the inference. Then each word's tag is sampled from the tags's posterior distribution given in Equation 21. This process is repeated until the system converges.

4.3. Joint Models for PoS Tagging and Stemming

We extend the baseline model that is explained in the previous section to perform joint PoS tagging and stemming in a joint model. To this end, we propose different extensions to the same model.

4.3.1. Stem-based Bayesian HMM (Bayesian S-HMM)

Most of the statistical stemming algorithms use the method of stripping suffixes from the word end without considering the syntactic similarity of the word and its stem. Inflectional affixation retains the PoS tag of the word, whereas derivational affixation may not. For instance, if *playing* is a noun, then stripping of suffix *-ing* is a stemming error, if *playing* is a verb, then removing suffix *-ing* will be correct. Using stem emissions instead of word emissions will reduce the emission sparsity, thereby mitigate the number of the OOV words. Thus, we propose to emit stems rather than words in the baseline model. The plate diagram of the model is given in Figure 4.2..

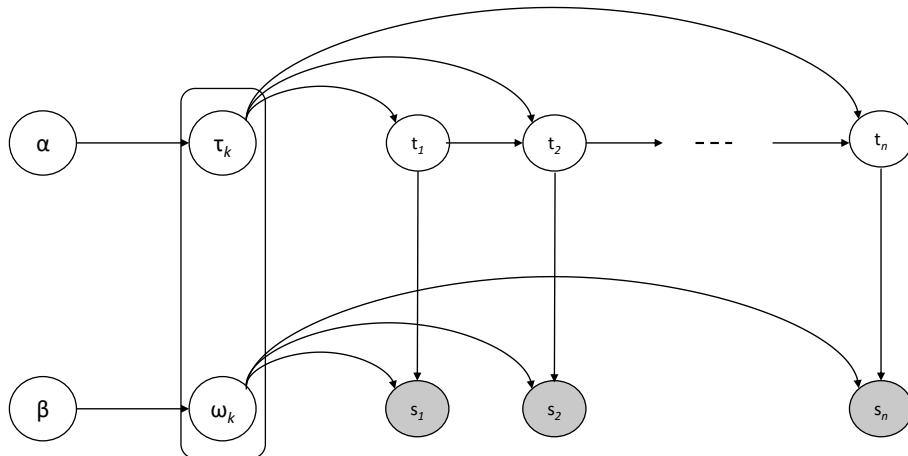


Figure 4.2. The plate diagram of the stem based Bayesian HMM.

The mathematical model is given as follows:

$$\begin{aligned}
 t_i | t_{i-1}, t_{i-2} = t' &\propto Mult(\tau^{(t,t')}) \\
 s_i | t_i = t, \omega^{(t)} &\propto Mult(\omega^{(t)}) \\
 \tau^{(t,t')} | \alpha &\propto Dirichlet(\alpha) \\
 \omega^{(t)} | \beta &\propto Dirichlet(\beta)
 \end{aligned} \tag{22}$$

Here, t_i and s_i are the i th tag and stem, where $w_i = s_i + m_i$, m_i being the suffix of w_i . $Mult(\omega^t)$ is the emission distribution in the form of a Multinomial distribution with parameters $\omega^{(t)}$ that is generated by $Dirichlet(\beta)$ with hyperparameter β . Analogously,

$Mult(\tau^{(t,t')})$ is the transition distribution with parameters $\tau^{(t,t')}$ that is generated by $Dirichlet(\alpha)$ with hyperparameter α .

Based on the mathematical model, the conditional probability of a tag and a stem are defined as follows:

$$P(t_i|\mathbf{t}_{-i}, \alpha) = \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \quad (23)$$

$$P(s_i|\mathbf{t}_{-i}, \mathbf{s}_{-i}, \beta) = \frac{n_{(t_i, s_i)} + \beta}{n_{(t_i)} + S_{t_i}\beta} \quad (24)$$

where \mathbf{s}_{-i} refers to stem set excluding the current stem s_i , S_{t_i} is the number of stem types in the corpus, T is the size of the tag set, n_{t_i} is the number of stems tagged with t_i , $n_{(t_i, s_i)}$ is the number of tag-stem pair (t_i, s_i) .

The inference involves estimating the following posterior distribution:

$$P(\mathbf{t}, \mathbf{s}|\alpha, \beta) \propto P(\mathbf{s}|\mathbf{t}, \beta)P(\mathbf{t}|\alpha) \quad (25)$$

The sampling distribution for t_i and s_i under this model is:

$$\begin{aligned} P(t_i, s_i | \mathbf{t}_{-i}, \mathbf{s}_{-i}, \alpha, \beta) &= \frac{n_{(t_i, s_i)} + \beta}{n_{t_i} + S_{t_i}\beta} \cdot \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \\ &\cdot \frac{n_{(t_{i-1}, t_i, t_{i+1})} + I(t_{i-2}=t_{i-1}=t_i=t_{i+1}) + \alpha}{n_{(t_{i-1}, t_i)} + I(t_{i-2}=t_{i-1}=t_i) + T\alpha} \\ &\cdot \frac{n_{(t_i, t_{i+1}, t_{i+2})} + I(t_{i-2}=t_i=t_{i+2}, t_{i-1}=t_{i+1}) + I(t_{i-1}=t_i=t_{i+1}=t_{i+2}) + \alpha}{n_{(t_i, t_{i+1})} + I(t_{i-2}=t_i, t_{i-1}=t_{i+1}) + I(t_{i-1}=t_i=t_{i+1}) + T\alpha} \end{aligned} \quad (26)$$

Algorithm of inference is given in Algorithm 1. All tags are randomly initialized and all words are split into two segments randomly as a stem and a suffix at the beginning of the inference. In each iteration of the algorithm, a tag and a stem are sampled for each word from the posterior distribution given in Equation 26 by using Gibbs sampling. This process is repeated until the system converges.

Algorithm 1: Stem-based Bayesian HMM

Input: $W, \alpha, \beta, \gamma, \delta, T, \text{iterasyon}$ **Output:** Tagged and stemmed corpus**for** w in W **do** $i \sim \text{uniform}(1, \text{length}(w)) s \leftarrow w[1 : i]$
 $t \sim \text{uniform}(1, T)$ **for** $k \leftarrow 1$ **to** iterasyon **do** **for** w in W **do**
 $t_i, s_i \leftarrow P(t_i, s_i | t_{-i}, s_{-i}, \alpha, \beta)$ choose new label and stem**return** W

4.3.2. Stem & Suffix-based Bayesian HMM (Bayesian SM-HMM)

In this model, we are inspired by the morphological similarity of words having the same PoS tag. Words belonging to the same syntactic category usually take similar suffixes. For example, words ending with *ly* are usually adverbs, whereas words ending with *ness* are usually nouns. We include suffixes in the emissions in addition to the stems as seen in the plate diagram of the model given in Figure 4.3..

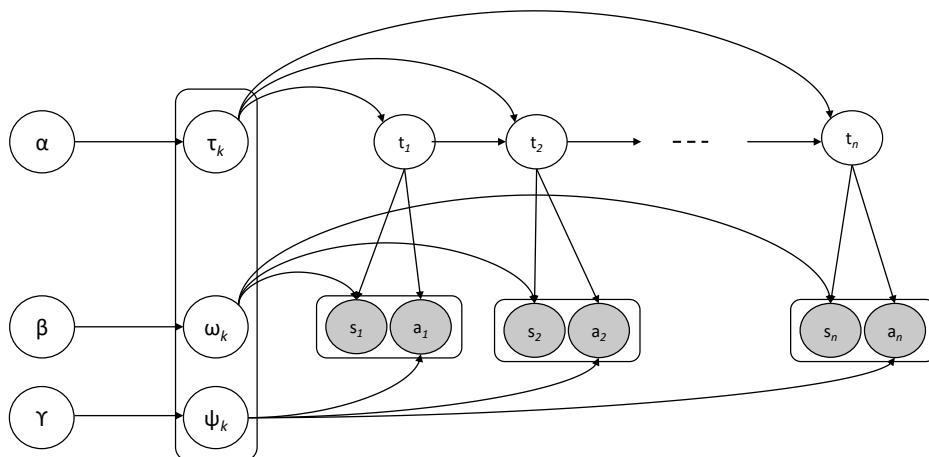


Figure 4.3. The plate diagram of the stem and suffix-based Bayesian HMM.

The extended mathematical model becomes as follows:

$$\begin{aligned}
t_i | t_{i-1}, t_{i-2} = t' , \tau^{(t,t')} &\propto Mult(\tau^{(t,t')}) \\
s_i | t_i = t, \omega^{(t)} &\propto Mult(\omega^{(t)}) \\
m_i | t_i = t, \psi^{(t)} &\propto Mult(\psi^{(t)}) \\
\tau^{(t,t')} | \alpha &\propto Dirichlet(\alpha) \\
\omega^t | \beta &\propto Dirichlet(\beta) \\
\psi^{(t)} | \gamma &\propto Dirichlet(\gamma)
\end{aligned} \tag{27}$$

Here, t_i , s_i and m_i are the i th tag, the stem and the suffix where $w_i = s_i + m_i$. $Mult(\omega^t)$ is the stem emission distribution in the form of a Multinomial distribution with parameters $\omega^{(t)}$ that is generated by $Dirichlet(\beta)$ with hyperparameter β and $Mult(\psi^{(t)})$ is the suffix emission distribution in the form of a Multinomial distribution with parameters $\psi^{(t)}$ that is generated by $Dirichlet(\gamma)$ with hyperparameter γ . Analogously, $Mult(\tau^{(t,t')})$ is the transition distribution with parameters $\tau^{(t,t')}$ that is generated by $Dirichlet(\alpha)$ with hyperparameter α .

Based on the mathematical model, the conditional probability of a tag, a stem and a suffix are defined respectively as follows:

$$P(t_i | \mathbf{t}_{-i}, \alpha) = \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \tag{28}$$

$$P(s_i | \mathbf{t}_{-i}, \mathbf{s}_{-i}, \beta) = \frac{n_{(t_i, s_i)} + \beta}{n_{(t_i)} + S_{t_i}\beta} \tag{29}$$

$$P(m_i | \mathbf{t}_{-i}, \mathbf{m}_{-i}, \gamma) = \frac{n_{(t_i, m_i)} + \gamma}{n_{(t_i)} + M_{t_i}\gamma} \tag{30}$$

where \mathbf{m}_{-i} denotes the suffix of all suffixes except m_i , M_{t_i} is the number of suffix types in the corpus, n_{t_i} is the number of stems tagged with t_i , $n_{(t_i, s_i)}$ is the number of tag-stem pairs (t_i, s_i) , $n_{(t_i, m_i)}$ is the number of tag-suffix pairs.

The inference involves estimating the following posterior distribution:

$$P(\mathbf{t}, \mathbf{s}, \mathbf{m} | \alpha, \beta, \gamma) \propto P(\mathbf{s} | \mathbf{t}, \beta) P(\mathbf{m} | \mathbf{t}, \gamma) P(\mathbf{t} | \alpha) \tag{31}$$

The new posterior distribution of t_i , s_i and m_i under this model is given as follows:

$$\begin{aligned}
P(t_i, s_i, m_i | \mathbf{t}_{-i}, \mathbf{s}_{-i}, \mathbf{m}_{-i}, \alpha, \beta, \gamma) &= \frac{n_{(t_i, s_i)} \beta}{n_{t_i} + S_{t_i} \beta} \cdot \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \\
&\cdot \frac{n_{(t_{i-1}, t_i, t_{i+1})} + I(t_{i-2} = t_{i-1} = t_i = t_{i+1}) + \alpha}{n_{(t_{i-1}, t_i)} + I(t_{i-2} = t_{i-1} = t_i) + T\alpha} \\
&\cdot \frac{n_{(t_i, t_{i+1}, t_{i+2})} + I(t_{i-2} = t_i = t_{i+2}, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1} = t_{i+2}) + \alpha}{n_{(t_i, t_{i+1})} + I(t_{i-2} = t_i, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1}) + T\alpha} \\
&\cdot \frac{n_{(t_i, m_i)} + \gamma}{n_{t_i} + M_{t_i} \gamma}
\end{aligned} \tag{32}$$

Here, we assume that stems and suffixes are independent from each other. For the inference, all tags are randomly initialized and all words are split into two segments randomly. In each iteration of the algorithm, a tag, a stem and a suffix are sampled for each word from the posterior distribution given in Equation 32.

4.3.3. Stem-based Bayesian HMM using Neural Word Embeddings (Bayesian CS-HMM)

Inflectional affixation preserves the meaning of the word in addition to its syntactic category. Thus, we add semantic features to the model as prior information and we use neural word embeddings obtained from word2vec [78].

The mathematical model is the same as the stem-based Bayesian HMM model given in Section 4.3.1..

The posterior distribution of t_i and s_i under this model is:

$$\begin{aligned}
P(t_i, s_i | \mathbf{t}_{-i}, \mathbf{s}_{-i}, \alpha, \beta) &= \frac{n_{(t_i, s_i)} + \beta}{n_{t_i} + S_{t_i} \beta} \cdot \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \\
&\cdot \frac{n_{(t_{i-1}, t_i, t_{i+1})} + I(t_{i-2} = t_{i-1} = t_i = t_{i+1}) + \alpha}{n_{(t_{i-1}, t_i)} + I(t_{i-2} = t_{i-1} = t_i) + T\alpha} \\
&\cdot \frac{n_{(t_i, t_{i+1}, t_{i+2})} + I(t_{i-2} = t_i = t_{i+2}, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1} = t_{i+2}) + \alpha}{n_{(t_i, t_{i+1})} + I(t_{i-2} = t_i, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1}) + T\alpha} \\
&\cdot \cos(s_i, w_i)
\end{aligned} \tag{33}$$

where $\cos(s_i, w_i)$ is the cosine similarity of the word vectors of s_i and w_i . The higher the cosine similarity is, semantically closer to the words are.

4.3.4. Stem & Suffix-based Bayesian HMM using Neural Word Embeddings (Bayesian CSM-HMM)

In this model, a stem-suffix pair is emitted from each HMM state analogously to the stem-suffix-based Bayesian HMM model. Additionally, we use the semantic information obtained from neural word embeddings. Therefore, the mathematical model is the same as the stem-suffix-based Bayesian HMM model given in Section 4.3.2..

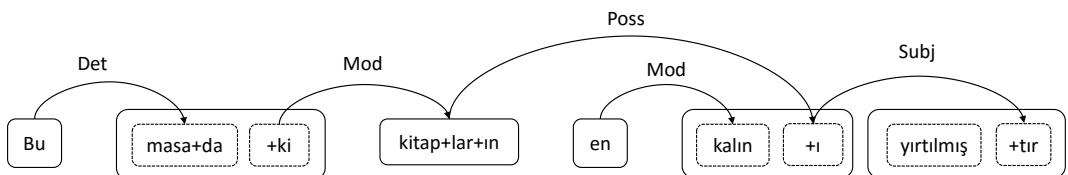
The new conditional distribution of t_i , s_i and m_i becomes:

$$\begin{aligned}
 P(t_i, s_i, m_i | \mathbf{t}_{-i}, \mathbf{s}_{-i}, \mathbf{m}_{-i}, \alpha, \beta, \gamma) &= \frac{n_{(t_i, s_i)} + \beta}{n_{t_i} + S_{t_i} \beta} \cdot \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \\
 &\cdot \frac{n_{(t_{i-1}, t_i, t_{i+1})} + I(t_{i-2} = t_{i-1} = t_i = t_{i+1}) + \alpha}{n_{(t_{i-1}, t_i)} + I(t_{i-2} = t_{i-1} = t_i) + T\alpha} \\
 &\cdot \frac{n_{(t_i, t_{i+1}, t_{i+2})} + I(t_{i-2} = t_i = t_{i+2}, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1} = t_{i+2}) + \alpha}{n_{(t_i, t_{i+1})} + I(t_{i-2} = t_i, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1}) + T\alpha} \\
 &\cdot \frac{n_{(t_i, m_i)} + \gamma}{n_{t_i} + M_{t_i} \gamma} \cdot \cos(s_i, w_i)
 \end{aligned} \tag{34}$$

Again each stem and suffix are assumed to be independent from each other.

4.3.5. Affix Transition-based Bayesian HMM Model (Bayesian A-HMM)

In the previous models, all suffixes are assumed to be independent. However, there is a dependency between the suffixes of each word in the same sentence, especially in agglutinative languages [79]. For example, we see dependency of suffixes of each word in a sentence in Figure 4.4..



Bu masadaki kitapların en kalını yırtılmıştır. (The thickest of books on this table is torn.)

Figure 4.4. Dependency of suffixes in an example Turkish sentence.

The plate diagram of the model is given in Figure 4.5..

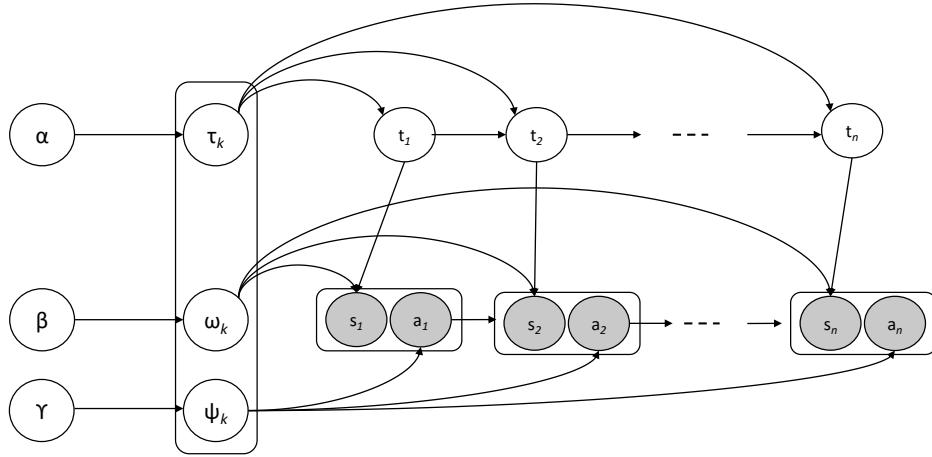


Figure 4.5. The plate diagram of the affix transition-based Bayesian HMM.

We adopt a trigram model for the final suffixes of each successive triples of words in each sentence. The mathematical model is given as follows:

$$\begin{aligned}
 t_i | t_{i-1}, t_{i-2} = t' &\propto Mult(\tau^{(t,t')}) \\
 s_i | t_i = t, \omega^{(t)} &\propto Mult(\omega^{(t)}) \\
 m_i | m_{i-1}, m_{i-2} = m' &\propto Mult(\psi^{(m,m')}) \\
 \tau^{(t,t')} | \alpha &\propto Dirichlet(\alpha) \\
 \omega^t | \beta &\propto Dirichlet(\beta) \\
 \psi^{(m,m')} | \gamma &\propto Dirichlet(\gamma)
 \end{aligned} \tag{35}$$

Here, $Mult(\psi^{(m,m')})$ defines the trigram model for the final suffixes of words with parameters $\psi^{(m,m')}$ that is generated by $Dirichlet(\gamma)$ with hyperparameter γ .

Based on the mathematical model, the conditional probability of a tag, a stem and a suffix are defined as follows:

$$P(t_i | \mathbf{t}_{-i}, \alpha) = \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \tag{36}$$

$$P(m_i | \mathbf{m}_{-i}, \gamma) = \frac{n_{(m_{i-2}, m_{i-1}, t_i)} + \gamma}{n_{(m_{i-2}, m_{i-1})} + A\gamma} \tag{37}$$

$$P(s_i | \mathbf{t}_{-i}, \mathbf{s}_{-i}, \beta) = \frac{n_{(t_i, s_i)} + \beta}{n_{(t_i)} + S_{t_i}\beta} \tag{38}$$

where A is the number of unique suffix types in the corpus, $n_{(m_{i-2}, m_{i-1})}$ is the frequency of the suffix bigram $\langle m_{i-2}, m_{i-1} \rangle$ and $n_{(m_{i-2}, m_{i-1}, m_i)}$ is the frequency of the suffix trigram $\langle m_{i-2}, m_{i-1}, m_i \rangle$.

The inference involves estimating the following posterior distribution:

$$P(\mathbf{t}, \mathbf{s}, \mathbf{m} | \alpha, \beta, \gamma) \propto P(\mathbf{s} | \mathbf{t}, \beta) P(\mathbf{m} | \gamma) P(\mathbf{t} | \alpha) \quad (39)$$

Here we again assume that each stem and suffix of a word are independent from each other. We generate the suffixes independently from the tags.

The new posterior distribution of t_i , s_i and m_i becomes:

$$\begin{aligned} P(t_i, s_i, m_i | \mathbf{t}_{-i}, \mathbf{s}_{-i}, \mathbf{m}_{-i}, \alpha, \beta, \gamma) &= \frac{n_{(t_i, s_i)} + \beta}{n_{t_i} + S_{t_i} \beta} \cdot \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \\ &\cdot \frac{n_{(t_{i-1}, t_i, t_{i+1})} + I(t_{i-2} = t_{i-1} = t_i = t_{i+1}) + \alpha}{n_{(t_{i-1}, t_i)} + I(t_{i-2} = t_{i-1} = t_i) + T\alpha} \\ &\cdot \frac{n_{(t_i, t_{i+1}, t_{i+2})} + I(t_{i-2} = t_i = t_{i+2}, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1} = t_{i+2}) + \alpha}{n_{(t_i, t_{i+1})} + I(t_{i-2} = t_i, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1}) + T\alpha} \\ &\cdot \frac{n_{(m_{i-2}, m_{i-1}, m_i)} + \gamma}{n_{(m_{i-2}, m_{i-1})} + A\gamma} \\ &\cdot \frac{n_{(m_{i-1}, m_i, m_{i+1})} + I(m_{i-2} = m_{i-1} = m_i = m_{i+1}) + \gamma}{n_{(m_{i-1}, m_i)} + I(m_{i-2} = m_{i-1} = m_i) + A\gamma} \\ &\cdot \frac{n_{(m_i, m_{i+1}, m_{i+2})} + I(m_{i-2} = m_i = m_{i+2}, m_{i-1} = m_{i+1}) + I(m_{i-1} = m_i = m_{i+1} = m_{i+2}) + \gamma}{n_{(m_i, m_{i+1})} + I(m_{i-2} = m_i, m_{i-1} = m_{i+1}) + I(m_{i-1} = m_i = m_{i+1}) + A\gamma} \end{aligned} \quad (40)$$

where $n_{(m_{i-1}, m_i)}$ is the frequency of the suffix bigram $\langle m_{i-1}, m_i \rangle$, $n_{(m_i, m_{i+1})}$ is the frequency of the suffix bigram $\langle m_i, m_{i+1} \rangle$, $n_{(m_{i-1}, m_i, m_{i+1})}$ is the frequency of the suffix trigram $\langle m_{i-1}, m_i, m_{i+1} \rangle$ and $n_{(m_i, m_{i+1}, m_{i+2})}$ is the frequency of the suffix trigram $\langle m_i, m_{i+1}, m_{i+2} \rangle$. Sampling each suffix affects three trigrams since each suffix exists in three suffix trigrams. Therefore, we consider the three affected trigrams in sampling using identity function analogously to the tags.

4.3.6. Stem & Affix Transition-based Bayesian HMM Model (Bayesian AS-HMM)

In this model, only suffixes are emitted from each HMM state and stems are independent. The plate diagram of the model is given in Figure 4.6..

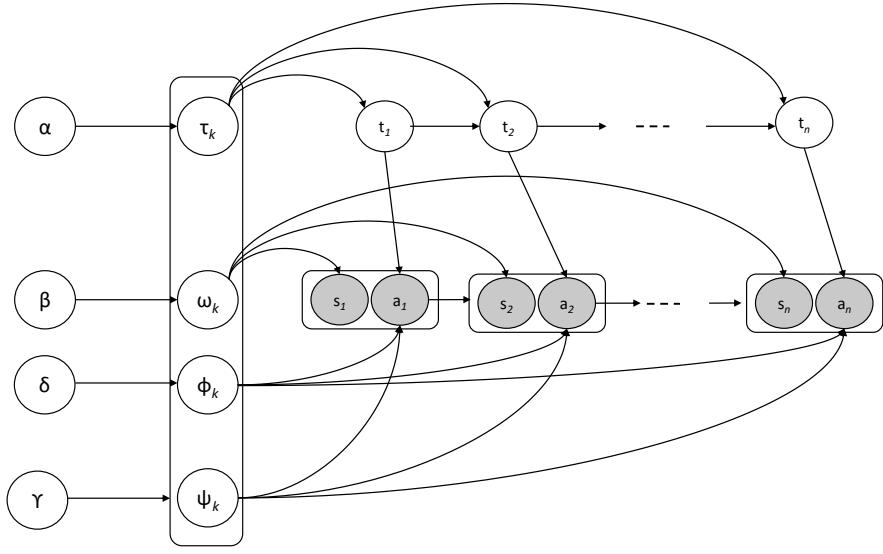


Figure 4.6. Stem and Transition based Bayesian HMM.

The mathematical model is given as follows:

$$\begin{aligned}
 t_i | t_{i-1}, t_{i-2} = t', \tau^{(t,t')} &\propto \text{Mult}(\tau^{(t,t')}) \\
 s_i, \rho &\propto \text{Mult}(\rho) \\
 m_i | t_i = t, \phi^{(t)} &\propto \text{Mult}(\phi^{(t)}) \\
 m_i | m_{i-1}, m_{i-2} = m', \psi^{(m,m')} &\propto \text{Mult}(\psi^{(m,m')}) \\
 \tau^{(t,t')} | \alpha &\propto \text{Dirichlet}(\alpha) \\
 \rho | \beta &\propto \text{Dirichlet}(\beta) \\
 \phi^t | \delta &\propto \text{Dirichlet}(\delta) \\
 \psi^{(m,m')} | \gamma &\propto \text{Dirichlet}(\gamma)
 \end{aligned} \tag{41}$$

where $\text{Mult}(\phi^t)$ is the emission distribution in the form of a Multinomial distribution with parameter $\phi^{(t)}$ that is generated by $\text{Dirichlet}(\delta)$ with hyperparameter δ and $\text{Mult}(\rho)$ defines the model for stems of words with parameters ρ that is generated by $\text{Dirichlet}(\beta)$ with hyperparameters β .

Based on the mathematical model, the conditional probability of a tag, a stem and a suffix are defined as follows:

$$P(s_i|\beta) = \frac{n(s_i) + \beta}{S + S_c\beta} \quad (42)$$

$$P(t_i|\mathbf{t}_{-i}, \alpha) = \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \quad (43)$$

$$P(m_i|\mathbf{m}_{-i}, \gamma) = \frac{n_{(m_{i-2}, m_{i-1}, t_i)} + \gamma}{n_{(m_{i-2}, m_{i-1})} + A\gamma} \quad (44)$$

$$P(m_i|\mathbf{t}_{-i}, \mathbf{m}_{-i}, \delta) = \frac{n_{(t_i, m_i)} + \delta}{n_{(t_i)} + M_{t_i}\delta} \quad (45)$$

where $n(s_i)$ defines the number of words stemmed as s_i , S is the number of stem in the corpus, and S_c is the number of unique stem types in the corpus.

Inference involves estimating the posterior distribution:

$$P(\mathbf{t}, \mathbf{s}, \mathbf{m}|\alpha, \beta, \gamma, \delta) \propto P(\mathbf{s}|\beta)P(\mathbf{m}|\mathbf{t}, \delta)P(\mathbf{t}|\alpha)P(\mathbf{m}|\gamma) \quad (46)$$

The posterior distribution of t_i , s_i and m_i under this model is:

$$\begin{aligned} P(t_i, s_i, m_i | \mathbf{t}_{-i}, \mathbf{s}_{-i}, \mathbf{m}_{-i}, \alpha, \beta, \gamma, \delta) &= \frac{n_{(t_i, m_i)} + \delta}{n_{t_i} + M_{t_i}\delta} \cdot \frac{n(s_i)\beta}{S + S_c\beta} \\ &\cdot \frac{n_{(t_{i-2}, t_{i-1}, t_i)} + \alpha}{n_{(t_{i-2}, t_{i-1})} + T\alpha} \\ &\cdot \frac{n_{(t_{i-1}, t_i, t_{i+1})} + I(t_{i-2} = t_{i-1} = t_i = t_{i+1}) + \alpha}{n_{(t_{i-1}, t_i)} + I(t_{i-2} = t_{i-1} = t_i) + T\alpha} \\ &\cdot \frac{n_{(t_i, t_{i+1}, t_{i+2})} + I(t_{i-2} = t_i = t_{i+2}, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1} = t_{i+2}) + \alpha}{n_{(t_i, t_{i+1})} + I(t_{i-2} = t_i, t_{i-1} = t_{i+1}) + I(t_{i-1} = t_i = t_{i+1}) + T\alpha} \\ &\cdot \frac{n_{(m_{i-2}, m_{i-1}, m_i)} + \gamma}{n_{(m_{i-2}, m_{i-1})} + A\gamma} \\ &\cdot \frac{n_{(m_{i-1}, m_i, m_{i+1})} + I(m_{i-2} = m_{i-1} = m_i = m_{i+1}) + \gamma}{n_{(m_{i-1}, m_i)} + I(m_{i-2} = m_{i-1} = m_i) + A\gamma} \\ &\cdot \frac{n_{(m_i, m_{i+1}, m_{i+2})} + I(m_{i-2} = m_i = m_{i+2}, m_{i-1} = m_{i+1}) + I(m_{i-1} = m_i = m_{i+1} = m_{i+2}) + \gamma}{n_{(m_i, m_{i+1})} + I(m_{i-2} = m_i, m_{i-1} = m_{i+1}) + I(m_{i-1} = m_i = m_{i+1}) + A\gamma} \end{aligned} \quad (47)$$

5. EXPERIMENTS AND RESULTS

In this chapter, we describe the datasets used in our experiments in Section 5.1. and evaluation metrics used in experiments in Section 5.2.. In Section 5.3., we present the results and compare our model’s performance for the PoS tagging and the stemming tasks to the other PoS tagging and stemming models.

5.1. Datasets

We ran experiments on several languages. The datasets used in the experiments are:

METU-Treebank [80] is a Turkish treebank built from newspapers, journal issues and books. The treebank involves 5620 sentences and 53,798 tokens.

Penn Treebank [81] is an English treebank collected from the Air Traffic Information System, the Wall Street Journal (WSJ), the Brown Corpus, Switchboard, and a variety of other sources. We used the first 12K and 24K words from the corpus for the experiments.

Finn Treebank [82] is a Finnish dataset annotated manually.

UD Dependency Treebank [83] is a cross-lingual treebank built especially for multilingual parsing and cross-lingual learning. We used only Basque, English and Hungarian portions.

Table 5.1. Datasets used in the experiments

Language	Source	# Tags
Basque	UD Dependency Treebank [83]	16
English	Penn Treebank [81]	45
English	UD Dependency Treebank [83]	17
Finnish	FinnTreeBank [82]	14
Hungarian	UD Dependency Treebank [83]	16
Turkish	METU Treebank [80]	31

All datasets contain different tagsets and this variety aggravates evaluating the results of PoS tagging. We use universal PoS tagset defined by **Petrov et al. (2011)** [2] in our experiments.

This set consists of 12 coarse-grained tags. Therefore, we reduce the size of the tagset to 12 based on the universal PoS tagset (see Appendix A).

Figure 5.1. gives an example of mapping for a sentence taken from Penn Treebank [81]. Original PoS tagset is Penn Treebank tagset [81] and universal PoS tagset is taken from universal PoS tagset defined by Petrov et al. (2011) [2].

Example

Sentence :	It	has	ho	hearing	on	our	work	force	today	.
Original :	It/PRP	has/VBZ	no/DT	bearing/NN	on/IN	our/PRP	work/NN	force/NN	today/NN	.
Universal :	It/PRON	has/VERB	no/DET	bearing/NOUN	on/ADP	our/PRON	work/NOUN	force/NOUN	today/NOUN	.

Figure 5.1. Example sentence with its specific and corresponding universal POS tags.

5.2. Evaluation Metrics

In this section, we briefly explain the evaluation metrics that we used for PoS taging and stemming.

5.2.1. PoS Tagging

Many-to-one

Many-to-one [23] accuracy maps each result tag to the most frequent gold standard tag. In this method, more than one cluster can be mapped to the same gold tag. It is one of the commonly used metrics in the literature.

One-to-one

One-to-one [31] accuracy maps each result tag to a single gold standard tag. This method uses a greedy algorithm to obtain the best relevant mapping in terms of accuracy.

Variation of Information (VI)

VI [84] is one of the most common approaches to evaluate PoS tagging. VI uses two parameters; homogeneity and completeness to measure the amount of information that changes from clustering C to clustering G. The expression of VI is given as follows:

$$\begin{aligned} VI(C_r, C_g) &= H(C_r) + H(C_g) - 2I(C_r, C_g) \\ &= H(C_r|C_g) + H(C_g|C_r) \end{aligned} \quad (48)$$

where $I(C_r, C_g)$ measures the mutual dependence between two clusterings, $H(C_r)$ is the entropy associated with the result clustering C_r (result clustering) and $H(C_g)$ is the entropy associated with the gold clustering C_g (the gold clustering).

Normalized Mutual Information (NMI)

NMI [85] normalizes the symmetric measure of statistical information between two distributions [86]. The formula of NMI is given as follows:

$$NMI(C_r, C_g) = \frac{I(C_r, C_g)}{\sqrt{H(C_r)H(C_g)}} \quad (49)$$

where $I(C_r, C_g)$ measures the mutual dependence between two clusterings, $H(C_r)$ is the entropy associated with the result clustering C_r , $H(C_g)$ is the entropy associated with the gold clustering C_g .

5.2.2. Stemming

Stemmer Strength Metrics

Strength of a stemmer is important due to the prediction of Recall and Precision of index compression in the character removal case. A strong stemmer indicates a higher Recall, index compression and a lower Precision.

We used five metrics to compare the strength of our models:

1. *The mean number of words per conflation class (MWC)* : It is the average number of words that correspond to the same stem. Stronger stemmers tend to have a higher

MWC. It is computed as given below:

$$MWC = \frac{N}{S} \quad (50)$$

where S is the number of unique stems once the stemming is performed and N is total number of unique words in the corpus.

2. *Index compression factor (ICF)* : It measures the decrease of the size of the corpus as a result of stemming. This can be calculated by;

$$ICF = \frac{N - S}{N} \quad (51)$$

where N is the number of unique words before stemming and N is the number of unique words after stemming. Stronger stemmers tend to have a higher ICF.

3. *The number of words and stems that differ (NWSF)* : It is the difference between the number of words before and after stemming. It indicates the strength of stemming because stronger stemmers tend to transform words more than weaker stemmers.
4. *The mean number of characters removed (MCRS)* : It counts the number of characters that are removed by the stemmer. Stronger stemmers tend to remove more characters from words to obtain stems compared to weaker stemmers.
5. *The mean and median Modified Hamming distance (MHD)* : It measures the distance between words and their stems. The modified Hamming distance is calculated by adding the Hamming distance to the difference in the length between the word and its stem.

Accuracy

Accuracy is used to measure the correctness of stems obtained at the end of stemming.

Frakes and Fox Similarity Metric (FSM)

This metric is proposed by Frakes and Fox [87] to evaluate the strength and similarity based on the Hamming distance measure. It is calculated by;

$$N_a = \frac{N_w}{N_s} \quad (52)$$

N_a is the average number of words per conflation class, N_w refers to the number of unique words before stemming, and N_s is the number of unique stems after stemming.

5.3. Experiments

We ran experiments for Turkish, English, Finnish, Hungarian and Basque. As defined in Chapter 4., our models have four hyperparameters: α , β , γ and δ . We manually set these hyperparameters for each experiment. We defined six sets for the values of the hyperparameters.

1. set $\alpha=0.003 \beta=1 \gamma=0.003 \delta=0.003$
2. set $\alpha=0.003 \beta=0.1 \gamma=0.003 \delta=0.003$
3. set $\alpha=0.001 \beta=1 \gamma=0.001 \delta=0.001$
4. set $\alpha=0.001 \beta=0.1 \gamma=0.001 \delta=0.001$
5. set $\alpha=0.03 \beta=1 \gamma=0.03 \delta=0.03$
6. set $\alpha=0.03 \beta=0.1 \gamma=0.03 \delta=0.03$

The evaluation scores for each set will be given separately in order to understand the efforts of hyperparameters better. For each experiment, we performed 5000 iterations in Gibbs sampling.

We implemented our models on Python 3.5, and tested it on a server with Express x3650 M5, Xeon 6C E5-2620v3 2.4GHz/1866MHz/15MB Intel Xeon 2.40GHz processor. Training Bayesian S-HMM and Bayesian SM-HMM models on 53K dataset takes about 15h, 24K datasets about 8h, and 12K datasets about 6h. Bayesian SM-HMM. Training Bayesian CS-HMM and Bayesian CSM-HMM models takes longer about 2-3h on datasets due to access neural word embeddings obtained from word2vec [78] and finally Bayesian A-HMM and Bayesian AS-HMM models on 53K dataset takes about 24h, 24K datasets about 18h, and 12K datasets about 12h.

We compare our PoS tagging results with Brown Clustering¹ [14] and Anchor HMM² [29] and our stemming results with HPS³ [45], Morfessor FlatCat⁴ [88], and Linguistica⁵ [53].

¹Brown Clustering: <http://www.cs.berkeley.edu/~pliang/software/brown-cluster-1.2.zip>(Percy Liang)

²Anchor HMM: <https://github.com/karlstratos/anchor>

³HPS: <http://liks.fav.zcu.cz/HPS/>

⁴Morfessor FlatCat: <https://github.com/aalto-speech/flatcat>

⁵Linguistica: <http://linguistica.uchicago.edu/>

Table 5.2. shows the PoS tagging results for Turkish for six hyperparameter sets. For PoS tagging task, in Many-to-one accuracy, Bayesian S-HMM gives better results and in other metrics, Bayesian CSM-HMM model gives better results than the other models. When we look at the results, generally results of Bayesian CSM-HMM are better for PoS tagging. This shows that using suffixes and semantic features helps in PoS tagging. The overall PoS tagging results show that Bayesian CSM-HMM outperforms both Brown Clustering [14], word-based Bayesian HMM [7](baseline model), and Anchor HMM [29] for Turkish according to both Many-to-one, One-to-one, NMI and VI measure.

Since Anchor HMM [29] evaluation is restricted, One-to-one, NMI and VI accuracy couldn't be computed.

In terms of the hyperparameter values, the results show that the sixth hyperparameter set gives better results than the others (see Figure 5.2.).

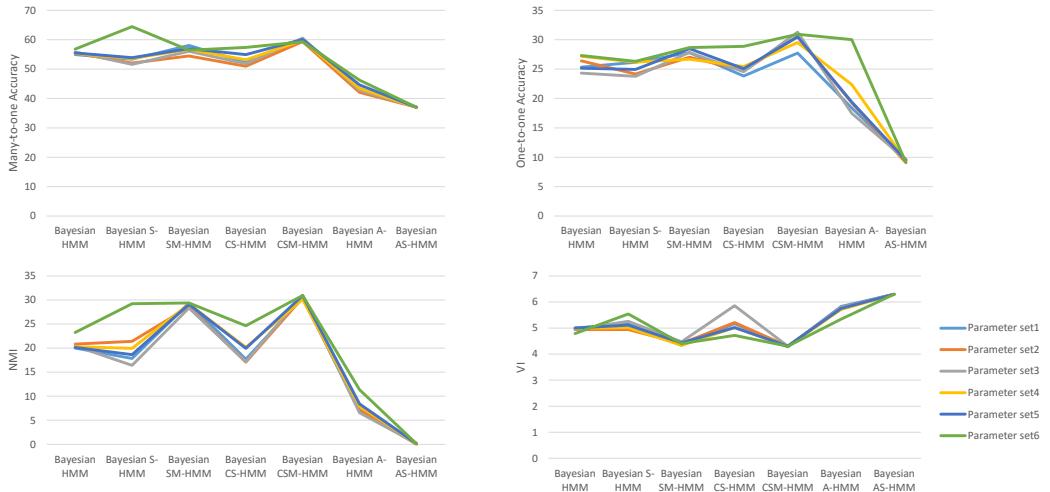


Figure 5.2. Sensitivity of hyperparameter sets for PoS tagging performance in Turkish

The stemming results for Turkish are given in Table 5.3.. The results show that Bayesian CS-HMM model generally gives better results in among the six hyperparameter sets. Our results are far better than HPS [45], Morfessor FlatCat [88], and Linguistica [53] for Turkish. Adding neural word embeddings to the model made a significant improvement. This may lead this conclusion:semantic information about stem and word has an important impact on stemming.

Table 5.2. Turkish PoS tagging results for different hyperparameter sets

		Metu-Sabancı Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	54.92	25.32	19.99	5.00
	Bayesian S-HMM	53.36	26.15	17.84	5.16
	Bayesian SM-HMM	58.07	27.80	29.35	4.34
	Bayesian CS-HMM	52.27	23.81	17.67	5.17
	Bayesian CSM-HMM	60.09	27.73	30.43	4.28
	Bayesian A-HMM	43.26	18.38	6.99	5.83
	Bayesian AS-HMM	37.03	9.64	0.13	6.30
2	Bayesian HMM	55.44	26.40	20.83	4.94
	Bayesian S-HMM	52.04	24.17	21.41	4.94
	Bayesian SM-HMM	56.46	27.02	28.55	4.41
	Bayesian CS-HMM	50.97	24.75	17.09	5.21
	Bayesian CSM-HMM	59.36	30.53	30.49	4.30
	Bayesian A-HMM	42.05	19.26	7.23	5.78
	Bayesian AS-HMM	37.03	9.06	0.09	6.30
3	Bayesian HMM	55.81	24.32	20.46	4.95
	Bayesian S-HMM	51.58	23.77	16.41	5.26
	Bayesian SM-HMM	56.05	27.83	28.33	4.47
	Bayesian CS-HMM	51.81	24.55	17.17	5.86
	Bayesian CSM-HMM	60.53	31.28	30.97	4.30
	Bayesian A-HMM	42.73	17.48	6.54	5.78
	Bayesian AS-HMM	37.03	9.60	0.16	6.30
4	Bayesian HMM	55.31	27.18	20.30	4.99
	Bayesian S-HMM	53.64	26.20	19.93	5.01
	Bayesian SM-HMM	56.70	26.68	29.16	4.34
	Bayesian CS-HMM	53.15	25.43	20.15	5.02
	Bayesian CSM-HMM	59.79	29.51	30.13	4.30
	Bayesian A-HMM	43.45	22.37	7.91	5.71
	Bayesian AS-HMM	37.03	9.41	0.17	6.29
5	Bayesian HMM	55.51	25.16	20.13	5.00
	Bayesian S-HMM	53.89	24.93	18.65	5.11
	Bayesian SM-HMM	56.97	28.48	29.11	4.43
	Bayesian CS-HMM	54.95	25.07	19.94	5.02
	Bayesian CSM-HMM	60.10	30.48	30.90	4.31
	Bayesian A-HMM	44.60	19.39	8.39	5.75
	Bayesian AS-HMM	37.03	9.48	0.15	6.30
6	Bayesian HMM	56.79	27.31	23.25	4.79
	Bayesian S-HMM	64.43	26.33	29.22	5.54
	Bayesian SM-HMM	56.43	28.65	29.39	4.40
	Bayesian CS-HMM	57.38	28.87	24.64	4.72
	Bayesian CSM-HMM	59.32	30.93	30.95	4.30
	Bayesian A-HMM	46.24	30.01	11.36	5.34
	Bayesian AS-HMM	37.03 ⁴⁸	9.13	0.12	6.30
	Brown Clustering	54.91	30.70	26.78	4.47
	Anchor HMM	58.82	-	-	-

Table 5.3. Turkish stemming results for different hyperparameter sets

		Metu-Sabancı Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	47.55	-0.97	4.24	28048	1.24	1.21	0.77
	Bayesian SM-HMM	34.97	-1.17	3.16	34685	1.56	1.37	0.60
	Bayesian CS-HMM	57.31	-1.41	3.61	2284	1.32	1.18	0.72
	Bayesian CSM-HMM	39.51	-1.15	3.45	32180	1.42	1.27	0.66
	Bayesian A-HMM	37.49	-1.53	3.46	33366	1.43	1.31	0.66
	Bayesian AS-HMM	43.57	-0.72	5.03	30211	1.09	1.11	0.87
2	Bayesian S-HMM	47.51	-0.02	8.52	28124	1.27	1.28	0.76
	Bayesian SM-HMM	34.97	-1.78	3.16	34684	1.56	1.37	0.60
	Bayesian CS-HMM	57.76	-1.40	3.63	22627	1.30	1.17	0.73
	Bayesian CSM-HMM	40.55	-1.43	3.62	31605	1.38	1.24	0.68
	Bayesian A-HMM	39.15	-0.97	4.45	32495	1.39	1.31	0.68
	Bayesian AS-HMM	43.43	-0.72	5.04	30279	1.09	1.10	0.87
3	Bayesian S-HMM	47.30	-0.97	4.42	28185	1.24	1.22	0.77
	Bayesian SM-HMM	34.97	-1.78	3.16	34684	1.56	1.37	0.60
	Bayesian CS-HMM	57.47	-1.43	3.59	22771	1.31	1.17	0.72
	Bayesian CSM-HMM	39.50	-1.56	3.43	32184	1.42	1.27	0.66
	Bayesian A-HMM	37.30	-1.57	3.41	33457	1.45	1.32	0.65
	Bayesian AS-HMM	43.22	-0.77	4.89	30377	1.11	0.86	
4	Bayesian S-HMM	47.46	-0.02	8.54	28119	1.27	1.28	0.76
	Bayesian SM-HMM	34.97	-1.78	3.16	34682	1.56	1.37	0.60
	Bayesian CS-HMM	63.71	-0.50	5.80	19431	1.07	1.03	0.88
	Bayesian CSM-HMM	40.22	-1.44	3.60	31771	1.39	1.25	0.67
	Bayesian A-HMM	39.04	-1.01	4.35	32549	1.40	1.32	0.68
	Bayesian AS-HMM	43.22	-0.78	4.87	30380	1.10	1.11	0.86
5	Bayesian S-HMM	47.45	-0.95	4.47	28139	1.23	1.21	0.77
	Bayesian SM-HMM	34.97	-1.78	3.16	34682	1.56	1.37	0.60
	Bayesian CS-HMM	57.85	-1.42	3.60	22563	1.30	1.16	0.73
	Bayesian CSM-HMM	39.62	-1.53	3.47	32116	1.41	1.26	0.66
	Bayesian A-HMM	38.00	-1.45	3.58	33040	1.41	1.31	0.67
	Bayesian AS-HMM	44.19	-0.60	5.41	29869	1.08	1.10	0.88
6	Bayesian S-HMM	47.29	-0.02	8.51	28225	1.28	1.29	0.75
	Bayesian SM-HMM	34.96	-1.78	3.17	34687	1.56	1.37	0.60
	Bayesian CS-HMM	63.83	-0.50	5.83	19401	1.07	1.02	0.89
	Bayesian CSM-HMM	41.08	-1.37	3.71	31312	1.37	1.23	0.69
	Bayesian A-HMM	40.89	-0.60	5.46	31567	1.34	1.30	0.71
	Bayesian AS-HMM	44.40	-0.59	5.43	29756	1.08	1.10	0.88
	HPS	53.79	-0.80	4.82	24521	1.18	1.08	0.81
	Morfessor	52.06	-0.85	5.10	23311	1.10	1.23	0.77
	Linguistica	52.33	-0.90	5.02	24021	1.07	0.81	0.76

We give also a summary of the stemming results of Turkish in Figure 5.3..

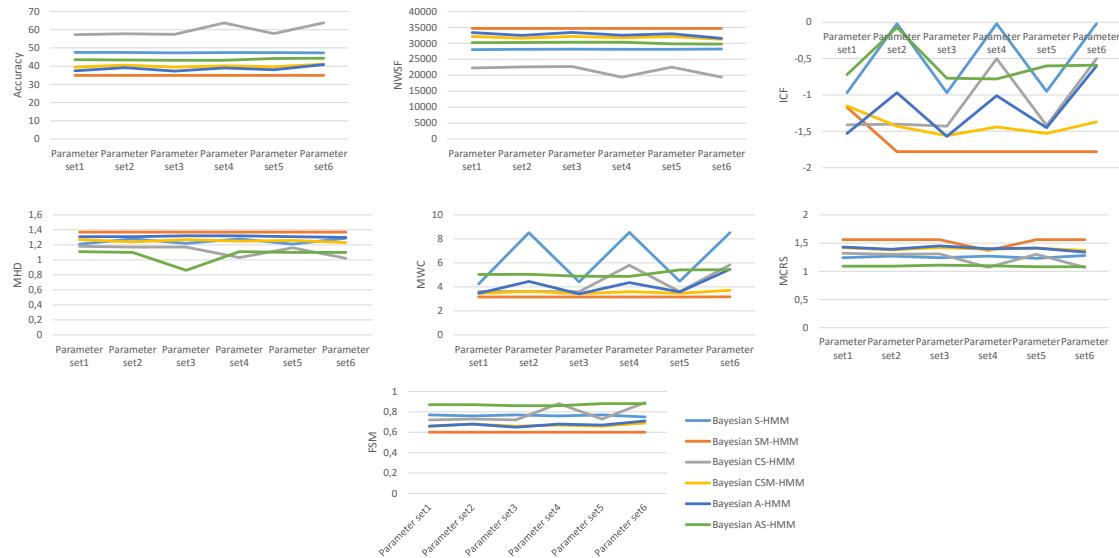


Figure 5.3. Sensitivity of parameter set for stemming performance of Turkish

The Figure 5.4. shows the relation of metrics. We can see that there is a reverse relationship between ICF, MWC, FSM and NWSF, MCRS, MHD. It can be concluded that the Bayesian CS-HMM model is acting stronger than the other models as a stemmer.

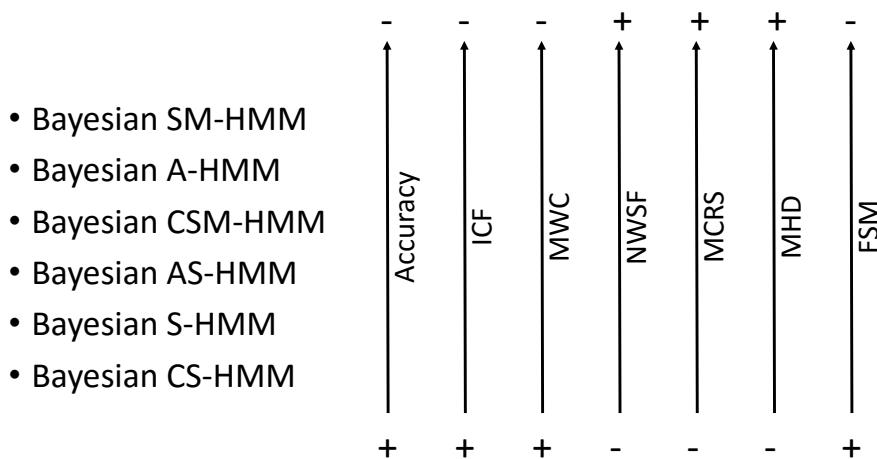


Figure 5.4. Features summary of proposed model for Turkish

The PoS tagging results for Hungarian are presented in Table 5.4. (see 12K results in Appendix C in Table 3.1.). The results show that Bayesian SM-HMM gives the highest scores for both Hungarian datasets. Bayesian CSM-HMM results are also close to Bayesian SM-HMM. We used a small dataset to train word2vec [78] in Hungarian. When we compare the Bayesian CSM-HMM results with the Turkish results, the small training set for word2vec [78] can be the reason of comparably low results of Bayesian-CSM HMM. This shows that using semantic features has a high impact on PoS tagging.

The overall PoS tagging results show that Bayesian-SM model outperform Brown Clustering [14], word-based Bayesian HMM [7], and Anchor HMM [29] for Hungarian according to Many-to-one, One-to-one, NMI, and VI measure.

When we compare the results of 12K and 24K datasets, it is seen that PoS tagging scores increase, as the dataset size increases.

Figure 5.5. shows the correlation of the dataset size and PoS tagging performance for Hungarian. The best PoS tagging results are obtained from Bayesian SM-HMM. Thus, we choose this model to analyze the affect of the size of the dataset.

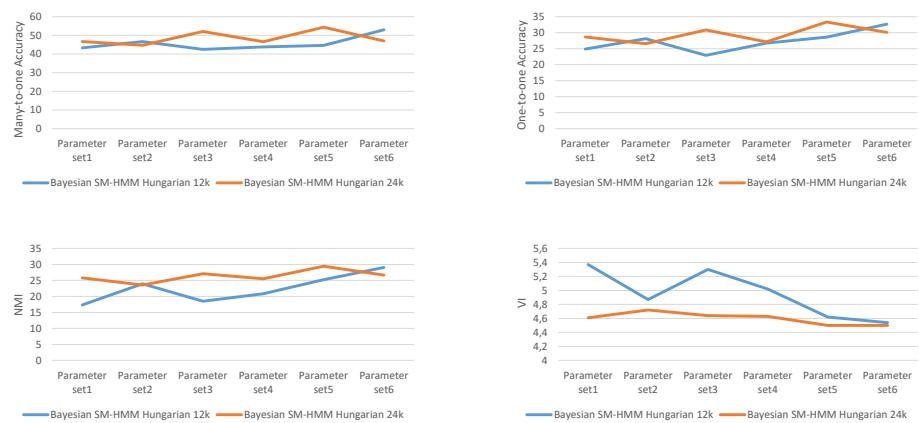


Figure 5.5. Sensitivity of dataset for PoS performance of Hungarian

Table 5.5. shows stemming results for 24K for six hyperparameter sets (see 12K results in Appendix C in Table 3.2.). Linguistica [53] results increase with the dataset size, whereas

Table 5.4. Hungarian24k PoS tagging results for different hyperparameter sets

		UD Hungarian24k Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	37.03	16.37	8.66	5.91
	Bayesian S-HMM	36.02	17.62	6.62	6.06
	Bayesian SM-HMM	46.61	28.67	25.80	4.61
	Bayesian CS-HMM	50.07	30.81	24.93	4.80
	Bayesian CSM-HMM	48.66	27.94	25.12	4.78
	Bayesian A-HMM	31.54	13.90	2.66	6.32
	Bayesian AS-HMM	29.07	9.70	0.19	6.50
2	Bayesian HMM	33.47	15.36	9.02	6.17
	Bayesian S-HMM	34.81	16.67	6.08	6.11
	Bayesian SM-HMM	44.70	26.52	23.58	4.72
	Bayesian CS-HMM	34.47	15.72	5.67	6.13
	Bayesian CSM-HMM	47.92	31.60	28.17	4.73
	Bayesian A-HMM	30.04	12.15	16.30	6.41
	Bayesian AS-HMM	29.07	9.79	0.21	6.50
3	Bayesian HMM	34.29	15.92	6.82	6.04
	Bayesian S-HMM	38.37	19.57	11.01	5.78
	Bayesian SM-HMM	52.07	30.86	27.10	4.64
	Bayesian CS-HMM	36.93	19.90	9.78	5.85
	Bayesian CSM-HMM	49.95	29.65	26.20	4.72
	Bayesian A-HMM	30.86	13.18	2.49	6.34
	Bayesian AS-HMM	29.07	9.25	0.18	6.50
4	Bayesian HMM	37.17	18.02	8.76	5.90
	Bayesian S-HMM	36.08	17.97	8.50	5.94
	Bayesian SM-HMM	46.59	27.12	25.51	4.63
	Bayesian CS-HMM	38.24	19.18	10.09	5.82
	Bayesian CSM-HMM	50.59	28.98	24.33	4.84
	Bayesian A-HMM	31.93	14.90	3.56	6.27
	Bayesian AS-HMM	29.07	9.40	0.18	6.51
5	Bayesian HMM	43.92	22.41	14.75	5.48
	Bayesian S-HMM	39.38	18.74	11.25	5.76
	Bayesian SM-HMM	54.38	33.33	29.46	4.50
	Bayesian CS-HMM	44.12	23.79	14.56	5.51
	Bayesian CSM-HMM	51.20	29.95	26.17	4.71
	Bayesian A-HMM	34.98	19.57	6.67	6.06
	Bayesian AS-HMM	29.07	9.68	0.21	6.50
6	Bayesian HMM	40.32	19.04	13.78	5.58
	Bayesian S-HMM	47.55	27.19	19.75	5.19
	Bayesian SM-HMM	47.00	30.08	26.65	4.50
	Bayesian CS-HMM	49.91	27.42	22.12	5.04
	Bayesian CSM-HMM	47.56	29.51	27.11	4.47
	Bayesian A-HMM	43.91	27.38	13.64	5.56
	Bayesian AS-HMM	29.07	9.54	0.22	6.50
	Brown Clustering	50.89	33.25	28.65	4.57
	Anchor HMM	48.86	-	-	-

our models’ results and HPS [45], Morfessor FlatCat [88] results are not. The best results are obtained from Linguistica [53]. We get the best results from the Bayesian CS-HMM model for stemming. This was the same for Turkish. Our results obtained from the Bayesian CS-HMM model are close to HPS [45] and far better than Morfessor FlatCat [88].

Table 5.6. shows PoS tagging results for Finnish 24K datasets for six hyperparameter sets (see 12K results in Appendix C in Table 3.3.). We get the highest scores in Many-to-one, NMI, and VI scores from the Bayesian CSM-HMM model. The highest One-to-one accuracy is obtained from Brown Clustering [14]. The results show that Bayesian SM-HMM and Bayesian CSM-HMM results are very close. This may be because of the small training set for word2vec [78] used for Bayesian CSM-HMM model.

Table 5.7. shows stemming results for Finnish 24K datasets for six hyperparameter sets (see 12K results in Appendix C in Table 3.4.). The best results are obtained from Linguistica [53]. Among our models, the highest scores are obtained from Bayesian CS-HMM model like the other two agglutinative languages: Turkish and Hungarian. Our results are far better than the results of the HPS [45] and Morfessor FlatCat [88].

Table 5.8. shows PoS tagging results for Basque 24K datasets for six hyperparameter sets (see 12K results in Appendix C in Table 3.5.). The best results are obtained from Brown Clustering [14]. The results obtained from 24k dataset shows that we get better result with larger dataset. The scores obtained from Bayesian SM-HMM and Bayesian CSM-HMM are very close to Brown Clustering [14] and Anchor HMM [29] for 24K dataset.

Table 5.9. shows stemming results for Basque 24K datasets for six hyperparameter sets (see 12K results in Appendix C in Table 3.6.). For 12K dataset, the best results are obtained from Morfessor FlatCat [88]. However, results of HPS [45], Morfessor FlatCat [88], Linguistica [53] and Bayesian CS-HMM are very close. In 24K dataset, the best results are obtained from Linguistica [53]. It is seen that, our stemming results are not affected by the dataset size.

Table 5.10. shows PoS tagging results for Penn 24K datasets for six hyperparameter sets (see 12K results in Appendix C in Table 3.7.). The best results are obtained from Anchor HMM [29] for Many-to-one accuracy. Brown Clustering [14] gives the highest for One-to-one, NMI, VI scores. However, it is seen that Bayesian HMM [7] model is far more behind the joint models.

Table 5.5. Hungarian24k stemming results for different hyperparameter sets

		UD Hungarian24k Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	48.26	-0.23	3.17	12400	1.46	1.50	0.62
	Bayesian SM-HMM	41.18	-0.39	2.83	14047	0.77	0.83	1.15
	Bayesian CS-HMM	57.03	0.72	13.00	13994	2.29	2.32	0.42
	Bayesian CSM-HMM	41.99	0.16	4.69	12036	0.92	0.96	1.02
	Bayesian A-HMM	41.37	-0.33	2.93	14008	1.16	1.22	0.80
	Bayesian AS-HMM	42.19	-0.11	3.49	13744	1.54	1.59	0.62
2	Bayesian S-HMM	48.00	-0.21	3.20	12451	1.48	1.52	0.64
	Bayesian SM-HMM	41.18	-0.39	2.83	14047	0.77	0.83	1.15
	Bayesian CS-HMM	56.95	-0.20	3.23	10314	1.25	1.28	0.76
	Bayesian CSM-HMM	42.12	-0.38	2.85	13826	0.80	0.86	1.11
	Bayesian A-HMM	41.42	-0.34	2.93	14000	1.17	1.22	0.80
	Bayesian AS-HMM	42.13	-0.13	3.42	13796	1.51	1.56	0.63
3	Bayesian S-HMM	46.31	0.20	4.86	12876	1.82	1.86	0.53
	Bayesian SM-HMM	41.18	-0.39	2.83	14048	0.77	0.83	1.15
	Bayesian CS-HMM	56.17	0.17	4.75	10503	1.51	1.54	0.63
	Bayesian CSM-HMM	42.19	-0.39	2.83	13861	0.79	0.85	1.12
	Bayesian A-HMM	41.07	-0.09	3.56	14100	1.42	1.46	0.67
	Bayesian AS-HMM	42.18	-0.10	3.51	13804	1.55	1.60	0.62
4	Bayesian S-HMM	45.50	0.20	4.89	13059	1.83	1.86	0.53
	Bayesian SM-HMM	41.18	-0.39	2.83	14047	0.77	0.83	1.115
	Bayesian CS-HMM	56.01	0.16	4.68	10552	1.51	1.54	0.63
	Bayesian CSM-HMM	41.99	-0.38	2.85	13814	0.79	0.85	1.12
	Bayesian A-HMM	41.16	-0.12	3.48	14074	1.38	1.43	0.68
	Bayesian AS-HMM	41.98	-0.13	3.43	13842	1.53	1.57	0.63
5	Bayesian S-HMM	48.17	-0.21	3.21	12420	1.48	1.51	0.65
	Bayesian SM-HMM	41.20	-0.39	2.83	14052	0.78	0.84	1.14
	Bayesian CS-HMM	57.17	-0.20	3.26	10254	1.24	1.28	0.76
	Bayesian CSM-HMM	42.13	-0.39	2.84	13828	0.80	0.86	1.12
	Bayesian A-HMM	41.40	-0.32	2.97	14005	1.19	1.24	0.79
	Bayesian AS-HMM	42.45	-0.05	3.67	13715	1.57	1.62	0.61
6	Bayesian S-HMM	45.87	0.22	5.01	12976	1.83	1.87	0.53
	Bayesian SM-HMM	41.18	-0.39	2.83	14048	0.77	0.83	1.15
	Bayesian CS-HMM	56.36	0.18	4.78	10469	1.50	1.53	0.64
	Bayesian CSM-HMM	42.25	-0.38	2.86	13800	0.80	0.86	1.11
	Bayesian A-HMM	41.27	-0.02	3.80	14045	1.50	1.55	0.63
	Bayesian AS-HMM	42.45	-0.06	3.62	13709	1.57	1.62	0.61
	HPS	58.98	0.09	3.87	9784	0.93	0.98	1.00
	Morfessor	45.89	-3.20	3.87	12907	2.36	0.99	0.41
	Linguistica	70.12	0.32	3.87	7154	0.73	0.82	1.21

Table 5.6. Finnish24k PoS tagging results for different hyperparameter sets

		Finnish24k Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	41.81	20.37	9.90	5.59
	Bayesian S-HMM	42.02	20.33	10.88	5.54
	Bayesian SM-HMM	46.98	22.54	18.86	5.03
	Bayesian CS-HMM	41.77	20.70	10.92	5.53
	Bayesian CSM-HMM	46.23	21.92	18.49	5.05
	Bayesian A-HMM	33.77	13.44	1.55	6.13
	Bayesian AS-HMM	32.10	9.92	0.18	6.22
2	Bayesian HMM	42.66	21.66	11.47	5.49
	Bayesian S-HMM	42.67	21.12	11.11	5.53
	Bayesian SM-HMM	47.25	23.25	19.79	4.95
	Bayesian CS-HMM	42.05	22.07	11.14	5.51
	Bayesian CSM-HMM	47.42	23.90	19.93	4.94
	Bayesian A-HMM	33.74	12.79	1.26	6.10
	Bayesian AS-HMM	32.10	9.80	0.22	6.21
3	Bayesian HMM	41.13	20.27	9.92	5.60
	Bayesian S-HMM	41.75	22.00	10.01	5.59
	Bayesian SM-HMM	46.78	22.42	17.58	5.11
	Bayesian CS-HMM	41.42	21.22	10.39	5.56
	Bayesian CSM-HMM	46.82	23.75	18.35	5.09
	Bayesian A-HMM	33.17	12.94	1.33	6.14
	Bayesian AS-HMM	32.10	9.37	0.15	6.22
4	Bayesian HMM	42.06	21.53	10.84	5.53
	Bayesian S-HMM	42.24	20.95	12.00	5.46
	Bayesian SM-HMM	46.34	22.82	18.92	5.02
	Bayesian CS-HMM	42.64	21.40	11.35	5.50
	Bayesian CSM-HMM	46.71	23.67	18.72	5.02
	Bayesian A-HMM	33.80	12.15	1.62	6.12
	Bayesian AS-HMM	32.10	9.40	0.16	6.22
5	Bayesian HMM	43.21	23.39	13.14	5.39
	Bayesian S-HMM	42.70	21.78	11.64	5.49
	Bayesian SM-HMM	48.46	23.45	20.24	4.94
	Bayesian CS-HMM	42.61	22.42	12.62	5.43
	Bayesian CSM-HMM	47.78	22.79	20.19	4.93
	Bayesian A-HMM	35.05	15.58	2.62	6.05
	Bayesian AS-HMM	32.10	9.47	0.17	6.22
6	Bayesian HMM	43.18	23.10	13.63	5.35
	Bayesian S-HMM	46.41	25.67	15.46	5.24
	Bayesian SM-HMM	47.65	22.98	20.33	4.92
	Bayesian CS-HMM	44.95	24.51	14.58	5.31
	Bayesian CSM-HMM	47.58	24.16	20.70	4.89
	Bayesian A-HMM	35.23	17.71	3.67	5.96
	Bayesian AS-HMM	32.10 ⁵⁵	9.42	0.21	6.22
	Brown Clustering	47.95	30.13	17.62	4.92
	Anchor HMM	43.73	-	-	-

Table 5.7. Finnish24k stemming results for different hyperparameter sets

		Finnish24k Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	27.40	-0.27	2.60	17400	1.83	2.08	0.47
	Bayesian SM-HMM	26.42	-0.51	2.19	17603	0.98	1.43	0.68
	Bayesian CS-HMM	38.98	-0.33	2.48	14618	1.50	1.82	0.53
	Bayesian CSM-HMM	27.22	-0.51	2.20	17411	0.99	1.43	0.68
	Bayesian A-HMM	25.90	-0.48	2.23	17747	1.25	1.64	0.60
	Bayesian AS-HMM	23.78	-0.10	2.99	18246	1.92	2.12	0.46
2	Bayesian S-HMM	24.55	0.25	4.48	18084	2.37	2.53	0.39
	Bayesian SM-HMM	26.42	-0.51	2.19	17604	0.98	1.43	0.68
	Bayesian CS-HMM	38.00	0.07	3.59	14856	1.89	2.15	0.45
	Bayesian CSM-HMM	27.55	-0.49	2.23	17332	1.00	1.44	0.68
	Bayesian A-HMM	25.10	-0.24	2.65	17943	1.60	1.92	0.51
	Bayesian AS-HMM	23.70	-0.11	2.98	18262	1.92	2.11	0.47
3	Bayesian S-HMM	27.65	-0.28	2.58	17346	1.83	2.09	0.47
	Bayesian SM-HMM	26.42	-0.52	2.19	17603	0.98	1.43	0.68
	Bayesian CS-HMM	38.71	-0.32	2.51	14684	1.52	1.84	0.53
	Bayesian CSM-HMM	27.22	-0.51	2.20	17408	0.99	1.43	0.68
	Bayesian A-HMM	25.95	-0.49	2.22	17736	1.25	1.64	0.60
	Bayesian AS-HMM	23.81	-0.12	2.95	18246	1.91	2.11	0.47
4	Bayesian S-HMM	24.76	0.27	4.54	18039	2.38	2.55	0.39
	Bayesian SM-HMM	26.42	-0.51	2.19	17603	0.98	1.43	0.68
	Bayesian CS-HMM	38.17	0.07	3.59	14815	1.88	2.13	0.46
	Bayesian CSM-HMM	27.55	-0.49	2.22	17333	1.00	1.43	0.68
	Bayesian A-HMM	25.16	-0.27	2.61	17929	1.55	1.88	0.52
	Bayesian AS-HMM	23.85	-0.13	2.91	18232	1.89	2.10	0.47
5	Bayesian S-HMM	28.28	-0.27	2.59	17193	1.82	2.08	0.47
	Bayesian SM-HMM	26.40	-0.52	2.19	17607	0.98	1.43	0.68
	Bayesian CS-HMM	38.94	-0.32	2.49	14631	1.52	1.84	0.53
	Bayesian CSM-HMM	27.32	-0.50	2.21	17386	1.00	1.44	0.68
	Bayesian A-HMM	26.13	-0.48	2.24	17688	1.25	1.64	0.60
	Bayesian AS-HMM	23.44	-0.06	3.11	18331	1.97	2.16	0.46
6	Bayesian S-HMM	24.10	0.27	4.54	18194	2.39	2.55	0.39
	Bayesian SM-HMM	26.42	-0.51	2.19	17603	0.98	1.43	0.68
	Bayesian CS-HMM	38.06	0.08	3.63	14846	1.90	2.15	0.45
	Bayesian CSM-HMM	27.61	-0.48	2.24	17318	1.00	1.44	0.68
	Bayesian A-HMM	24.75	-0.14	2.91	18025	1.74	2.02	0.48
	Bayesian AS-HMM	23.70	-0.07	3.09	18255	1.96	2.15	0.46
	HPS	27.18	-0.17	17984	1.75	1.2	1.98	0.58
	Morfessor	25.93	-0.34	18234	1.84	1.7	2.06	0.54
	Linguistica	45.40	0.37	3.32	13102	1.04	1.57	0.62

Table 5.8. Basque24k PoS tagging results for different hyperparameter sets

		Basque24k Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	44.37	20.91	12.17	5.41
	Bayesian S-HMM	45.53	20.92	11.21	5.47
	Bayesian SM-HMM	53.98	24.58	24.02	4.58
	Bayesian CS-HMM	44.26	21.34	11.02	5.48
	Bayesian CSM-HMM	55.01	26.25	25.03	4.52
	Bayesian A-HMM	35.17	12.58	2.07	6.04
	Bayesian AS-HMM	31.02	10.16	0.19	6.15
2	Bayesian HMM	47.45	23.36	14.05	5.27
	Bayesian S-HMM	48.32	19.41	14.34	5.23
	Bayesian SM-HMM	56.29	28.80	26.28	4.42
	Bayesian CS-HMM	49.00	23.42	13.57	5.30
	Bayesian CSM-HMM	54.90	24.55	25.68	4.43
	Bayesian A-HMM	37.76	14.37	3.86	5.93
	Bayesian AS-HMM	31.05	9.67	0.17	6.16
3	Bayesian HMM	44.42	22.03	11.19	5.49
	Bayesian S-HMM	45.05	19.84	10.92	5.39
	Bayesian SM-HMM	54.92	26.56	24.04	4.52
	Bayesian CS-HMM	44.68	20.05	10.69	5.38
	Bayesian CSM-HMM	54.24	25.87	24.05	4.55
	Bayesian A-HMM	35.48	12.61	2.33	5.95
	Bayesian AS-HMM	30.88	9.78	0.17	6.15
4	Bayesian HMM	43.60	20.75	11.00	5.49
	Bayesian S-HMM	46.65	20.80	12.10	5.39
	Bayesian SM-HMM	53.63	23.48	24.72	4.52
	Bayesian CS-HMM	47.60	20.46	12.22	5.38
	Bayesian CSM-HMM	54.31	20.81	24.22	4.55
	Bayesian A-HMM	38.16	14.52	3.23	5.95
	Bayesian AS-HMM	31.33	9.63	0.25	6.15
5	Bayesian HMM	49.90	24.50	15.49	5.20
	Bayesian S-HMM	47.73	23.75	13.79	5.31
	Bayesian SM-HMM	56.49	23.83	26.24	4.44
	Bayesian CS-HMM	51.44	25.14	16.83	5.13
	Bayesian CSM-HMM	58.68	28.25	27.05	4.45
	Bayesian A-HMM	41.17	18.63	6.27	5.76
	Bayesian AS-HMM	30.82	9.70	0.13	6.16
6	Bayesian HMM	57.07	29.54	22.47	4.75
	Bayesian S-HMM	57.38	29.95	23.63	4.66
	Bayesian SM-HMM	57.31	28.49	26.64	4.45
	Bayesian CS-HMM	58.13	30.41	23.90	4.66
	Bayesian CSM-HMM	58.37	29.48	27.65	4.36
	Bayesian A-HMM	44.82	18.75	9.22	5.52
	Bayesian AS-HMM	31. ⁵⁷ 53	10.36	0.23	6.16
Brown Clustering		60.63	30.37	28.71	4.31
Anchor HMM		58.20	-	-	-

Table 5.9. Basque24k stemming results for different hyperparameter sets

		Basque24k Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	35.55	-0.45	3.62	15460	1.39	1.80	0.54
	Bayesian SM-HMM	33.59	-0.69	3.11	15910	0.88	1.30	0.71
	Bayesian CS-HMM	49.12	-0.44	3.64	12197	1.09	1.52	0.62
	Bayesian CSM-HMM	36.97	-0.65	3.19	15098	0.89	1.31	0.71
	Bayesian A-HMM	33.25	-0.67	3.15	16000	0.99	1.42	0.66
	Bayesian AS-HMM	31.85	-0.28	4.09	16341	1.43	1.84	0.53
2	Bayesian S-HMM	31.85	0.10	5.89	16357	1.72	2.12	0.46
	Bayesian SM-HMM	33.55	-0.69	3.12	15923	0.88	1.30	0.71
	Bayesian CS-HMM	48.48	-0.03	5.07	12351	1.25	1.67	0.57
	Bayesian CSM-HMM	37.18	-0.62	3.25	15048	0.89	1.32	0.71
	Bayesian A-HMM	32.77	-0.49	3.51	16112	1.12	1.55	0.62
	Bayesian AS-HMM	31.75	-0.27	4.13	16364	1.42	1.83	0.53
3	Bayesian S-HMM	35.93	-0.46	3.60	15367	1.36	1.78	0.55
	Bayesian SM-HMM	33.59	-0.69	3.11	15910	0.88	1.30	0.71
	Bayesian CS-HMM	49.54	-0.43	3.68	12098	1.09	1.51	0.63
	Bayesian CSM-HMM	36.95	-0.66	3.18	15103	0.89	1.31	0.72
	Bayesian A-HMM	33.32	-0.68	3.12	15979	0.98	1.40	0.67
	Bayesian AS-HMM	32.07	-0.31	4.00	16294	1.40	1.81	0.54
4	Bayesian S-HMM	32.41	0.09	5.78	16216	1.71	2.11	0.47
	Bayesian SM-HMM	33.59	-0.69	3.12	15911	0.88	1.30	0.71
	Bayesian CS-HMM	48.85	-0.03	5.09	12267	1.24	1.66	0.57
	Bayesian CSM-HMM	37.26	-0.62	3.25	15029	0.89	1.31	0.72
	Bayesian A-HMM	32.82	-0.51	3.47	16109	1.11	1.54	0.62
	Bayesian AS-HMM	32.04	-0.31	3.98	16300	1.40	1.81	0.54
5	Bayesian S-HMM	36.18	-0.47	3.58	15305	1.35	1.76	0.55
	Bayesian SM-HMM	33.59	-0.69	3.11	15911	0.88	1.30	0.71
	Bayesian CS-HMM	49.10	-0.43	3.69	12199	1.09	1.51	0.63
	Bayesian CSM-HMM	36.97	-0.64	3.20	15099	0.90	1.32	0.71
	Bayesian A-HMM	33.32	-0.66	3.17	15980	1.01	1.44	0.66
	Bayesian AS-HMM	31.49	-0.19	4.04	16433	1.47	1.88	0.52
6	Bayesian S-HMM	31.92	0.09	5.78	16334	1.69	2.10	0.47
	Bayesian SM-HMM	33.54	-0.69	3.12	15922	0.88	1.30	0.71
	Bayesian CS-HMM	48.72	-0.01	5.16	12299	1.24	1.67	0.57
	Bayesian CSM-HMM	37.27	-0.60	3.28	15029	0.90	1.32	0.71
	Bayesian A-HMM	32.28	-0.38	3.81	16237	1.23	1.65	0.58
	Bayesian AS-HMM	31.53	-0.19	4.38	16417	1.48	1.89	0.52
	HPS	50.06	0.24	5.24	11579	0.77	1.21	0.77
	Morfessor	55.50	-0.64	5.24	10723	1.06	1.16	0.66
	Linguistica	53.17	0.43	5.24	11211	0.92	1.39	0.68

Since the English stems are not covered in Penn Treebank [81], we were not able to evaluate the English stemming results for this dataset.

Table 5.11. shows PoS tagging results for udEnglish 24K datasets for six hyperparameter sets (see 12K results in Appendix C in Table 3.8.). It is seen that the results of these datasets are very similar to the results on Penn Treebank [81].

Table 5.12. shows stemming results for udEnglish 24K datasets for six hyperparameter sets (see 12K results in Appendix C in Table 3.9.). The best results are obtained from Linguis-tica [53]. However, Bayesian CS-HMM model is much better than HPS [45] and Morfessor FlatCat [88].

Examples to correct and incorrect stems in all languages are given in Tables 5.13., 5.14., 5.15., 5.16., and 5.17. respectively. The results show that our joint model can find common endings, such as *si*, *a*, *mıştım* in Turkish.

Table 5.13. Examples to correct and incorrect stems of Turkish

Turkish	
Correct	Incorrect
koca-sı	hiç-bir
tuhaf-#	operasy-onunda
bil-miyor	Grossm-an
duvar-a	eşleri-ni
anla-mıştım	yirm-i

Table 5.14. Examples to correct and incorrect stems of Hungarian

Hungarian	
Correct	Incorrect
tanár-ok	pedi-g
fizetés-ének	ide-i
bértábla-bér	befagyasz-tott
az-#	pedag-ógus
felada-ként	emelle-t

Table 5.10. Penn 24K PoS tagging results for different hyperparameter sets

		Penn 24K Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	32.83	17.52	7.84	6.15
	Bayesian S-HMM	32.32	16.88	6.63	6.27
	Bayesian SM-HMM	43.95	29.19	26.94	4.70
	Bayesian CS-HMM	32.52	16.41	5.65	6.34
	Bayesian CSM-HMM	46.16	31.63	27.85	4.73
	Bayesian A-HMM	28.98	12.27	2.24	6.57
	Bayesian AS-HMM	28.98	10.08	0.19	6.69
2	Bayesian HMM	35.82	19.59	11.42	5.90
	Bayesian S-HMM	33.04	17.48	8.74	6.12
	Bayesian SM-HMM	47.24	32.84	28.56	4.61
	Bayesian CS-HMM	35.97	21.04	11.73	5.91
	Bayesian CSM-HMM	45.84	30.55	28.67	4.56
	Bayesian A-HMM	29.69	15.01	4.92	6.37
	Bayesian AS-HMM	28.98	9.47	0.17	6.72
3	Bayesian HMM	31.46	15.29	5.39	6.34
	Bayesian S-HMM	32.45	15.92	5.90	6.32
	Bayesian SM-HMM	45.77	28.81	25.96	4.81
	Bayesian CS-HMM	32.23	16.54	6.13	6.30
	Bayesian CSM-HMM	39.78	22.16	22.30	5.07
	Bayesian A-HMM	29.98	11.44	11.90	6.62
	Bayesian AS-HMM	28.98	9.76	0.14	6.70
4	Bayesian HMM	32.95	19.27	8.56	6.09
	Bayesian S-HMM	33.10	19.12	8.70	6.11
	Bayesian SM-HMM	48.10	33.02	28.26	4.65
	Bayesian CS-HMM	33.49	17.07	8.35	6.13
	Bayesian CSM-HMM	43.10	29.43	26.22	4.67
	Bayesian A-HMM	29.23	13.23	3.07	6.51
	Bayesian AS-HMM	28.98	10.51	.14	6.69
5	Bayesian HMM	36.86	22.07	13.81	5.74
	Bayesian S-HMM	33.82	19.17	9.08	6.10
	Bayesian SM-HMM	48.61	32.16	29.85	4.55
	Bayesian CS-HMM	36.60	24.98	14.41	5.73
	Bayesian CSM-HMM	52.28	37.27	33.64	4.36
	Bayesian A-HMM	30.19	16.73	6.04	6.30
	Bayesian AS-HMM	28.98	9.93	0.17	6.70
6	Bayesian HMM	48.14	32.19	25.56	4.97
	Bayesian S-HMM	43.91	28.22	23.38	5.09
	Bayesian SM-HMM	48.39	34.79	31.26	4.47
	Bayesian CS-HMM	46.75	32.65	26.22	4.91
	Bayesian CSM-HMM	45.60	31.92	28.71	4.58
	Bayesian A-HMM	36.11	22.35	15.19	5.67
	Bayesian AS-HMM	28.98 ⁶⁰	9.93	0.18	6.70
Brown Clustering		50.49	45.71	38.23	4.02
Anchor HMM		55.39	-	-	-

Table 5.11. udEnglish 24K PoS tagging results for different hyperparameter sets

		udEnglish 24K Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	22.87	16.20	6.62	6.60
	Bayesian S-HMM	20.40	15.48	4.83	6.77
	Bayesian SM-HMM	36.69	30.98	24.34	5.22
	Bayesian CS-HMM	21.07	15.80	5.37	6.73
	Bayesian CSM-HMM	36.11	29.75	24.43	5.24
	Bayesian A-HMM	18.76	11.51	2.44	6.93
	Bayesian AS-HMM	16.04	9.25	0.26	7.07
2	Bayesian HMM	22.49	15.76	6.97	6.58
	Bayesian S-HMM	21.54	15.37	6.81	6.58
	Bayesian SM-HMM	35.77	29.39	23.75	5.19
	Bayesian CS-HMM	22.85	16.97	7.57	6.55
	Bayesian CSM-HMM	34.54	29.88	24.37	5.16
	Bayesian A-HMM	17.18	11.52	2.25	6.96
	Bayesian AS-HMM	15.97	9.46	0.26	7.07
3	Bayesian HMM	22.18	16.55	6.02	6.64
	Bayesian S-HMM	20.52	14.65	4.71	6.78
	Bayesian SM-HMM	33.61	27.47	21.68	5.36
	Bayesian CS-HMM	20.18	14.77	4.89	6.77
	Bayesian CSM-HMM	34.98	29.43	23.74	5.23
	Bayesian A-HMM	16.60	10.79	1.19	7.03
	Bayesian AS-HMM	15.97	9.25	0.26	7.09
4	Bayesian HMM	24.93	18.69	7.94	6.49
	Bayesian S-HMM	21.46	15.86	6.85	6.62
	Bayesian SM-HMM	37.21	33.49	24.57	5.18
	Bayesian CS-HMM	21.00	14.78	5.77	6.70
	Bayesian CSM-HMM	35.64	29.85	24.96	5.06
	Bayesian A-HMM	17.28	11.16	2.02	6.97
	Bayesian AS-HMM	15.97	9.16	0.20	7.10
5	Bayesian HMM	31.39	24.43	15.33	5.94
	Bayesian S-HMM	25.65	19.80	8.86	6.47
	Bayesian SM-HMM	38.96	34.06	27.20	5.02
	Bayesian CS-HMM	24.07	18.02	8.01	6.54
	Bayesian CSM-HMM	38.90	33.27	27.18	5.03
	Bayesian A-HMM	25.98	20.08	8.76	6.45
	Bayesian AS-HMM	15.97	9.21	0.28	7.09
6	Bayesian HMM	36.19	27.89	19.96	5.65
	Bayesian S-HMM	30.43	23.34	15.00	6.01
	Bayesian SM-HMM	39.69	32.55	28.17	4.95
	Bayesian CS-HMM	37.71	31.30	22.99	5.40
	Bayesian CSM-HMM	39.06	32.95	28.55	4.95
	Bayesian A-HMM	22.77	16.80	7.76	6.51
	Bayesian AS-HMM	15.97 ⁶¹	9.16	0.26	7.09
Brown Clustering		51.97	47.32	40.25	4.14
Anchor HMM		48.79	-	-	-

Table 5.12. udEnglish 24K stemming results for different hyperparameter sets

		udEnglish 24K Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	54.70	-0.31	4.64	11199	0.97	1.05	0.93
	Bayesian SM-HMM	47.82	-0.12	5.42	12648	0.52	0.61	1.59
	Bayesian CS-HMM	79.95	-0.18	5.17	5134	0.34	0.43	2.20
	Bayesian CSM-HMM	50.07	-0.07	5.70	12109	0.63	0.72	1.35
	Bayesian A-HMM	47.07	-0.19	5.10	12833	0.66	0.75	1.30
	Bayesian AS-HMM	46.07	-0.10	5.52	13071	1.07	1.16	0.85
2	Bayesian S-HMM	49.49	0.09	6.76	12272	1.24	1.32	0.74
	Bayesian SM-HMM	47.82	-0.12	5.42	12648	0.52	0.61	1.59
	Bayesian CS-HMM	78.94	-0.01	6.01	5368	0.43	0.52	1.85
	Bayesian CSM-HMM	50.10	-0.03	5.89	12102	0.66	0.75	1.30
	Bayesian A-HMM	47.00	-0.13	5.38	12854	0.68	0.77	1.26
	Bayesian AS-HMM	46.03	-0.10	5.50	13083	1.07	1.16	0.85
3	Bayesian S-HMM	53.67	-0.30	4.68	11303	0.97	1.06	0.92
	Bayesian SM-HMM	47.82	-0.12	5.42	12647	0.51	0.61	1.59
	Bayesian CS-HMM	79.82	-0.18	5.17	5172	0.35	0.44	2.16
	Bayesian CSM-HMM	50.08	-0.08	5.66	12105	0.63	0.71	1.36
	Bayesian A-HMM	47.13	-0.20	5.07	12819	0.66	0.75	1.30
	Bayesian AS-HMM	46.16	-0.11	5.46	13050	1.04	1.13	0.87
4	Bayesian S-HMM	49.72	0.10	6.81	12217	1.23	1.31	0.75
	Bayesian SM-HMM	47.83	-0.12	5.42	12647	0.52	0.61	1.59
	Bayesian CS-HMM	78.79	-0.009	6.04	5404	0.44	0.52	1.83
	Bayesian CSM-HMM	50.10	-0.04	5.84	12100	0.64	0.73	1.33
	Bayesian A-HMM	47.00	-0.14	5.35	12851	0.68	0.77	1.26
	Bayesian AS-HMM	46.37	-0.13	5.40	13000	1.02	1.11	0.88
5	Bayesian S-HMM	53.41	-0.31	4.65	11372	0.98	1.07	0.92
	Bayesian SM-HMM	47.82	-0.12	5.42	12648	0.52	0.61	1.59
	Bayesian CS-HMM	80.01	-0.17	5.19	5125	0.34	0.43	2.21
	Bayesian CSM-HMM	50.14	-0.05	5.79	12092	0.65	0.73	1.33
	Bayesian A-HMM	47.02	-0.20	5.07	12845	0.66	0.75	1.29
	Bayesian AS-HMM	45.54	-0.04	5.82	13201	1.16	1.24	0.79
6	Bayesian S-HMM	49.30	0.11	6.86	12324	1.24	1.32	0.74
	Bayesian SM-HMM	47.82	-0.12	5.43	12650	0.52	0.61	1.58
	Bayesian CS-HMM	78.99	-0.01	6.04	5350	0.43	0.52	1.84
	Bayesian CSM-HMM	50.13	-0.006	6.07	12095	0.69	0.78	1.25
	Bayesian A-HMM	46.85	-0.11	5.47	12886	0.71	0.80	1.22
	Bayesian AS-HMM	45.54	-0.04	5.84	13198	1.16	1.24	0.79
	HPS	75.21	-0.02	6.00	6012	0.81	0.55	0.79
	Morfessor	63.05	-0.9	5.96	7048	0.72	0.78	1.02
	Linguistica	83.84	0.16	5.86	5040	0.69	0.83	1.17

Table 5.15. Examples to correct and incorrect stems of Finnish

Finnish	
Correct	Incorrect
niska+an	sai+si
suomenmaa+#	tänn+e
pappila+ssa	tul+ee
piste+ttä	kotii+n
valinta+nsa	oll+a

Table 5.16. Examples to correct and incorrect stems of Basque

Basque	
Correct	Incorrect
lortu-tako	mas-a
gero-#	mol-de
palestinar-rak	ematen
lan-ean	nahas-mendu
etxe-ra	baldin-tzak

Table 5.17. Examples to correct and incorrect stems of English

English	
Correct	Incorrect
respect-ed	caus-ing
year-s	troubl-e
of-#	thir-d
rumour-s	investme-nt
kill-ed	opera-ting

5.4. Conclusion

We proposed six different models that learn PoS tags and stems jointly for agglutinative languages. We did experiments Turkish, Finnish, Hungarian as agglutinative languages and

also Basque and English. We compared our PoS tagging and stemming results with other models. We showed that especially Bayesian CSM-HMM outperforms other models for POS tagging task and Bayesian CS-HMM outperforms other models for the stemming task.

6. CONCLUSION

6.1. Conclusion

In this thesis, we described the joint task of stemming and PoS tagging and presented six different unsupervised Bayesian joint PoS tagging and stemming models for agglutinative languages which extended the word-based Bayesian HMM model [7]. In agglutinative languages like Turkish, Finnish or Hungarian stemming and PoS tagging are crucial and connected processes. Joint models reveals effect of stemming and PoS tagging on each other. We did experiments on Turkish, Finnish, Hungarian, Basque and English. Results showed that the models can be applicable on other languages as well, although they are proposed for agglutinative languages.

The research questions are concluded accordingly:

- We showed joint learning of PoS tagging and stemming helps in both tasks in terms of their performance.
- We compared our PoS tagging with the Baseline model which is word-based. The results show that using stems and affixes rather than words improve PoS tagging results. Stemming results also improve in the joint task.
- Semantic information play an important role in this thesis as Schone and Jurafsky (2000) [89], Brychcin and Konopik (2015) [45], and Narasimhan et al. (2015) [90] demonstrated the value of semantic information on PoS tagging and morphology.

Our study makes a valuable contribution to the joint learning in the PoS tagging and stemming literature. Our experiments provides two major additions. First, we show that using stems instead of words for PoS tagging task is more suitable for agglutinative languages. Learning stems reduces sparsity in PoS tagging task. Second, using semantic information for PoS tagging and stemming helps improve the scores of both tasks (We hypothesize that morphologically related words - stem and word - have semantic similarity. The similarity can be calculated by embedding vectors of words [78].).

6.2. Future Research Directions

The proposed joint PoS tagging and stemming model is a fully unsupervised model and can be applicable to any language. However, there are many problems left unsolved by the proposed models in this thesis. Experiments prove that stemming is effective but that is limited to only regular words. In other words, model cannot handle stemming irregular words. For instance, if a suffix starts with a vowel in some disyllables, haplology can be seen in Turkish (*aǵız+i* ← *aǵızı*). This can be overcome by adding operations as features that describe the transformation of a word to another word to capture irregular words. Adding mechanism to find stems of irregular words may bring higher accuracy than current models. Applying the model for a high order NLP task such as Text Classification for an extrinsic evaluation remains as another future work.

A APPENDIX : PoS TAGSET REDUCTION

In this appendix, we present PoS tagset reduction for used datasets based on universal PoS tagset of **Petrov et al. (2011)** [2]. The reduced tagset for the Penn Treebank is given in Table 1.1., for the Finn Treebank in Table 1.2., for the UD Treebank for Basque, Hungarian and English in Table 1.3., 1.4., 1.5. respectievly, and for the Metu-Sabancı Turkish Treebank in Table 1.6..

Table 1.1. The mapping of the Universal tagset to the Penn Treebank tagset

Universal tagset	Penn Treebank tagset
VERB	VBP,VBD,VBG,VBN,VB,VBZ,MD
PRON	WP,PRP, <i>PRP</i> , <i>WP</i>
PUNCT	(“),(,),-LRB-,NONE-,RRB-,(.),(:),(”),\$
PRT	RP,TO
DET	WDT,EX,PDT,DT
NOUN	NN,NNP,NNPS,NNS
ADV	RB,RBR,WRB,RBS
ADJ	JJ,JJS
UNKNOWN	FW,UH
ADP	IN
NUM	CD
CONJ	CC

Table 1.2. The mapping of the Universal tagset to the FinnTreeBank tagset

Universal tagset	FinnTreeBank tagset
VERB	V
PRON	Pron
PUNCT	Punct
PRT	Pcle
DET	Det
NOUN	N
ADV	Adv
ADJ	A
UNKNOWN	Symb, Foreign, Interj
ADP	Adp
NUM	Num
CONJ	C

Table 1.3. The mapping of the Universal tagset to UD Basque TreeBank tagset

Universal tagset	UD Basque TreeBank tagset
VERB	VERB, AUX
PRON	PRON
PUNCT	PUNCT
PRT	PART
DET	DET
NOUN	NOUN, PROPN
ADV	ADV
ADJ	ADJ
UNKNOWN	SYM, INTJ, X
ADP	ADP
NUM	NUM
CONJ	CONJ

Table 1.4. The mapping of the Universal tagset to UD Hungarian TreeBank tagset

Universal tagset	UD Hungarian TreeBank tagset
VERB	VERB, AUX
PRON	PRON
PUNCT	PUNCT
PRT	PART
DET	DET
NOUN	NOUN, PROPN
ADV	ADV
ADJ	ADJ
UNKNOWN	X, INTJ
ADP	ADP
NUM	NUM
CONJ	CONJ , SCONJ

Table 1.5. The mapping of the Universal tagset to UD English TreeBank tagset

Universal tagset	UD English TreeBank tagset
VERB	VERB, AUX
PRON	PRON
PUNCT	PUNCT
PRT	PART
DET	DET
NOUN	NOUN, PROPN
ADV	ADV
ADJ	ADJ
UNKNOWN	X, INTJ, SYM
ADP	ADP
NUM	NUM
CONJ	CONJ , SCONJ

Table 1.6. The mapping of the Universal tagset to the Metu-Sabancı Turkish Treebank tagset

Universal tagset	Metu-Sabancı Turkish Treebank tagset
Noun	Noun_Pron,Noun_Ins,Noun_Nom,Noun_Verb,Noun_Loc, Noun_Acc,Noun_Abl,Noun_Gen, Noun_Dat,Noun_Adj, Noun_Num,Noun_Pnon,Noun_Postp,Noun_Equ
Adj	Adj_Noun,Adj_Verb,Adj_Adj_Pron,Adj_Postp,Adj_Num
Adv	Adv_Verb,Adv_Adj,Adv_Noun,Adv
Conj	Conj
Det	Det
Interj	Interj
Ques	Ques
Verb	Verb,Negp,Verb_Noun,Verb_Postp,Verb_Adj,Verb_Adv,Verb_Verb
Postp	Postp
Num	Num
Pron	Pron,Pron_Noun
Punc	Punc

B APPENDIX : Word2vec DATA

Word2vec data

In this appendix, we present datasets used to learn distributed representations of words by word2vec [78].

English : The corpus with first one billion characters from Wikipedia. (<http://mattmahoney.net/dc/text8.zip>)

Finn Treebank [82]: It is a treebank that has approximately 19 000 sentences or sentence fragments, and 162 000 word forms.

The Basque UD Treebank [83]: It is part of the Basque Dependency Treebank(BDT) [91]. The treebank consists of 8.993 sentences and 121.443 tokens.

The Hungarian UD Treebank [83]: It is derived from the Szeged Dependency Treebank [92]. It contains 1299 sentences and 42.032 words.

Turkish Boun Corpus [93]: A web corpus for Turkish is composed of four subcorpora. It has 423M words and 491M tokens.

C APPENDIX : RESULTS FOR 12K DATASETS

In this appendix, we present PoS tagging and stemming results for 12K datasets for six hyperparameter sets.

Table 3.1. Hungarian12K PoS tagging results for different hyperparameter sets

		UD Hungarian 12K Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	32.22	15.12	4.47	6.21
	Bayesian S-HMM	33.35	16.37	4.90	6.21
	Bayesian SM-HMM	43.25	24.85	17.34	5.37
	Bayesian CS-HMM	33.53	16.39	5.90	6.14
	Bayesian CSM-HMM	44.54	25.78	21.68	5.03
	Bayesian A-HMM	27.74	10.82	1.01	6.47
	Bayesian AS-HMM	27.74	10.15	0.27	6.52
2	Bayesian HMM	33.33	15.56	5.60	6.16
	Bayesian S-HMM	34.86	17.15	6.20	6.06
	Bayesian SM-HMM	47.61	28.12	23.94	4.87
	Bayesian CS-HMM	32.86	15.50	6.18	6.17
	Bayesian CSM-HMM	48.00	28.04	25.22	4.77
	Bayesian A-HMM	29.05	13.23	1.84	6.40
	Bayesian AS-HMM	2.74	10.25	0.36	6.15
3	Bayesian HMM	32.18	14.87	4.26	6.24
	Bayesian S-HMM	31.79	15.47	3.96	6.27
	Bayesian SM-HMM	42.44	22.92	18.50	5.30
	Bayesian CS-HMM	33.29	16.43	5.75	6.15
	Bayesian CSM-HMM	44.22	21.30	19.10	5.25
	Bayesian A-HMM	27.81	10.80	0.86	6.48
	Bayesian AS-HMM	27.74	9.84	0.28	6.52
4	Bayesian HMM	31.51	14.60	4.31	6.21
	Bayesian S-HMM	33.32	16.16	5.62	6.13
	Bayesian SM-HMM	43.80	26.74	20.84	5.02
	Bayesian CS-HMM	31.54	14.86	4.95	6.19
	Bayesian CSM-HMM	44.52	24.57	20.57	5.13
	Bayesian A-HMM	29.42	12.00	1.64	6.42
	Bayesian AS-HMM	27.74	9.60	0.29	6.52
5	Bayesian HMM	34.46	16.81	9.36	5.89
	Bayesian S-HMM	34.31	17.46	7.32	6.04
	Bayesian SM-HMM	44.60	28.60	25.21	4.62
	Bayesian CS-HMM	35.67	19.15	8.91	5.94
	Bayesian CSM-HMM	52.58	32.53	28.20	4.60
	Bayesian A-HMM	31.84	16.16	4.53	6.22
	Bayesian AS-HMM	27.74	9.99	0.28	6.51
6	Bayesian HMM	37.93	19.29	11.94	5.73
	Bayesian S-HMM	41.39	21.49	15.37	5.49
	Bayesian SM-HMM	52.96	32.66	29.04	4.54
	Bayesian CS-HMM	39.20	22.02	12.92	5.65
	Bayesian CSM-HMM	50.81	29.53	28.15	4.58
	Bayesian A-HMM	35.75	19.45	7.99	5.99
	Bayesian AS-HMM	27.74	10.08	0.31	6.52
Brown Clustering		50.98	33.83	28.42	4.62
Anchor HMM		48.56	-	-	-

Table 3.2. Hungarian12k stemming results for different hyperparameter sets

		UD Hungarian12k Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	48.90	-0.25	2.63	6116	1.37	1.41	0.69
	Bayesian SM-HMM	41.82	-0.32	2.52	6940	0.77	0.83	1.14
	Bayesian CS-HMM	57.07	-0.19	2.78	5131	1.21	1.24	0.78
	Bayesian CSM-HMM	42.37	-0.32	2.53	6876	0.79	0.86	1.11
	Bayesian A-HMM	41.91	-0.27	2.60	6937	1.15	1.21	0.80
	Bayesian AS-HMM	42.70	-0.10	2.96	6836	1.46	1.51	0.65
2	Bayesian S-HMM	46.32	0.12	3.75	6432	1.71	1.75	0.56
	Bayesian SM-HMM	41.82	-0.32	2.52	6939	0.77	0.83	1.14
	Bayesian CS-HMM	56.23	0.07	3.57	5244	1.40	1.43	0.68
	Bayesian CSM-HMM	42.63	-0.31	2.54	6845	0.80	0.86	1.11
	Bayesian A-HMM	42.67	-0.09	3.02	6958	1.34	1.40	0.70
	Bayesian AS-HMM	42.67	-0.12	2.92	6845	1.47	1.52	0.65
3	Bayesian S-HMM	49.00	-0.26	2.62	6104	1.36	1.40	0.69
	Bayesian SM-HMM	41.82	-0.32	2.52	6941	0.77	0.83	1.14
	Bayesian CS-HMM	57.72	-0.20	2.74	5057	1.19	1.22	0.79
	Bayesian CSM-HMM	42.39	-0.32	2.52	6873	0.79	0.85	1.12
	Bayesian A-HMM	41.70	-0.27	2.59	6966	1.17	1.22	0.79
	Bayesian AS-HMM	42.80	-0.12	2.92	6822	1.45	1.50	0.66
4	Bayesian S-HMM	46.10	0.09	3.67	6460	1.70	1.74	0.56
	Bayesian SM-HMM	41.82	-0.32	2.52	6940	0.77	0.83	1.14
	Bayesian CS-HMM	56.36	0.05	3.50	5231	1.38	1.42	0.68
	Bayesian CSM-HMM	42.60	-0.31	2.54	6849	0.79	0.86	1.17
	Bayesian A-HMM	41.93	-0.09	3.02	6942	1.34	1.39	0.70
	Bayesian AS-HMM	43.02	-0.12	2.92	6805	1.44	1.49	0.66
5	Bayesian S-HMM	48.46	-0.23	2.68	6166	1.38	1.43	0.68
	Bayesian SM-HMM	41.82	-0.32	2.52	6939	0.77	0.84	1.14
	Bayesian CS-HMM	57.40	-0.20	2.75	5086	1.19	1.22	0.79
	Bayesian CSM-HMM	42.58	-0.32	2.53	6851	0.80	0.86	1.10
	Bayesian A-HMM	41.82	-0.27	2.61	6953	1.17	1.23	0.79
	Bayesian AS-HMM	43.10	-0.07	3.05	6776	1.49	1.54	0.64
6	Bayesian S-HMM	46.25	0.12	3.76	6440	1.71	1.74	0.56
	Bayesian SM-HMM	41.82	-0.32	2.53	6940	0.78	0.84	1.13
	Bayesian CS-HMM	56.70	0.09	3.64	5187	1.41	1.44	0.67
	Bayesian CSM-HMM	42.76	-0.30	2.55	6831	0.80	0.86	1.10
	Bayesian A-HMM	41.93	-0.06	3.10	6947	1.40	1.45	0.67
	Bayesian AS-HMM	42.93	-0.08	3.03	6803	1.50	1.54	0.64
	HPS	58.69	0.13	3.29	4941	0.95	1.01	0.97
	Morfessor	45.45	-2.97	3.29	6541	2.39	0.99	0.41
	Linguistica	69.70	0.27	3.29	3621	0.74	0.84	1.87

Table 3.3. Finnish 12K PoS tagging results for different hyperparameter sets

		Finnish 12K Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	39.76	19.80	9.67	5.63
	Bayesian S-HMM	39.81	20.83	9.50	5.64
	Bayesian SM-HMM	45.61	22.51	16.33	5.23
	Bayesian CS-HMM	40.81	20.80	9.38	5.65
	Bayesian CSM-HMM	44.46	22.07	16.15	5.23
	Bayesian A-HMM	31.89	10.65	0.79	6.20
	Bayesian AS-HMM	31.69	9.82	0.35	6.23
2	Bayesian HMM	40.83	20.58	10.18	5.59
	Bayesian S-HMM	39.97	17.87	9.92	5.62
	Bayesian SM-HMM	46.28	22.61	18.06	5.10
	Bayesian CS-HMM	39.45	16.91	9.72	5.63
	Bayesian CSM-HMM	47.52	24.35	18.10	5.11
	Bayesian A-HMM	32.20	11.41	1.18	6.17
	Bayesian AS-HMM	31.69	10.08	0.44	6.22
3	Bayesian HMM	38.95	18.80	8.25	5.72
	Bayesian S-HMM	39.40	19.25	8.51	5.71
	Bayesian SM-HMM	44.79	22.41	14.79	5.31
	Bayesian CS-HMM	38.35	19.47	8.46	5.70
	Bayesian CSM-HMM	45.71	22.95	16.94	5.18
	Bayesian A-HMM	31.69	10.01	0.49	6.22
	Bayesian AS-HMM	31.69	10.21	0.32	6.22
4	Bayesian HMM	38.86	19.84	8.69	5.68
	Bayesian S-HMM	38.40	17.26	8.95	5.67
	Bayesian SM-HMM	45.70	22.56	16.13	5.22
	Bayesian CS-HMM	41.09	20.85	9.33	5.65
	Bayesian CSM-HMM	45.99	23.21	17.43	5.15
	Bayesian A-HMM	31.83	10.60	0.64	6.21
	Bayesian AS-HMM	31.69	9.70	0.31	6.23
5	Bayesian HMM	42.07	21.78	12.65	5.44
	Bayesian S-HMM	41.68	21.74	12.21	5.47
	Bayesian SM-HMM	48.23	23.68	20.80	4.92
	Bayesian CS-HMM	41.68	22.39	11.52	5.52
	Bayesian CSM-HMM	47.69	22.76	20.21	4.98
	Bayesian A-HMM	32.61	13.22	1.89	6.13
	Bayesian AS-HMM	31.69	9.89	0.36	6.32
6	Bayesian HMM	43.47	22.05	12.26	5.46
	Bayesian S-HMM	42.87	23.94	13.06	5.41
	Bayesian SM-HMM	47.60	23.35	21.36	4.87
	Bayesian CS-HMM	42.16	22.68	12.68	5.43
	Bayesian CSM-HMM	48.49	24.84	21.41	4.87
	Bayesian A-HMM	33.26	13.51	2.91	6.03
	Bayesian AS-HMM	31.69	9.88	0.32	6.22
Brown Clustering		44.33	28.21	17.58	4.96
Anchor HMM		45.23	-	-	-

Table 3.4. Finnish 12K stemming results for different hyperparameter sets

		Finnish 12K Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	29.40	-0.22	2.13	8467	1.74	2.00	0.49
	Bayesian SM-HMM	26.69	-0.35	1.94	8764	0.94	1.37	0.72
	Bayesian CS-HMM	39.38	-0.25	2.10	7261	1.49	1.79	0.54
	Bayesian CSM-HMM	27.22	-0.34	1.95	8700	0.95	1.37	0.71
	Bayesian A-HMM	26.40	-0.32	1.99	8815	1.29	1.64	0.59
	Bayesian AS-HMM	24.54	-0.10	2.38	9037	1.87	2.08	0.47
2	Bayesian S-HMM	26.48	0.16	3.16	8812	2.20	2.39	0.41
	Bayesian SM-HMM	26.68	-0.32	1.94	8766	0.94	1.37	0.71
	Bayesian CS-HMM	38.10	-0.001	2.62	7420	1.79	2.05	0.47
	Bayesian CSM-HMM	27.49	-0.33	1.97	8668	0.96	1.38	0.71
	Bayesian A-HMM	25.91	-0.21	2.16	8871	1.45	1.77	0.55
	Bayesian AS-HMM	24.35	-0.10	2.38	9058	1.88	2.08	0.47
3	Bayesian S-HMM	29.51	-0.24	2.11	8450	1.72	1.98	0.49
	Bayesian SM-HMM	26.69	-0.35	1.94	8765	0.94	1.37	0.71
	Bayesian CS-HMM	39.15	-0.24	2.10	7292	1.52	1.82	0.53
	Bayesian CSM-HMM	27.03	-0.34	1.95	8723	0.95	1.37	0.71
	Bayesian A-HMM	26.30	-0.31	1.99	8824	1.31	1.67	0.59
	Bayesian AS-HMM	24.56	-0.11	2.35	9032	1.84	2.06	0.48
4	Bayesian S-HMM	26.35	0.14	3.07	8833	2.19	2.38	0.41
	Bayesian SM-HMM	26.69	-0.35	1.94	8764	0.94	1.37	0.71
	Bayesian CS-HMM	38.40	-0.01	2.58	7381	1.77	2.03	0.48
	Bayesian CSM-HMM	27.32	-0.33	1.97	8688	0.95	1.37	0.71
	Bayesian A-HMM	25.90	-0.22	2.14	8872	1.44	1.77	0.55
	Bayesian AS-HMM	24.72	-0.10	2.37	9013	1.84	2.05	0.48
5	Bayesian S-HMM	28.85	-0.22	2.14	8527	1.76	2.01	0.49
	Bayesian SM-HMM	26.70	-0.35	1.94	8764	0.94	1.37	0.71
	Bayesian CS-HMM	39.08	-0.25	2.08	7303	1.50	1.80	0.54
	Bayesian CSM-HMM	27.39	-0.33	1.96	8680	0.94	1.37	0.71
	Bayesian A-HMM	26.30	-0.31	2.00	8825	1.30	1.65	0.59
	Bayesian AS-HMM	24.38	-0.06	2.47	9060	1.92	2.12	0.46
6	Bayesian S-HMM	26.07	0.17	3.19	8864	2.21	2.39	0.41
	Bayesian SM-HMM	26.69	-0.35	1.94	8765	0.94	1.37	0.71
	Bayesian CS-HMM	38.50	0.01	2.66	7373	1.77	2.03	0.48
	Bayesian CSM-HMM	27.64	-0.31	2.00	8650	0.98	1.39	0.70
	Bayesian A-HMM	25.69	-0.16	2.25	8897	1.51	1.82	0.54
	Bayesian AS-HMM	24.20	-0.05	2.48	9079	1.94	2.13	0.46
	HPS	28.19	0.15	1.85	8618	1.48	1.84	0.79
	Morfessor	24.47	-0.32	1.94	9049	1.54	2.49	0.85
	Linguistica	47.16	0.28	2.62	6340	0.96	1.48	0.66

Table 3.5. Basque 12K PoS tagging results for different hyperparameter sets

		Basque 12K Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	33.87	19.31	9.23	6.03
	Bayesian S-HMM	43.11	18.62	9.62	5.58
	Bayesian SM-HMM	48.49	24.96	17.07	5.10
	Bayesian CS-HMM	41.99	18.00	8.45	5.64
	Bayesian CSM-HMM	50.50	23.71	20.59	4.87
	Bayesian A-HMM	32.63	10.92	1.10	6.11
	Bayesian AS-HMM	31.63	10.56	0.44	6.14
2	Bayesian HMM	32.12	18.32	8.72	6.07
	Bayesian S-HMM	44.38	22.07	10.89	5.46
	Bayesian SM-HMM	49.68	22.32	20.98	4.86
	Bayesian CS-HMM	44.87	20.72	10.60	5.49
	Bayesian CSM-HMM	52.30	24.44	22.64	4.76
	Bayesian A-HMM	34.49	12.11	1.71	6.06
	Bayesian AS-HMM	31.57	11.11	0.32	6.14
3	Bayesian HMM	33.41	18.90	8.82	6.07
	Bayesian S-HMM	39.48	15.05	5.95	5.80
	Bayesian SM-HMM	46.37	22.87	16.12	5.17
	Bayesian CS-HMM	43.52	19.26	9.43	5.59
	Bayesian CSM-HMM	46.26	20.85	17.76	5.06
	Bayesian A-HMM	33.75	11.02	0.99	6.11
	Bayesian AS-HMM	31.29	10.23	0.31	6.15
4	Bayesian HMM	33.63	18.79	9.57	6.01
	Bayesian S-HMM	44.24	21.29	10.50	5.52
	Bayesian SM-HMM	50.96	25.79	18.84	4.98
	Bayesian CS-HMM	43.47	20.15	10.09	5.54
	Bayesian CSM-HMM	49.10	24.55	19.09	4.97
	Bayesian A-HMM	33.82	12.07	1.57	6.08
	Bayesian AS-HMM	31.67	11.09	0.36	6.14
5	Bayesian HMM	34.04	20.34	11.59	5.88
	Bayesian S-HMM	44.95	21.16	11.34	5.47
	Bayesian SM-HMM	53.98	26.37	23.70	4.64
	Bayesian CS-HMM	45.63	21.65	12.02	5.43
	Bayesian CSM-HMM	55.97	26.46	25.45	4.50
	Bayesian A-HMM	34.70	13.95	2.60	6.01
	Bayesian AS-HMM	31.37	11.09	0.45	6.14
6	Bayesian HMM	37.23	22.58	13.00	5.79
	Bayesian S-HMM	39.84	25.34	15.54	5.61
	Bayesian SM-HMM	46.27	25.49	24.68	4.92
	Bayesian CS-HMM	36.95	22.91	15.24	5.62
	Bayesian CSM-HMM	46.38	26.84	25.36	4.93
	Bayesian A-HMM	27.57	16.20	5.27	6.28
	Bayesian AS-HMM	24.35	10.01	0.31	6.64
Brown Clustering		58.33	27.50	25.67	4.53
Anchor HMM		56.37	-	-	-

Table 3.6. Basque 12K stemming results for different hyperparameter sets

		Basque 12K Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	36.49	-0.41	2.83	7627	1.30	1.72	0.56
	Bayesian SM-HMM	33.52	-0.55	2.58	7974	0.88	1.31	0.71
	Bayesian CS-HMM	49.28	-0.36	2.94	6091	1.06	1.48	0.63
	Bayesian CSM-HMM	36.62	-0.52	2.63	7601	0.88	1.30	0.72
	Bayesian A-HMM	33.00	-0.54	2.60	8039	1.02	1.45	0.65
	Bayesian AS-HMM	32.01	-0.20	3.32	8159	1.40	1.82	0.54
2	Bayesian S-HMM	32.95	-0.001	4.00	8055	1.59	2.00	0.49
	Bayesian SM-HMM	33.52	-0.55	2.58	7973	0.88	1.30	0.71
	Bayesian CS-HMM	49.16	-0.10	3.63	6106	1.16	1.58	0.60
	Bayesian CSM-HMM	36.95	-0.48	2.70	7562	0.89	1.31	0.72
	Bayesian A-HMM	32.30	-0.39	2.89	8125	1.14	1.57	0.61
	Bayesian AS-HMM	31.63	-0.22	3.26	8211	1.40	1.81	0.54
3	Bayesian S-HMM	36.64	-0.42	2.82	7613	1.30	1.73	0.56
	Bayesian SM-HMM	33.52	-0.55	2.58	7973	0.88	1.30	0.71
	Bayesian CS-HMM	48.63	-0.37	2.92	6167	1.08	1.51	0.62
	Bayesian CSM-HMM	36.68	-0.53	2.62	7594	0.87	1.29	0.72
	Bayesian A-HMM	33.10	-0.52	2.63	8030	1.03	1.46	0.65
	Bayesian AS-HMM	32.06	-0.25	3.19	8157	1.38	1.79	0.54
4	Bayesian S-HMM	33.28	-0.02	3.92	8017	1.58	1.98	0.49
	Bayesian SM-HMM	33.55	-0.55	2.58	7969	0.88	1.30	0.71
	Bayesian CS-HMM	48.96	-0.09	3.68	6131	1.17	1.60	0.60
	Bayesian CSM-HMM	37.22	-0.49	2.69	7528	0.88	1.30	0.72
	Bayesian A-HMM	32.48	-0.40	2.86	8106	1.14	1.56	0.61
	Bayesian AS-HMM	31.92	-0.26	3.17	8174	1.38	1.79	0.54
5	Bayesian S-HMM	36.16	-0.42	2.83	7669	1.33	1.74	0.55
	Bayesian SM-HMM	33.55	-0.55	2.58	7969	0.88	1.31	0.71
	Bayesian CS-HMM	49.27	-0.35	2.97	6093	1.08	1.50	0.63
	Bayesian CSM-HMM	36.80	-0.50	2.66	7579	0.90	1.32	0.71
	Bayesian A-HMM	33.21	-0.52	2.64	8016	1.05	1.47	0.64
	Bayesian AS-HMM	31.55	-0.16	3.43	8216	1.45	1.86	0.53
6	Bayesian S-HMM	32.95	0.003	4.03	8056	1.60	2.00	0.49
	Bayesian SM-HMM	33.49	-0.55	2.59	7977	0.88	1.31	0.71
	Bayesian CS-HMM	49.04	-0.08	3.68	6124	1.17	1.59	0.60
	Bayesian CSM-HMM	37.36	-0.47	2.73	7512	0.90	1.32	0.71
	Bayesian A-HMM	32.43	-0.29	3.10	8111	1.23	1.65	0.58
	Bayesian AS-HMM	31.36	-0.16	3.44	8238	1.45	1.87	0.53
	HPS	48.95	0.26	4.00	6153	0.81	1.26	0.74
	Morfessor	51.70	-0.50	4.00	5829	1.18	1.24	0.61
	Linguistica	52.75	0.38	4.00	5668	0.93	1.40	0.67

Table 3.7. Penn 12K PoS tagging results for different hyperparameter sets

		Penn 12K Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	32.85	14.47	4.50	6.39
	Bayesian S-HMM	32.94	15.13	4.52	6.40
	Bayesian SM-HMM	41.86	27.58	19.34	5.34
	Bayesian CS-HMM	33.01	14.80	4.49	6.40
	Bayesian CSM-HMM	42.54	24.70	20.48	5.15
	Bayesian A-HMM	29.86	10.76	1.25	6.63
	Bayesian AS-HMM	29.86	10.19	0.36	6.68
2	Bayesian HMM	32.96	16.60	5.49	6.28
	Bayesian S-HMM	33.41	15.56	5.37	6.29
	Bayesian SM-HMM	48.12	32.80	28.11	4.62
	Bayesian CS-HMM	33.21	16.13	5.27	6.33
	Bayesian CSM-HMM	45.70	28.25	25.62	4.80
	Bayesian A-HMM	29.86	14.16	2.86	6.50
	Bayesian AS-HMM	29.86	10.30	0.33	6.67
3	Bayesian HMM	32.15	13.75	3.18	6.48
	Bayesian S-HMM	32.20	14.46	3.86	6.44
	Bayesian SM-HMM	38.90	21.28	17.42	5.48
	Bayesian CS-HMM	33.14	14.77	4.34	6.41
	Bayesian CSM-HMM	39.93	23.34	17.55	5.45
	Bayesian A-HMM	29.86	10.64	1.08	6.64
	Bayesian AS-HMM	29.86	10.43	0.30	6.67
4	Bayesian HMM	32.11	15.21	4.22	6.39
	Bayesian S-HMM	33.21	14.98	5.55	6.29
	Bayesian SM-HMM	45.28	28.24	25.96	4.77
	Bayesian CS-HMM	32.78	14.67	4.10	6.42
	Bayesian CSM-HMM	44.29	29.75	24.71	4.79
	Bayesian A-HMM	29.86	11.24	1.49	6.60
	Bayesian AS-HMM	29.86	10.07	0.28	6.68
5	Bayesian HMM	33.30	18.25	8.35	6.12
	Bayesian S-HMM	33.18	16.92	7.14	6.21
	Bayesian SM-HMM	50.67	35.23	31.55	4.48
	Bayesian CS-HMM	33.28	16.79	6.79	6.24
	Bayesian CSM-HMM	48.01	31.24	28.43	4.63
	Bayesian A-HMM	29.86	13.52	3.09	6.49
	Bayesian AS-HMM	29.86	9.79	0.44	6.66
6	Bayesian HMM	35.04	21.91	11.58	5.88
	Bayesian S-HMM	39.02	23.73	14.89	5.66
	Bayesian SM-HMM	51.52	36.78	32.33	4.38
	Bayesian CS-HMM	38.64	23.18	15.11	5.67
	Bayesian CSM-HMM	52.21	34.74	31.58	4.50
	Bayesian A-HMM	34.46	19.54	11.04	5.90
	Bayesian AS-HMM	29.86	10.20	0.33	6.67
Brown Clustering		46.58	43.15	36.60	4.15
Anchor HMM		52.36	-	-	-

Table 3.8. udEnglish 12K PoS tagging results for different hyperparameter sets

		udEnglish 12K Treebank			
		Many-to-one	One-to-one	NMI	VI
1	Bayesian HMM	19.98	14.56	3.84	6.83
	Bayesian S-HMM	19.23	13.44	3.67	6.85
	Bayesian SM-HMM	32.75	29.93	19.81	5.55
	Bayesian CS-HMM	19.25	13.48	3.79	6.83
	Bayesian CSM-HMM	30.09	22.23	18.26	5.72
	Bayesian A-HMM	16.74	10.84	1.56	7.01
	Bayesian AS-HMM	15.58	9.69	0.44	7.07
2	Bayesian HMM	19.57	14.40	4.67	6.76
	Bayesian S-HMM	19.48	13.88	4.73	6.75
	Bayesian SM-HMM	35.06	27.06	21.96	5.44
	Bayesian CS-HMM	19.43	14.78	4.38	6.80
	Bayesian CSM-HMM	33.97	29.12	22.95	5.21
	Bayesian A-HMM	16.06	10.66	1.42	7.02
	Bayesian AS-HMM	15.54	9.32	0.35	7.09
3	Bayesian HMM	19.10	14.00	3.55	6.86
	Bayesian S-HMM	18.73	13.11	3.29	6.88
	Bayesian SM-HMM	28.91	22.44	16.24	5.83
	Bayesian CS-HMM	19.58	13.96	3.86	6.84
	Bayesian CSM-HMM	29.43	21.96	15.06	5.94
	Bayesian A-HMM	16.26	10.48	1.21	7.03
	Bayesian AS-HMM	15.51	9.64	0.37	7.09
4	Bayesian HMM	19.49	14.28	4.12	6.79
	Bayesian S-HMM	19.82	14.89	4.87	6.75
	Bayesian SM-HMM	28.78	21.35	18.08	5.71
	Bayesian CS-HMM	21.02	14.48	4.82	6.76
	Bayesian CSM-HMM	31.23	24.00	18.43	5.73
	Bayesian A-HMM	16.32	11.41	1.68	6.98
	Bayesian AS-HMM	15.67	9.36	0.38	7.09
5	Bayesian HMM	21.55	16.09	6.13	6.66
	Bayesian S-HMM	19.87	14.68	5.23	6.75
	Bayesian SM-HMM	37.26	30.92	25.11	5.71
	Bayesian CS-HMM	19.97	14.57	4.90	6.76
	Bayesian CSM-HMM	35.45	29.83	24.23	5.19
	Bayesian A-HMM	17.11	11.58	1.89	6.98
	Bayesian AS-HMM	15.55	9.82	0.41	7.06
6	Bayesian HMM	23.91	18.60	9.74	6.38
	Bayesian S-HMM	23.42	16.32	7.74	6.54
	Bayesian SM-HMM	38.86	32.89	27.21	5.01
	Bayesian CS-HMM	23.66	17.72	9.08	6.45
	Bayesian CSM-HMM	36.96	31.91	26.52	5.02
	Bayesian A-HMM	18.61	14.28	4.39	6.78
	Bayesian AS-HMM	15.53	9.35	0.37	7.04
Brown Clustering		47.92	42.37	37.23	4.36
Anchor HMM		48.72	-	-	-

Table 3.9. udEnglish 12K stemming results for different hyperparameter sets

		udEnglish 12K Treebank						
		Accuracy	ICF	MWC	NWSF	MCRS	MHD	FSM
1	Bayesian S-HMM	54.90	-0.31	3.69	5551	0.93	1.02	0.96
	Bayesian SM-HMM	47.29	-0.11	4.37	6417	0.52	0.61	1.58
	Bayesian CS-HMM	79.85	-0.17	4.14	2625	0.35	0.44	2.17
	Bayesian CSM-HMM	49.80	-0.07	4.49	6115	0.63	0.71	1.37
	Bayesian A-HMM	46.25	-0.19	4.08	6543	0.71	0.80	1.22
	Bayesian AS-HMM	45.45	-0.10	4.39	6640	1.09	1.17	0.84
2	Bayesian S-HMM	50.83	-0.01	4.79	6044	1.15	1.23	0.80
	Bayesian SM-HMM	47.29	-0.10	4.37	6417	0.53	0.62	1.58
	Bayesian CS-HMM	79.71	-0.06	4.56	2645	0.39	0.47	2.02
	Bayesian CSM-HMM	49.75	-0.04	4.64	6121	0.67	0.75	1.29
	Bayesian A-HMM	46.05	-0.13	4.28	6567	0.75	0.84	1.16
	Bayesian AS-HMM	45.43	-0.11	4.34	6641	1.08	1.16	0.84
3	Bayesian S-HMM	54.61	-0.32	3.66	5583	0.93	1.01	0.97
	Bayesian SM-HMM	47.29	-0.11	4.37	6417	0.53	0.61	1.58
	Bayesian CS-HMM	79.79	-0.16	4.14	2629	0.36	0.44	2.16
	Bayesian CSM-HMM	49.80	-0.08	4.49	6115	0.62	0.71	1.38
	Bayesian A-HMM	46.26	-0.19	4.04	6541	0.72	0.81	1.20
	Bayesian AS-HMM	45.67	-0.12	4.31	6614	1.06	1.14	0.86
4	Bayesian S-HMM	51.30	-0.02	4.72	5959	1.13	1.21	0.81
	Bayesian SM-HMM	47.29	-0.11	4.37	6417	0.52	0.61	1.58
	Bayesian CS-HMM	79.23	-0.05	4.59	2699	0.41	0.49	1.95
	Bayesian CSM-HMM	49.84	-0.05	4.61	6111	0.67	0.76	1.29
	Bayesian A-HMM	46.14	-0.13	4.28	6555	0.74	0.82	1.18
	Bayesian AS-HMM	45.78	-0.13	4.27	6600	1.05	1.14	0.86
5	Bayesian S-HMM	54.94	-0.32	3.66	5547	0.92	1.00	0.97
	Bayesian SM-HMM	47.29	-0.11	4.36	6417	0.53	0.62	1.58
	Bayesian CS-HMM	79.92	-0.16	4.15	2610	0.34	0.42	2.25
	Bayesian CSM-HMM	49.86	-0.06	4.55	6108	0.67	0.75	1.30
	Bayesian A-HMM	46.18	-0.19	4.04	6551	0.73	0.81	1.20
	Bayesian AS-HMM	45.00	-0.05	4.59	6693	1.16	1.24	0.79
6	Bayesian S-HMM	51.19	0.01	4.89	5972	1.15	1.23	0.80
	Bayesian SM-HMM	47.29	-0.10	4.39	6417	0.53	0.62	1.57
	Bayesian CS-HMM	79.05	-0.04	4.64	2714	0.41	0.50	1.93
	Bayesian CSM-HMM	49.95	-0.06	4.82	6098	0.71	0.80	1.23
	Bayesian A-HMM	46.11	-0.11	4.35	6560	0.77	0.86	1.14
	Bayesian AS-HMM	44.91	-0.05	4.61	6699	1.17	1.25	0.78
	HPS	71.69	-0.15	4.65	3012	0.89	0.95	1.85
	Morfessor	64.74	-0.8	4.74	5142	1.02	1.02	1.46
	Linguistica	83.57	0.14	4.67	2514	0.69	0.83	1.17

REFERENCES

- [1] Kairit Sirts and Tanel Alumäe. A hierarchical dirichlet process model for joint part-of-speech and morphology induction. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 407–416. Association for Computational Linguistics, **2012**.
- [2] Slav Petrov, Dipanjan Das, and Ryan McDonald. A universal part-of-speech tagset. *arXiv preprint arXiv:1104.2086*, **2011**.
- [3] Robert Krovetz. Viewing morphology as an inference process. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 191–202. ACM, **1993**.
- [4] Jorge Hankamer. Turkish generative morphology and morphological parsing. In *Second International Conference on Turkish Linguistics. Istanbul, Turkey*. **1984**.
- [5] Yu Hu, Irina Matveeva, John Goldsmith, and Colin Sprague. Using morphology and syntax together in unsupervised learning. In *Proceedings of the Workshop on Psychocomputational Models of Human Language Acquisition*, pages 20–27. Association for Computational Linguistics, **2005**.
- [6] Nicholas Ostler. *Endangered Languages: What Role for the Specialist? Proceedings of the Foundation for Endangered Languages (FEL) Conference (2nd, Edinburgh, Scotland, September 25-27, 1998)*. ERIC, **1998**.
- [7] Sharon Goldwater and Tom Griffiths. A fully Bayesian approach to unsupervised part-of-speech tagging. In *Annual meeting-association for computational linguistics*, volume 45, page 744. Citeseer, **2007**.
- [8] Jorge Hankamer. Morphological parsing and the lexicon. In *Lexical representation and process*, pages 392–408. MIT Press, **1989**.
- [9] Noam Chomsky. Aspects of the theory of syntax. *Cambridge, Mass*, **1965**.
- [10] Doug Cutting, Julian Kupiec, Jan Pedersen, and Penelope Sibun. A practical part-of-speech tagger. In *Proceedings of the third conference on Applied natural*

- language processing*, pages 133–140. Association for Computational Linguistics, **1992**.
- [11] Adwait Ratnaparkhi et al. A maximum entropy model for part-of-speech tagging. In *Proceedings of the conference on empirical methods in natural language processing*, volume 1, pages 133–142. Philadelphia, USA, **1996**.
- [12] Ezra Black, Fred Jelinek, John Lafferty, Robert Mercer, and Salim Roukos. Decision tree models applied to the labeling of text with parts-of-speech. In *Proceedings of the workshop on Speech and Natural Language*, pages 117–121. Association for Computational Linguistics, **1992**.
- [13] Noah A Smith and Jason Eisner. Contrastive estimation: Training log-linear models on unlabeled data. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 354–362. Association for Computational Linguistics, **2005**.
- [14] Peter F Brown, Peter V Desouza, Robert L Mercer, Vincent J Della Pietra, and Jenifer C Lai. Class-based n-gram models of natural language. *Computational linguistics*, 18(4):467–479, **1992**.
- [15] Steven Finch and Nick Chater. Bootstrapping syntactic categories using statistical methods. *Background and Experiments in Machine Learning of Natural Language*, 229:235, **1992**.
- [16] Alexander Clark. 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-Volume 7*, pages 91–94. Association for Computational Linguistics, **2000**.
- [17] Chris Biemann. Unsupervised part-of-speech tagging employing efficient graph clustering. In *Proceedings of the 21st international conference on computational linguistics and 44th annual meeting of the association for computational linguistics: student research workshop*, pages 7–12. Association for Computational Linguistics, **2006**.
- [18] Hinrich Schütze. Part-of-speech induction from scratch. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*, pages 251–258. Association for Computational Linguistics, **1993**.

- [19] Hinrich Schütze. Distributional part-of-speech tagging. In *Proceedings of the seventh conference on European chapter of the Association for Computational Linguistics*, pages 141–148. Morgan Kaufmann Publishers Inc., **1995**.
- [20] Lawrence R Rabiner, CH Lee, BH Juang, and JG Wilpon. Hmm clustering for connected word recognition. In *Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on*, pages 405–408. IEEE, **1989**.
- [21] Bernard Merialdo. Tagging english text with a probabilistic model. *Computational linguistics*, 20(2):155–171, **1994**.
- [22] Michele Banko and Robert C Moore. Part of speech tagging in context. In *Proceedings of the 20th international conference on Computational Linguistics*, page 556. Association for Computational Linguistics, **2004**.
- [23] Mark Johnson. Why doesn’t em find good hmm pos-taggers? In *EMNLP-CoNLL*, pages 296–305. **2007**.
- [24] Leonard E Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The annals of mathematical statistics*, 41(1):164–171, **1970**.
- [25] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6):721–741, **1984**.
- [26] David J.C. MacKay. Ensemble learning for hidden markov models. Technical report, **1997**.
- [27] Jianfeng Gao and Mark Johnson. A comparison of bayesian estimators for unsupervised hidden markov model pos taggers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 344–352. Association for Computational Linguistics, **2008**.
- [28] Jurgen Van Gael, Andreas Vlachos, and Zoubin Ghahramani. The infinite hmm for unsupervised pos tagging. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 678–687. Association for Computational Linguistics, **2009**.

- [29] Karl Stratos, Michael Collins, and Daniel Hsu. Unsupervised part-of-speech tagging with anchor hidden markov models. *Transactions of the Association for Computational Linguistics*, 4:245–257, **2016**.
- [30] Christos Christodoulopoulos, Sharon Goldwater, and Mark Steedman. Two decades of unsupervised pos induction: How far have we come? In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 575–584. Association for Computational Linguistics, **2010**.
- [31] Aria Haghghi and Dan Klein. Prototype-driven learning for sequence models. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 320–327. Association for Computational Linguistics, **2006**.
- [32] Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 582–590. Association for Computational Linguistics, **2010**.
- [33] Xipeng Qiu, F Eng Ji, Jiayi Zhao, and Xuanjing Huang. Joint segmentation and tagging with coupled sequences labeling. **2012**.
- [34] Kairit Sirts, Jacob Eisenstein, Micha Elsner, Sharon Goldwater, et al. Pos induction with distributional and morphological information using a distance-dependent chinese restaurant process. In *ACL (2)*, pages 265–271. **2014**.
- [35] Kemal Oflazer and İlker Kuruöz. Tagging and morphological disambiguation of turkish text. In *Proceedings of the fourth conference on Applied natural language processing*, pages 144–149. Association for Computational Linguistics, **1994**.
- [36] Kemal Oflazer and Gökhan Tür. Morphological disambiguation by voting constraints. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics*, pages 222–229. Association for Computational Linguistics, **1997**.

- [37] Dilek Z Hakkani-Tür, Kemal Oflazer, and Gökhan Tür. Statistical morphological disambiguation for agglutinative languages. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*, pages 285–291. Association for Computational Linguistics, **2000**.
- [38] Levent Altunyurt, Zihni Orhan, and Tunga Güngör. A composite approach for part of speech tagging in turkish. In *Proceeding of International Scientific Conference on Computer Science, Istanbul, Turkey*. **2006**.
- [39] Taner Dincer, Bahar Karaoglan, and Tarik Kisla. A suffix based part-of-speech tagger for turkish. In *Information Technology: New Generations, 2008. ITNG 2008. Fifth International Conference on*, pages 680–685. IEEE, **2008**.
- [40] Bi kent Universitfi. A tool for tagging turkish text.
- [41] Julie B Lovins. Development of a stemming algorithm. **1968**.
- [42] John Dawson. Suffix removal and word conflation. *ALLC bulletin*, 2(3):33–46, **1974**.
- [43] Martin F Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, **1980**.
- [44] D Paice Chris et al. Another stemmer. In *ACM SIGIR Forum*, volume 24, pages 56–61. **1990**.
- [45] Tomáš Brychcín and Miloslav Konopík. Hps: High precision stemmer. *Information Processing & Management*, 51(1):68–91, **2015**.
- [46] Jiaul H Paik, Dipasree Pal, and Swapan K Parui. A novel corpus-based stemming algorithm using co-occurrence statistics. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 863–872. ACM, **2011**.
- [47] Jiaul H Paik, Mandar Mitra, Swapan K Parui, and Kalervo Järvelin. Gras: An effective and efficient stemming algorithm for information retrieval. *ACM Transactions on Information Systems (TOIS)*, 29(4):19, **2011**.

- [48] Jiaul H Paik, Swapan K Parui, Dipasree Pal, and Stephen E Robertson. Effective and robust query-based stemming. *ACM Transactions on Information Systems (TOIS)*, 31(4):18, **2013**.
- [49] Prasenjit Majumder, Mandar Mitra, Swapan K Parui, Gobinda Kole, Pabitra Mitra, and Kalyankumar Datta. Yass: Yet another suffix stripper. *ACM transactions on information systems (TOIS)*, 25(4):18, **2007**.
- [50] Zellig S Harris. From phoneme to morpheme. *Language*, 31(2):190–222, **1955**.
- [51] Jinxi Xu and W Bruce Croft. Corpus-based stemming using cooccurrence of word variants. *ACM Transactions on Information Systems (TOIS)*, 16(1):61–81, **1998**.
- [52] John Goldsmith. Unsupervised learning of the morphology of a natural language. *Computational linguistics*, 27(2):153–198, **2001**.
- [53] John Goldsmith. An algorithm for the unsupervised learning of morphology. *Natural Language Engineering*, 12(04):353–371, **2006**.
- [54] Michela Bacchin, Nicola Ferro, and Massimo Melucci. The effectiveness of a graph-based algorithm for stemming. In *International Conference on Asian Digital Libraries*, pages 117–128. Springer, **2002**.
- [55] Massimo Melucci and Nicola Orio. A novel method for stemmer generation based on hidden markov models. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 131–138. ACM, **2003**.
- [56] Paul McNamee and James Mayfield. Character n-gram tokenization for european language text retrieval. *Information retrieval*, 7(1-2):73–97, **2004**.
- [57] Michela Bacchin, Nicola Ferro, and Massimo Melucci. A probabilistic model for stemmer generation. *Information processing & management*, 41(1):121–137, **2005**.
- [58] Jon M Kleinberg, Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew S Tomkins. The web as a graph: measurements, models, and methods. In *International Computing and Combinatorics Conference*, pages 1–17. Springer, **1999**.

- [59] Fuchun Peng, Nawaaz Ahmed, Xin Li, and Yumao Lu. Context sensitive stemming for web search. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 639–646. ACM, **2007**.
- [60] Jiaul H Paik and Swapan K Parui. A fast corpus-based stemmer. *ACM Transactions on Asian Language Information Processing (TALIP)*, 10(2):8, **2011**.
- [61] Manish Shrivastava, Nitin Agrawal, Bibhuti Mohapatra, Smriti Singh, and Pushpak Bhattacharya. Morphology based natural language processing tools for indian languages. In *Proceedings of the 4th Annual Inter Research Institute Student Seminar in Computer Science, IIT, Kanpur, India, April*. Citeseer, **2005**.
- [62] A Goweder, H Alhami, Tarik Rashed, and A Al-Musrati. A hybrid method for stemming arabic text. *Journal of computer Science*, URL: <http://eref.uqu.edu.sa/files/eref2/folder6/f181.pdf>, **2008**.
- [63] Giorgos Adam, Konstantinos Asimakis, Christos Bouras, and Vassilis Poulopoulos. An efficient mechanism for stemming and tagging: the case of greek language. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pages 389–397. Springer, **2010**.
- [64] Pratikkumar Patel Kashyap Popat and Pushpak Bhattacharyya. Hybrid stemmer for gujarati. In *23rd International Conference on Computational Linguistics*, page 51. **2010**.
- [65] Kartik Suba Dipti Jiandani and Pushpak Bhattacharyya. Hybrid inflectional stemmer and rule-based derivational stemmer for gujarati. In *Proceedings of the 2nd Workshop on South and Southeast Asian Natural Language Processing (WS-SANLP 2011)*, page 1. **2011**.
- [66] Tarık KİŞLA and Bahar KARAOĞLAN. A hybrid statistical approach to stemming in turkish: An agglutinative language. *Anadolu University of Sciences & Technology-A: Applied Sciences & Engineering*, 17(2), **2016**.
- [67] A Köksal. Bilgi erişim sorunu ve bir belge dizinleme ve erişim dizgesi tasarım ve gerçekleştirmesi. *ed: Yayınlanmamış Doçentlik Tezi, Hacettepe Üniversitesi, Ankara, Turkey*, **1979**.

- [68] Kemal Oflazer. Two-level description of turkish morphology. *Literary and linguistic computing*, 9(2):137–148, **1994**.
- [69] Aysın Solak and Fazlı Can. Effects of stemming on turkish text retrieval. In *Proceedings of the Ninth Int. Symp. on Computer and Information Sciences (ISCIS'94)*, pages 49–56. **1994**.
- [70] Hayri Sever and Yıldan Bitirim. Findstem: Analysis and evaluation of a turkish stemming algorithm. In *International Symposium on String Processing and Information Retrieval*, pages 238–251. Springer, **2003**.
- [71] B Dincer and Bahar Karaoglan. Stemming in agglutinative languages: A probabilistic stemmer for turkish. *Computer and Information Sciences-ISCIS 2003*, pages 244–251, **2003**.
- [72] Gülsen Eryigit and Eref Adali. An affix stripping morphological analyzer for turkish. In *Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, Innsbruck, Austria*, pages 299–304. **2004**.
- [73] Mehmet Dündar Akın and Ahmet Afşin Akın. Türk dilleri için açık kaynaklı doğal dil işleme kütüphanesi: Zemberek. *Elektrik Mühendisliği*, 431:38, **2007**.
- [74] Evren Kapusuz Çilden. Stemming turkish words using snowball, **2014**.
- [75] Muhammed Yavuz Nuzumlalı Arzucan Ozgür. Analyzing stemming approaches for turkish multi-document summarization.
- [76] Gunes Erkan and Dragomir R Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479, **2004**.
- [77] Ayberk Ozgür and BOUN EDU. An unsupervised stemming method for agglutinative languages.
- [78] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119. **2013**.
- [79] Gülşen Eryiğit and Kemal Oflazer. Statistical dependency parsing of turkish. **2006**.

- [80] Kemal Oflazer, Bilge Say, Dilek Zeynep Hakkani-Tür, and Gökhan Tür. Building a turkish treebank. *Treebanks*, pages 261–277, **2003**.
- [81] Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330, **1993**.
- [82] Atro Voutilainen, Tanja Purtonen, and Kristiina Muhonen. Outsourcing parsebanking: The finntreebank project. In *Shall We Play the Festschrift Game?*, pages 117–131. Springer, **2012**.
- [83] Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1659–1666. **2016**.
- [84] Marina Meilă. Comparing clusterings—an information based distance. *Journal of multivariate analysis*, 98(5):873–895, **2007**.
- [85] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617, **2002**.
- [86] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, **2012**.
- [87] William B Frakes and Christopher J Fox. Strength and similarity of affix removal stemming algorithms. In *ACM SIGIR Forum*, volume 37, pages 26–30. ACM, **2003**.
- [88] Stig-Arne Grönroos, Sami Virpioja, Peter Smit, and Mikko Kurimo. Morfessor FlatCat: An HMM-based method for unsupervised and semi-supervised learning of morphology. In *COLING*, pages 1177–1185. **2014**.
- [89] Patrick Schone and Daniel Jurafsky. Knowledge-free induction of inflectional morphologies. In *Proceedings of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*, pages 1–9. Association for Computational Linguistics, **2001**.

- [90] Karthik Narasimhan, Regina Barzilay, and Tommi Jaakkola. An unsupervised method for uncovering morphological chains. *arXiv preprint arXiv:1503.02335*, **2015**.
- [91] Itziar Aduriz et al. Construction of a basque dependency treebank. **2003**.
- [92] Veronika Vincze, Dóra Szauter, Attila Almási, György Móra, Zoltán Alexin, and János Csirik. Hungarian dependency treebank. In *LREC*, volume 10, pages 1855–1862. Citeseer, **2010**.
- [93] Haşim Sak, Tunga Güngör, and Murat Saraçlar. Turkish language resources: Morphological parser, morphological disambiguator and web corpus. In *GoTAL 2008*, volume 5221 of *LNCS*, pages 417–427. Springer, **2008**.

CURRICULUM VITAE

Credentials

Name,Surname: Necva BÖLÜCÜ
Place of Birth: Hatay,Turkey
Marital Status: Single
E-mail: necvaa@gmail.com
Address: Computer Engineering Dept., Hacettepe University
Beytepe-ANKARA

Education

BSc. : Computer Education and Instructional Technology Dept.,
Çukurova University, Turkey
BSc. : Computer Engineering Dept., Mustafa Kemal University, Turkey

Foreign Languages

English

Work Experience

Computer Teacher (2008-2015)
Research Assistant (2015-Present)

Areas of Experiences

Machine Learning, NLP, Unsupervised Learning

Project and Budgets

-

Oral and Poster Presentations

-

PUBLICATIONS

Bölükü, N.,Can, B. "Joint PoS Tagging and Stemming for Agglutinative Languages." *18th International Conference on Computational Linguistics and Intelligent Text Processing*. 2016.

Bölükü, Necva, Can, B. "Stem-based PoS Tagging for Agglutinative Languages." *Signal Processing and Communication Application Conference (SIU), 2017 25th. IEEE.* **2016**.

6

HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
THESIS/DISSERTATION ORIGINALITY REPORT

HACETTEPE UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
TO THE DEPARTMENT OF Computer Engineering.....

Date: 23/06/2017

Thesis Title / Topic: ...Unsupervised Joint Part-of-Speech Tagging and Stemming.....
....for Agglutinative Languages.....

According to the originality report obtained by myself/my thesis advisor by using the Turnitin plagiarism detection software and by applying the filtering options stated below on 23/06/2017 for the total of ...108..... pages including the a) Title Page, b) Introduction, c) Main Chapters, d) Conclusion sections of my thesis entitled as above, the similarity index of my thesis is ..11..... %.

Filtering options applied:

1. Bibliography/Works Cited excluded
2. Quotes ~~excluded~~ / included
3. Match size up to 5 words excluded

I declare that I have carefully read Hacettepe University Graduate School of Science and Engineering Guidelines for Obtaining and Using Thesis Originality Reports; that according to the maximum similarity index values specified in the Guidelines, my thesis does not include any form of plagiarism; that in any future detection of possible infringement of the regulations I accept all legal responsibility; and that all the information I have provided is correct to the best of my knowledge.

I respectfully submit this for approval.

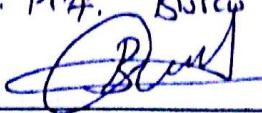
23.06.2017 
Date and Signature

Name Surname: Necmi Boluč
Student No: N15124594
Department: Computer Engineering
Program: Computer Engineering
Status: Masters Ph.D. Integrated Ph.D.

ADVISOR APPROVAL

APPROVED.

Asst. Prof. Burcu Can Başlalılar



(Title, Name Surname, Signature)