Applications of Hidden Markov Models in Natural Language Processing

Liviu-Ștefan Neacșu-Miclea
ICA 246/2

Agenda

- Introduction
- Motivation of the study
- Hidden Markov Models (HMMs)
- HMMs in Handwritten Character Recognition
- HMMs in Named Entity Recognition (NER)
- HMMs in Machine Translation (MT)
- Discussion
- Conclusion

Introduction

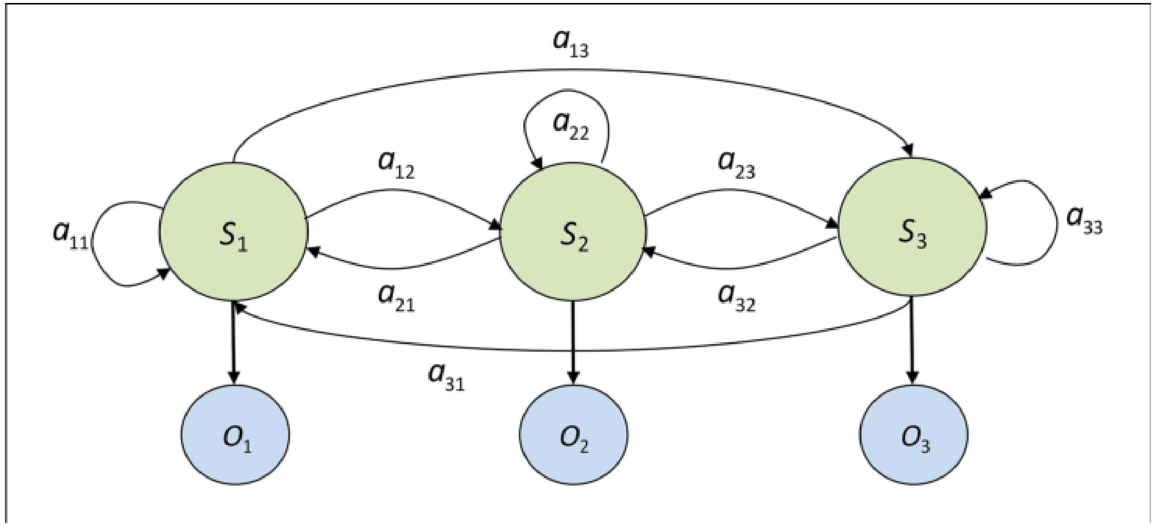
- **Hidden Markov Models (HMMs)**
  - Handle sequential data where the true process (hidden states) is not directly observable.
  - Captures temporal dependencies
  - Models latent processes affecting visible outcomes
  - Applications in: Speech recognition, bioinformatics (gene prediction), finance, NLP (part-of-speech tagging)
- **Offline Handwritten Character Recognition (HTR)**
  - Convert handwritten text into its corresponding sequence of characters
- **Named Entity Recognition (NER)**
  - identify and classify entities such persons, locations and organizations in a text
- **Machine Translation (MT)**
  - automatically translate text from one language to another without human intervention

Motivation

HTR	NER	MT

- Historical texts digitization
- Postal services and form processing
 	
- Variations and noise make it a difficult problem
 	
- HMMs were an early solution, but now CNNs/RNNs are better
	- Preprocessing step in some NLP applications (text summarization, MT, information retrieval)
	
	- For this reason, high NER performance is critical
		- Increased necessity in our interconnected world
		- Challenge: capture the nuances of different languages
		
		- History:
		  - Rule-Based-Systems
		  - **Statistical Machine Translation (SMT)**
		  - Neural Machine Translation (Google Translate)

Hidden Markov Models (HMMs) – Quick Recap

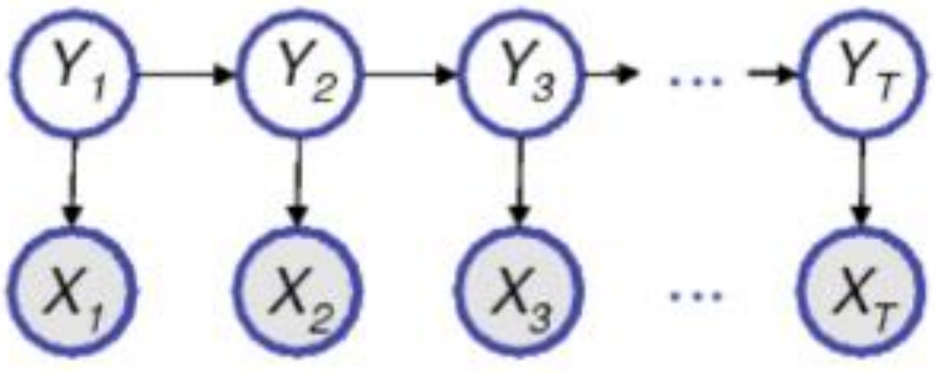


- A statistical model
  - Markov assumption
  - Hidden (unobservable) states
  - Used in sequence modeling
- Key components
  - **States:** Hidden variables that follow the Markov property
  - **Observations:** Visible outputs linked probabilistically to the states
  - **Transition Probabilities:** Likelihood of moving from one state to another
  - **Emission Probabilities:** Likelihood of an observation given a state
  - **Initial Probabilities**: Starting probability distribution over states
- Associated problems
  - **Evaluation**: Compute probability of an observed sequence
  - **Decoding**: Find the most likely state sequence (e.g., with Viterbi algorithm)
  - **Learning**: Estimate model parameters (e.g., using Baum-Welch algorithm)

HMM inHandwritten Word Recognition

Krevat, E., Cuzzillo, E., 2006. Improving off-line handwritten character recognition with hidden markov models. Transaction on Pattern Analysis and Machine Learning 33

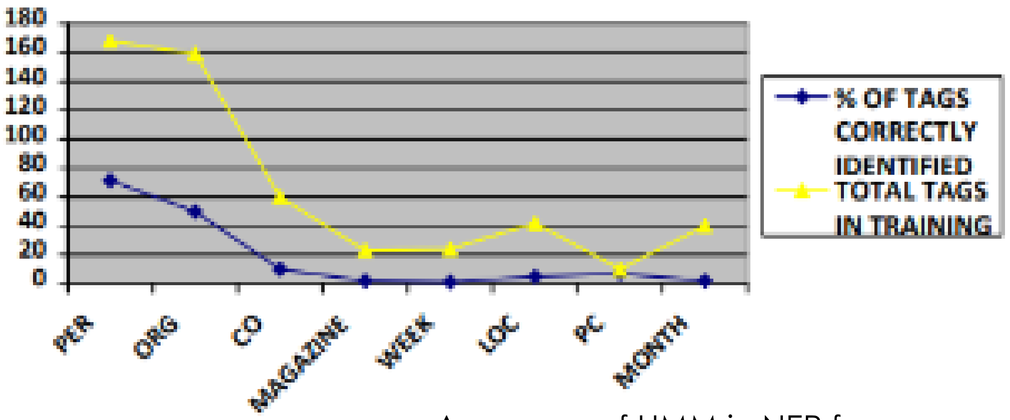
- **Goal**: Improve handwritten word recognition by modeling correlations between adjacent characters with HMMs
- **Approach:**
  - **Presegmentation of characters required.**
  - Hidden states = characters, observations = pixel images
  - First-order HMM; Naive Bayes used for emission probabilities.
  - **Optimized Viterbi** to reduce transition noise at word ends.
- **Experimental results:**
  - HMM achieved higer accuracy (71%) than Naïve Bayes (62%)
  - Dictionary creation and lookup works best on small datasets (99.3%)
- **Findings of the study:**
  - HMMs effectively capture character dependencies and optimizations payed off.
  - Dataset size and characteristics heavily impact model performance.



HMM in Named Entity Recognition

Morwal, S., Jahan, N., Chopra, D., 2012. Named entity recognition using hidden markov model (hmm). International Journal on NaturalLanguage Computing (IJNLC) Vol 1.

- **Goal:** Improve named entity recognition by designing an efficient approach that can be applied to any language independently
- **Approach:**
  - Hidden states = named entity tags; observations = words
  - HMM parameters learned from annotated corpus.
  - Viterbi algorithm used for sequence decoding.
  - No hard-coded entity states — dynamic and flexible
  - Minimal human effort to adapt to new languages
  - Captures local context effectively
- **Experimental results:**
  - No quantitative metrics in original metrics (proof by examples on Hindi)
  - High performance in languages like Dari (94% acc) and English (74% F-Measure)
- **Findings of the study:**
  - Practical, scalable method for NER tasks.



Accuracy of HMM in NER for English language

HMM in Machine Translation (1)

Horiguchi, S., Shimazu, A., Nguyen, M.L., 2003. Translation template learning based on hidden markov modeling, in: Language, Informationand Computation: Proceedings of the 17th Pacific Asia Conference, 1-3 October, 2003, Sentosa, Singapore, Waseda University. pp. 269–276.

- **Goal:** Improve machine translation by combining Example-Based Machine Translation (EBMT) with HMMs
- **Two-steps method**
  - Learn translation templates from bilingual corpora (TTL algorithm).
  - Reformulate translation as finding the most probable sequence of templates via HMM.
- **Obtaining translation templates**
  - Identify similarities and differences
    - I bought the book for John | => TR : "I bought the" X "for John"
I bought the ring for John |
    - Found using two heuristics (combined form Translation Template Learning TTL):
      - Similarity Translation Template (STTL) = replace differences with variables
      - Difference Translation Template (DTTL) = replace similarities with variables

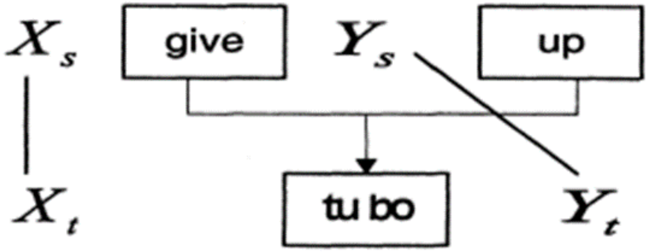


Figure 1. Template rule example

$$S_1S_2\ldots S_n \leftrightarrow T_1T_2\ldots T_k$$

Template rule: X "necessary to launch" Y Z <=>
X' "can thiet de bat dau" Y' Z'

HMM in Machine Translation (2)

Horiguchi, S., Shimazu, A., Nguyen, M.L., 2003. Translation template learning based on hidden markov modeling, in: Language, Informationand Computation: Proceedings of the 17th Pacific Asia Conference, 1-3 October, 2003, Sentosa, Singapore, Waseda University. pp. 269–276.

- **HMM specification:**
  - Hidden states = lexical rules (translation template without variables)
  - Observation = input sentence
  - Transition probabilities = likelihood of moving between lexical rules
  - Emission probabilities = matching input substrings with lexical rules
  - Viterbi algorithm used for decoding; Forward-Backward for training
- **Experimental results**
  - Translation accuracy **81%** compared to SOTA at that time 34%.
  - Effective on English-Vietnamese corpus (1200 sentence pairs, 11034 template rules)
- **Findings of this study**
  - Captures contextual nuances and handles structural variations
  - Reduces computational complexity compared to pure EBMT
  - Stronger translation quality, faster processing
  - Potential for broader application in multilingual NLP tasks

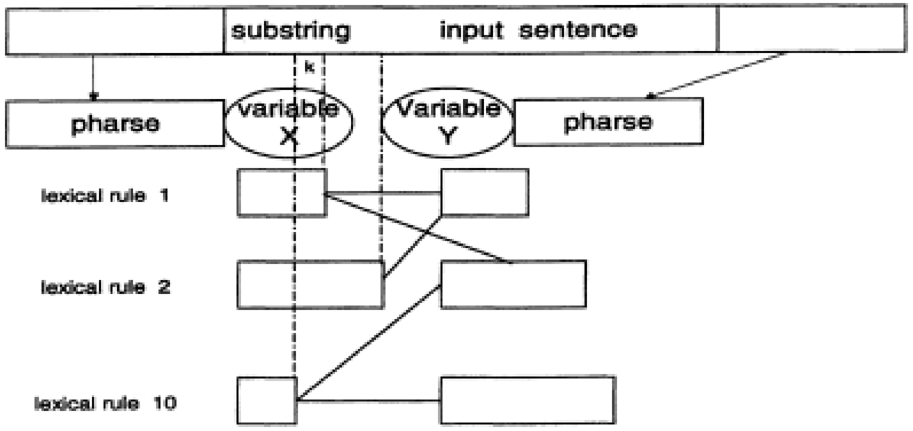


Figure 2. Example of translation based HMM

Discussion

- **Flexibility**
  - HMMs effectively model sequential data across handwriting recognition, NER, and machine translation
- **Problem-Specific Setup**
  - Hidden states and observations differ by task (characters, entity tags, lexical rules)
- **Common Techniques**
  - All use the Viterbi algorithm for optimal sequence prediction
  - Evaluation with accuracy, precision, recall, and F-measure
- **Strengths and Limits**
  - Capture contextual dependencies and patterns
  - Markov assumption simplifies the model but may limit learning depth.

	Krevat & Cuzzillo (2006) [11]	Morwal, Jahan, & Chopra (2012) [13]	Horiguchi, Shimazu, & Nguyen (2003) [8]
Task	Off-line handwritten character recognition	Named entity recognition	Translation template learning for machine translation
HMM Techniques	First-order HMM, Naive Bayes for emission probabilities over pixels, Viterbi optimization ignoring inter-word transitions	Language-agnostic HMM with dynamic states based on NE tags, learning by example from annotated corpus, Viterbi decoding	HMM for learning lexical translation rules (hidden states), input substrings as observations, Viterbi algorithm for template application
Evaluation Metrics	Accuracy (compared to Naive Bayes and dictionary lookup)	Precision, Recall, F-Measure [5]	Improvement in translation accuracy, low computational complexity, outperformed TTL in sentence reduction

Conclusion

- HMMs remain a flexible and powerful tool for complex sequence modeling across disciplines
- Effective in handwritten recognition, NER, and machine translation by modeling hidden processes behind observed data
- Advantages:
  - High adaptability even with limited data
  - Boost in accuracy over simpler methods
  - Greater interpretability compared to many modern models
- Continued Relevance
  - Still valuable when data is scarce, resources are limited, or interpretability matters
  - Opportunities exist to hybridize HMMs with modern deep learning techniques

Bibliography & References

- Jurafsky, D., Martin, J.H., 2018. Hidden markov models. Speech and language processing 3.
- Chopra, D., Morwal, S., 2013. Named entity recognition in english using hidden markov model. International Journal
- Chopra, D., Morwal, S., Purohit, G., 2013. Hidden markov model based named entity recognition tool. International Journal in Foundationsof Computer Science & Technology (IJFCST) 3, 67–73
- Morwal, S., Jahan, N., Chopra, D., 2012. Named entity recognition using hidden markov model (hmm). International Journal on NaturalLanguage Computing (IJNLC) Vol 1.
- Horiguchi, S., Shimazu, A., Nguyen, M.L., 2003. Translation template learning based on hidden markov modeling, in: Language, Informationand Computation: Proceedings of the 17th Pacific Asia Conference, 1-3 October, 2003, Sentosa, Singapore, Waseda University. pp. 269–276.
- Krevat, E., Cuzzillo, E., 2006. Improving off-line handwritten character recognition with hidden markov models. Transaction on PatternAnalysis and Machine Learning 33.
- https://www.quantconnect.com/research/17900/intraday-application-of-hidden-markov-models/