



Pós-Graduação em Ciência da Computação

Raphael Cândido Brito

Mapas Auto-Organizáveis para Reconhecimento de Palavras e Descoberta de Padrões em Séries Temporais



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
<http://cin.ufpe.br/~posgraduacao>

Recife
2018

Raphael Cândido Brito

Mapas Auto-Organizáveis para Reconhecimento de Palavras e Descoberta de Padrões em Séries Temporais

Trabalho apresentado ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Inteligência Computacional.

Orientador: Hansenclever de França Bassani

Recife
2018

Catálogo na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

B862m Brito, Raphael Cândido
Mapas auto-organizáveis para reconhecimento de plavaras e descoberta de padrões em séries temporais / Raphael Cândido Brito. – 2018.
65 f.: il., fig., tab.

Orientador: Hansenclever de França Bassani.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2018.
Inclui referências.

1. Inteligência computacional. 2. Séries temporais. I. Bassani, Hansenclever de França (orientador). II. Título.

006.3

CDD (23. ed.)

UFPE- MEI 2018-146

Raphael Cândido Brito

Mapas Auto-Organizáveis para Reconhecimento de Palavras e Descoberta de Padrões em Séries Temporais

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Aprovado em: 23/07/2018.

BANCA EXAMINADORA

Profa. Dra. Judith Kelner
Centro de Informática / UFPE

Prof. Dr. Carmelo Jose Albanez Bastos Filho
Escola Politécnica de Pernambuco / UPE

Prof. Dr. Hansenclever de França Bassani
Centro de Informática / UFPE
(Orientador)

Dedico esta dissertação primeiramente a Deus que me dá a sabedoria que preciso para tal, a minha família, amigos e professores que me deram todo o suporte para chegar até aqui.

AGRADECIMENTOS

Agradeço primeiramente a Deus, que me permitiu e me deu sabedoria para chegar até aqui e realizar esta pesquisa. Em segundo lugar, a minha esposa, meus pais, a minha família, que sempre me ajudou de todas as maneira possíveis. Ao meu orientador, Hansenclever, que sempre me deu todo o suporte acadêmico. Por fim, aos amigos de laboratório que geralmente fizeram parte dos momentos de *insight* na pesquisa.

RESUMO

Segmentar palavras - identificar palavras em fala contínua - é um passo fundamental na aquisição de linguagem e, especialmente, no desenvolvimento léxico. No entanto, a ausência de silêncios sistemáticos (pausas) entre as palavras em fala contínua torna esta tarefa mais difícil, especialmente no início do desenvolvimento humano, em que outros componentes-chave da linguagem (fonologia, morfologia, sintaxe, etc.) ainda não são totalmente conhecidos pelos bebês. Este processo de aquisição de linguagem em bebês com poucos meses de vida também pode ser descrito como um processo de descoberta de padrões recorrentes (*Motif*). Três características principais na tarefa de aquisição de linguagem realizada pelas crianças são observadas e trazidas para o contexto computacional neste trabalho. Os Mapas Auto-Organizáveis (SOM), são muito relevantes para o contexto desta Dissertação, por serem modelos neurais biologicamente inspirados, utilizando aprendizagem não-supervisionada e incremental, que são capazes de produzir como resultado protótipos que podem ser considerados como padrões frequentes em uma série temporal. Todavia, o SOM possui algumas limitações que inviabilizam a realização da tarefa de reconhecer entradas com tamanhos variáveis, como por exemplo, o SOM não implementa dimensões que possuem diferentes relevâncias e a não está apto a ajustar a quantidade de protótipos no mapa ao longo do tempo. Alguns modelos baseados em SOM, apresentam características relevantes para o desenvolvimento desta Dissertação, sendo portanto apresentados no decorrer deste trabalho. Assim, esta Dissertação possui como objetivo criar um modelo, que seja capaz de aprender padrões de tamanhos variáveis e desconhecidos. Partindo de modelos inspirados em SOM, este trabalho propõe uma série de modificações para permitir que o modelo consiga se adaptar a padrões de entrada de tamanhos desconhecidos e variados, assim como os bebês aprendem. No decorrer deste trabalho, estão apresentados experimentos utilizando-se de dados reais e de dados simulados. Ao fim desta Dissertação, os resultados estão apresentados e assim, é possível mostrar que o modelo proposto apresenta a maioria das características desejadas.

Palavras-chave: Mapas Auto-Organizáveis. Tamanho de Entrada Variável. Agrupamento em Subespaços. Descoberta de Padrões. Reconhecimento de Palavras. Segmentação de Palavras.

ABSTRACT

Word Segmentation - identify words in continuous speech - is a fundamental step in the acquisition of language and especially in lexical development. However, the deficiency of systematic silences (pauses) between words in continuous speech makes this task more difficult, especially at the beginning of human development, in which other key components of language (phonology, morphology, syntax, etc.) are not fully known to babies. This process of language acquisition in infants with a few months of life can also be described as a process of discovering recurrent patterns (Motif). The Self-Organizing Maps (SOM), are very relevant to the context of this Dissertation because they are neural models biologically inspired using non-supervised and incremental learning, that are able of producing prototypes that can be considered as frequent patterns in a time series. However, the SOM has some limitations that make it impossible to perform the task of recognizing inputs with variable sizes, for example, SOM does not implement dimensions that have different relevancies and is not able to adjust the number of prototypes on the map over time. Some models based on SOM, present characteristics relevant to the development of this Dissertation, and are therefore presented in the course of this work. Thus, this Dissertation aims to create a model that is capable of learning patterns of variable and unknown sizes. Based on SOM-inspired models, this work proposes a series of modifications to allow the model to adapt to input patterns of unknown and variable sizes. In the course of this work, experiments will be presented using real data and simulated data. At the end of this Dissertation, the results will be presented and then the final considerations. Thus, it will be possible to show that the proposed model has the desired characteristics.

Keywords: Self-Organizing Maps. Variable Input Length. Subspace Clustering. Motif Discovery. Word Recognition. Word Segmentation.

LISTA DE ILUSTRAÇÕES

Figura 1	– Latência média para mudar o olhar de uma imagem de distração para a imagem alvo é mostrada para crianças de diferentes idades (JUSCZYK, 1999).	16
Figura 2	– Na Figura (a) pode ser visto uma Série Temporal. E na Figura (b), pode-se verificar um exemplo de <i>Motifs</i> encontrados em na Série Temporal mostrada na Figura (a). L é um tamanho que é especulado manualmente e procurado por toda a extensão da série temporal. No exemplo, existem padrões de dois tamanhos diferentes (YINGCHARE-ONTHAWORNCHAI et al., 2013).	17
Figura 3	– Exemplo de Série Temporal. Histórico do valor do Bitcoin ao longo do tempo (TAR, 2017).	23
Figura 4	– Gravação de fala da sílaba aaa hhh amostrada em 10.000 pontos por segundo com $n = 1020$ pontos (SHUMWAY; STOFFER, 2000).	23
Figura 5	– Uma série temporal astronômica (parte de cima) contém 3 subsequências semelhantes. Uma aproximação na série temporal (parte de baixo) revela quão similares são as 3 subsequências (LIN et al., 2002).	24
Figura 6	– Estrutura básica de um SOM. Onde \mathbf{x} é o padrão de entrada, \mathbf{x}_i são os valores do nodo da i -ésima camada de entrada. O \mathbf{w}_{ji} , representa o peso entre o j -ésimo nodo na camada de saída (camada de organização) com o i -ésimo nodo na camada de entrada. Nesta configuração, cada nó na camada de saída é conectado diretamente com quatro vizinhos na grade retangular. Os tons de azul significa a influencia que um nodo vencedor (tonalidade mais escura) tem na atualização de sua vizinhança.	26
Figura 7	– Mapa gerado após o experimento com comandos de máquinas com palavras finlandesas executado por (KOHONEN; SUMERVUO, 1997).	28
Figura 8	– Exemplo que ilustra a atualização da dimensão do nodo vencedor sendo atualizada. Estão ilustrados o vetor de protótipos e o vetor de relevâncias.	35
Figura 9	– Exemplo de como a ativação é calculada em um conjunto de dados binário para um padrão de entrada com um comprimento menor que o nodo.	36
Figura 10	– Exemplo que ilustra a atualização da dimensão dos vetores do nodo.	36
Figura 11	– Todos os valores gerados pelo LHS para os 100 conjuntos de parâmetros. Com isso, pode-se observar que toda a região do intervalo definido foi ocupada pelos pontos.	40
Figura 12	– Ilustra o cálculo, de maneira intuitiva, da medida de precisão e da medida de revocação (Fonte: Wikipédia, acessado em Junho/2018).	42

Figura 13 – Primeira e segunda dimensões das quatro classes geradas para serem verdadeiros positivos.	43
Figura 14 – Primeira e segunda dimensões das quatro classes geradas para serem falsos positivos.	44
Figura 15 – Visualização da primeira e segunda dimensões da mistura de dados gerados para treinamento e teste do modelo.	45
Figura 16 – Primeira sequência de treinamentos e testes. Sequência executada com o objetivo de verificar a capacidade do modelo em aprender padrões de tamanhos variáveis.	45
Figura 17 – Precisão, Medida-f e Revocação resultantes do experimento com o melhor conjunto de parâmetros encontrado.	46
Figura 18 – Segunda sequência de treinamentos e testes. Sequência executada com o objetivo de verificar a capacidade do modelo em aprender padrões de tamanhos crescentes sem esquecer os padrões de tamanhos menores.	47
Figura 19 – Resultados do experimento com o melhor conjunto de parâmetros encontrado usando as duas formas de treinamento e teste (a) e (b). O objetivo, neste caso, foi verificar o esquecimento catastrófico gerado pelo modelo.	48
Figura 20 – Curva da porcentagem de recordação do modelo. Por exemplo, depois que a rede foi treinada até o tamanho 100, se for apresentada ao <i>Variable Input Length Map</i> (VILMAP) padrões de tamanho 70 é possível que 89,3% desses padrões de tamanho 70 ainda sejam reconhecidos/recordados.	49
Figura 21 – Exemplo da base de dados GunPoint (RATANAMAHATANA; KEOGH, 2005).	50
Figura 22 – Resultado para a primeira classe da base de dados GunPoint, executados com o modelo VILMAP.	51
Figura 23 – Resultado para a segunda classe da base de dados GunPoint, executados com o modelo VILMAP.	52
Figura 24 – Resultado para a segunda classe da base de dados GunPoint, executados com o modelo LARFDSSOM.	53
Figura 25 – Resultado para a segunda classe da base de dados GunPoint, executados com o modelo LARFDSSOM.	53
Figura 26 – Tradução das palavras em características. Exemplo que mostra o tamanho da janela de tempo igual a 4 fonemas, logo, o tamanho mínimo da entrada do modelo será de 48, já que as características de cada fonema são concatenadas em um vetor único que serve como entrada para a rede (BASSANI, 2014).	55

Figura 27 – Exemplo de tradução de uma palavra até o seu respectivo vetor de características, que neste caso, possui tamanho 60.	56
Figura 28 – Agrupamento feito sem o deslocamento do VILMAP. Na primeira coluna, estão os estímulos, traduzidos para fonemas, recebidos pelo mapa; na segunda coluna, estão os códigos identificadores do nodo para o mapa; na terceira coluna, está o resultado do cálculo da ativação entre o estímulo e o nodo (na mesma linha); e na colunas seguintes, estão: acima as relevâncias e abaixo o protótipo traduzido para fonemas. . . .	57
Figura 29 – Agrupamento feito com o deslocamento do VILMAP. Neste caso, está em negrito, no protótipo, em qual parte do nodo a entrada foi reconhecida.	57
Figura 30 – Vizinhança do nodo localizado no centro da figura.	59

LISTA DE TABELAS

Tabela 1 – Alguns comandos utilizados por (KOHONEN; SUMERVUO, 1997) no experimento com sequências de fonemas.	28
Tabela 2 – Intervalo de parâmetros VILMAP	41
Tabela 3 – Exemplos de palavras da língua inglesa traduzidas para fonemas utilizando o dicionário fonético disponibilizado pela Carnegie Mellon (CMU, 2011).	55
Tabela 4 – Resultados de Medida-F, precisão e revocação para o experimento de segmentação de palavras.	59

LISTA DE ABREVIATURAS E SIGLAS

AGu	<i>Unigram Adapter Grammar</i>
DiBS	<i>Diphone-Based Segmentation</i>
DSSOM	<i>Dimension Selective Self-Organizing Map</i>
LARFDSSOM	<i>Local Adaptive Receptive Field Dimension Selective Self-organizing Map</i>
LHS	<i>Latin Hypercube Sampling</i>
PUDDLE	<i>Phonotactics from Utterances Determine Distributional Lexical Elements</i>
SOM	<i>Self Organized Map</i>
ST	<i>Série Temporal</i>
TPs	<i>Transitional Probabilities</i>
TSMD	<i>Time Series Motif Discovery</i>
VILMAP	<i>Variable Input Length Map</i>
VLMD	<i>Variable-Length Motif Discovery</i>

SUMÁRIO

1	INTRODUÇÃO	15
2	FUNDAMENTAÇÃO TEÓRICA	20
2.1	SEGMENTAÇÃO DE PALAVRAS	20
2.1.1	DiBS	20
2.1.2	TPs	21
2.1.3	PUDDLE	21
2.1.4	AGu	22
2.2	SÉRIES TEMPORAIS	22
2.2.1	Descoberta de Padrões Previamente Desconhecidos	24
2.3	MAPAS AUTO-ORGANIZÁVEIS - SOM	25
2.3.1	Mapas Auto-Organizáveis Adaptativos com Dimensões Relevantes	29
3	VARIABLE INPUT LENGTH MAP - VILMAP	32
3.1	COMPETIÇÃO E INSERÇÃO DOS NODOS	35
3.2	ATUALIZAÇÃO DO NODO VENCEDOR E DA SUA VIZINHANÇA	37
3.3	AGRUPAMENTO COM O VILMAP	37
4	VALIDAÇÃO DO MODELO	39
4.1	AJUSTE PARAMÉTRICO DO VILMAP	39
4.2	MÉTRICAS	41
4.3	EXPERIMENTAÇÃO COM DADOS SIMULADOS	42
4.3.1	Bases de Dados Simulados	42
4.3.2	Reconhecimento de Padrões de Tamanhos Diferentes - Experimento 1	45
4.3.3	Reconhecimento de Padrões com Tamanhos Crescentes sem Esquecer os Padrões Menores Previamente Aprendidos - Experimento 2	46
4.4	EXPERIMENTAÇÃO COM DADOS REAIS	49
4.4.1	Descoberta de Padrões Previamente Desconhecidos em Séries Temporais - Experimento 3	50
4.4.1.1	Bases de Dados	50
4.4.1.2	Análise dos resultados	50
4.4.2	Aprendizagem de Sequência de Fonemas de Tamanhos Variáveis - Experimento 4	54
4.4.2.1	Bases de Dados	54
4.4.2.2	Análise dos resultados	56

4.4.3	Reconhecimento de Palavras - Experimento 5	57
4.4.3.1	Bases de Dados	58
4.4.3.2	Análise dos resultados	58
4.4.4	Coerência Topológica do Mapa - Nodos Vizinhos são Semelhantes	
	- Experimento 6	59
5	CONSIDERAÇÕES FINAIS	61
5.1	ANÁLISE DA MODELAGEM PROPOSTA	61
5.2	CONTRIBUIÇÕES PARA A CIÊNCIA	62
5.2.1	Trabalhos Publicados	62
5.3	LIMITAÇÕES DO MODELO	62
5.4	TRABALHOS FUTUROS E APLICAÇÕES	62
	REFERÊNCIAS	63

1 INTRODUÇÃO

Segmentar palavras - identificar palavras em fala contínua - é um passo fundamental na aquisição de linguagem e, especialmente, no desenvolvimento léxico. No entanto, a ausência de silêncios sistemáticos (pausas) entre palavras em fala contínua torna esta tarefa difícil, especialmente no início do desenvolvimento humano, em que outros componentes-chave da linguagem (fonologia, morfologia, sintaxe, etc.) ainda não são totalmente conhecidos pelos bebês.

Alguns trabalhos abordam uma questão-chave de pesquisa que é: como os bebês obtêm a segmentação de palavras do zero? Um deles é o (RÄSÄNEN, 2012) que revisa uma série de estudos computacionais existentes concentrados na questão de como a linguagem falada pode ser aprendida a partir da fala contínua na ausência de conhecimento prévio, linguístico ou foneticamente motivado, uma situação enfrentada por bebês humanos no momento em que eles tentam aprender sua língua nativa.

Três características principais na tarefa de aquisição de linguagem realizada pelas crianças podem ser observadas e trazidas para o contexto computacional (CORRÊA, 2007):

- **Aprendizado Online** - As crianças podem aprender palavras a qualquer momento, ou seja, o processo de aprendizagem acontece a medida que as crianças vão escutando e tentando reconhecer as palavras;
- **Aprendizado Não-supervisionado** - As crianças ajustam todos os estímulos causados pela língua sem receber nenhuma supervisão externa ao seu corpo;
- **Aprendizado Incremental** - As crianças aprendem as palavras/fonemas da língua de forma crescente, ou seja, aprendem primeiro palavras menores e aos poucos o nível de dificuldade e tamanhos das palavras aprendidas vão aumentando.

Em um estudo realizado por (JUSCZYK, 1999) familiarizou as crianças, de 6 meses e 7,5 meses de vida, por 30s com cada par de palavras, como *pés e bicicleta*, ou *xícara e cachorro*, para determinar quando os bebês que aprendem inglês podem começar a segmentar palavras em fala fluente. O objetivo da familiarização era preparar os bebês para responder às palavras-alvo que apareceriam posteriormente em algumas passagens de fala fluentes. Um teste realizado ainda por (JUSCZYK, 1999) tem o seu resultado mostrado na Figura 1 que teve como objetivo mensurar o aprendizado e a associação de sons com os significados. Depois de ouvir as palavras-alvo, os bebês foram expostos a duas fotos: uma associada a essa palavra-alvo e uma foto para causar distração. Na Figura 1, a latência média para mudar o olhar de uma imagem de distração para a imagem alvo é mostrada para crianças de diferentes idades. As crianças mais velhas foram mais rápidas em entender as palavras que ouviram.

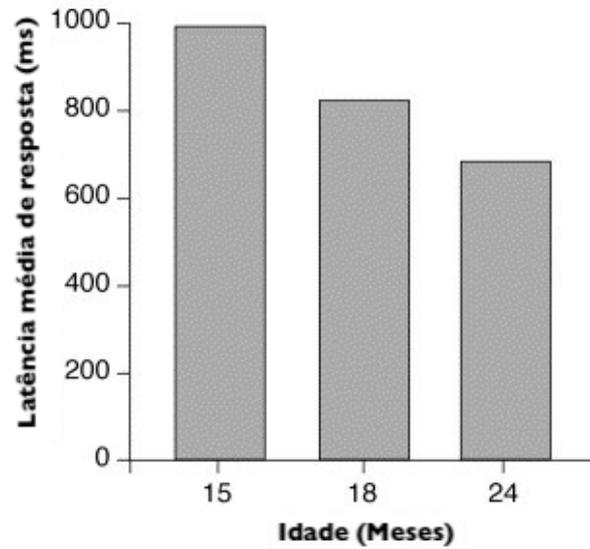


Figura 1 – Latência média para mudar o olhar de uma imagem de distração para a imagem alvo é mostrada para crianças de diferentes idades (JUSCZYK, 1999).

O processo de aquisição de linguagem em bebês com poucos meses de vida pode ser descrito como uma Descoberta de padrões recorrentes. Na Figura 2 (b), são apresentados dois exemplos de padrões encontrados em uma série temporal, apresentada em (a), onde L é o tamanho da janela de busca que tem o objetivo de delimitar o tamanho dos padrões frequentes. No exemplo da Figura 2 foram encontrados *Motifs* de tamanho $L = 163$ e $L = 286$.

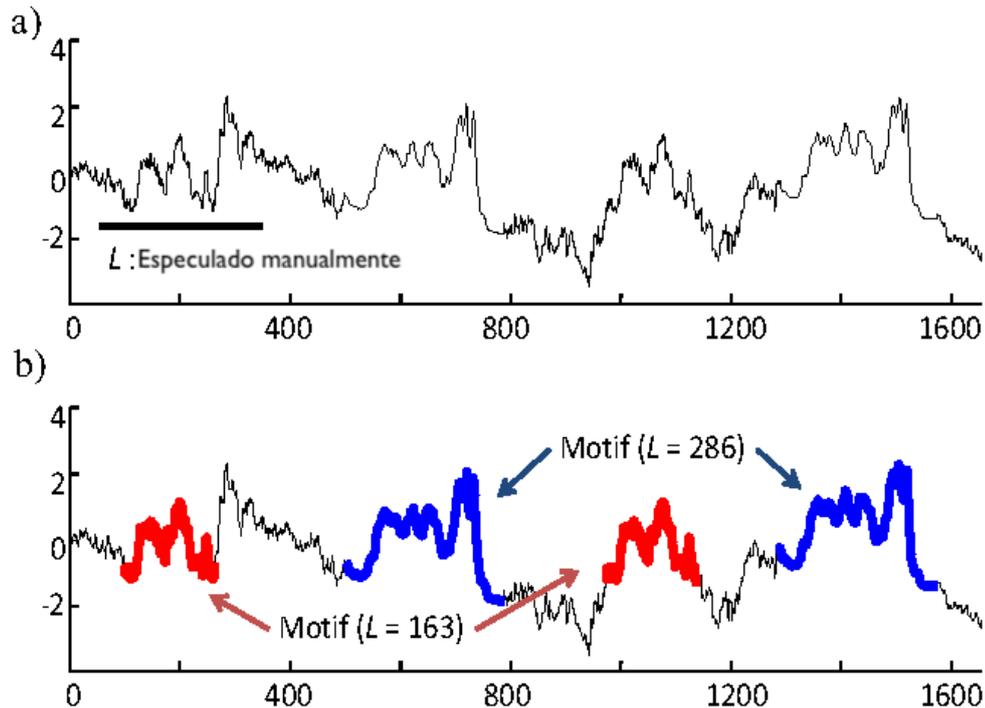


Figura 2 – Na Figura (a) pode ser visto uma Série Temporal. E na Figura (b), pode-se verificar um exemplo de *Motifs* encontrados em na Série Temporal mostrada na Figura (a). L é um tamanho que é especulado manualmente e procurado por toda a extensão da série temporal. No exemplo, existem padrões de dois tamanhos diferentes (YINGCHAREONTHAWORNCHAI et al., 2013).

Os *Motifs* são descritos na literatura como padrões recorrentes, tendências frequentes, sucessões, formas ou subsequências frequentes que ocorrem em séries temporais (TORKAMANI; LOHWEG, 2017). Os métodos de descoberta de *Motifs* pesquisam padrões frequentes, anteriormente desconhecidos, em uma série temporal (LIN et al., 2002). A tarefa pode ser vista como um problema de agrupamento de séries temporais, supondo que cada grupo deve reunir padrões semelhantes encontrados na série temporal que, portanto, representam o mesmo *Motif*. O problema da Segmentação de Palavras em fala fluente pode ser visto como um problema de descoberta de *Motifs*, onde as palavras são os *Motifs* de uma série temporal composta pela informação de áudio bruta, por características extraídas do áudio, ou mesmo por uma representação dos fonemas contidos no som. Este problema tem recebido muita atenção da comunidade que estuda a aquisição precoce de linguagem em bebês e crianças pequenas e é um dos focos do presente estudo. Assim, o desenvolvimento de modelos biologicamente plausíveis para a segmentação de palavras poderia avançar significativamente o desenvolvimento desta área.

Modelos de redes neurais têm sido utilizados para processamento sensorial. Um dos modelos utilizados com este objetivo são os Mapas Auto-organizáveis (SOMs) (KOHONEN, 1990). O reconhecimento de fala baseado em unidades fonéticas é capaz de produzir sequências de fonemas como saída. Estes, geralmente não são idênticos aos modelos de pronúncia encontrados em um dicionário convencional e devido aos efeitos de co-articulação,

os fonemas dependem de seus contextos (KOHONEN; SUMERVUO, 1997).

O Mapa Auto-Organizável (SOM), definido por (KOHONEN, 1998), é uma rede neural biologicamente inspirada, frequentemente usada para visualização de dados de alta dimensão por ser capaz de comprimir a informação, preservando as relações topológicas e capturando as características mais importantes dos dados de entrada. No entanto, é possível encontrar aplicações em muitos outros tipos de problemas, como reconstrução de superfície em (REGO; ARAUJO; NETO, 2007) e agrupamento de dados em (BASSANI; ARAUJO, 2015). Também vale a pena mencionar que o SOM tem sido aplicado a uma variedade de problemas que envolvem o processamento sensorial, incluindo o visual mostrado em (MIKKULAINEN et al., 2005) e a informação auditiva apresentado em (KANGAS, 1991).

O SOM original, apresentado por (KOHONEN, 1982), define um mapa composto por um conjunto de nodos, ou protótipos, que competem e cooperam para representar uma determinada região do espaço de entrada. Os nodos são geralmente organizados em uma grade bidimensional fixa e o modelo emprega uma distância euclidiana para comparar os padrões de entrada com cada nodo no mapa, o que pode não ser adequado para agrupar conjuntos de dados de alta dimensão, devido à maldição da dimensionalidade, explicada em (KEOGH; MUEEN, 2011), ou em conjuntos de dados nos quais nem todas as dimensões são igualmente relevantes para os diferentes grupos, como no agrupamento do subespaço (PARSONS; HAQUE; LIU, 2004; KRIEGEL; KRÖGER; ZIMEK, 2009) e na descoberta de *Motifs* (LIN et al., 2002). No entanto, novos modelos baseados em SOM foram desenvolvidos para melhorar seu desempenho em casos que precisam ser tratados utilizando a aplicação de relevâncias distintas em cada dimensão de entrada ou que precisam ter topologia variante no tempo como os apresentados em (ARAÚJO; COSTA, 2009; BASSANI; ARAÚJO, 2012; BASSANI; ARAUJO, 2015).

Assim, o objetivo geral desta Dissertação é desenvolver um modelo, biologicamente inspirado, que aprenda a reconhecer palavras com características semelhantes as da forma como os bebês aprendem. Portanto, o *Variable Input Length MAP* (VILMAP) será apresentado no Capítulo 3 assumindo como objetivos os seguintes tópicos:

Objetivos Específicos

- Viabilizar o agrupamento de padrões de entrada de tamanhos diferentes;
 - Ser capaz de ajustar a relevância de cada uma das dimensões para permitir que as amostras de entrada sejam variáveis em relação ao número de dimensões de entrada;
- Reconhecer padrões com tamanhos crescentes sem esquecer os padrões menores previamente aprendidos;
- Realizar um agrupamento topológico, de forma que padrões semelhantes sejam agrupados em nodos próximos no mapa;

Nos experimentos apresentados no Capítulo 5, é mostrado que o VILMAP é capaz de encontrar padrões em um conjunto de dados de descoberta *Motifs* (o conjunto de dados GunPoint (CHEN et al., 2015)). Além disso, é verificado se o VILMAP evita o esquecimento catastrófico (ou interferência catastrófica), ou seja, sem esquecer os padrões aprendidos anteriormente (MCCLOSKEY; COHEN, 1989), quando treinado com conjuntos de dados com tamanhos de entrada crescentes. Por fim, é avaliado como o VILMAP se sai na tarefa referente a segmentação de palavras em comparação com outros métodos descritos na literatura, desenvolvida para a tarefa de segmentação de palavras.

O restante desta dissertação está organizado da seguinte maneira: o Capítulo 2 apresenta a fundamentação teórica necessária para o desenvolvimento deste modelo (VILMAP); o Capítulo 3 descreve o modelo proposto; o Capítulo 4 apresenta a configuração experimental e os resultados obtidos; e finalmente, no Capítulo 5, são apresentadas as considerações finais.

2 FUNDAMENTAÇÃO TEÓRICA

No presente capítulo, estão abordados alguns conceitos com o objetivo de criar um alicerce teórico para contextualizar a pesquisa. Na Seção 2.1, é introduzido o problema da segmentação de palavras. Logo após, na Seção 2.2, está abordado o tema referente as Séries Temporais e também é apresentado o problema referente a descoberta de padrões previamente desconhecidos. Na Seção 2.3, é explicado o que são os Mapas Auto-Organizáveis e como eles funcionam. E por fim, na Seção 2.3.1, é explicado o modelo *Local Adaptive Receptive Field Dimension Selective Self-organizing Map* (LARFDSSOM).

2.1 SEGMENTAÇÃO DE PALAVRAS

Ao contrário do espaçamento existente entre as palavras em um texto escrito, a fala raramente é delimitada por pausas; portanto, as crianças precisam aprender de alguma forma a identificar os limites entre as palavras enquanto ouvem as frases. Como a estrutura das palavras é significativamente variável em todos os idiomas, é difícil saber como uma criança entre 9 e 15 meses consegue adquirir essa capacidade de identificar os limites entre as palavras. Portanto, a segmentação é um passo fundamental na aquisição da linguagem, especificamente no desenvolvimento léxico (SAFFRAN; NEWPORT; ASLIN, 1996). Vale ressaltar que encontrar os limites exatos de todas as palavras em uma frase não é estritamente necessário para que as crianças entendam o que é falado por alguma pessoa. O que as crianças precisam é realmente reconhecer as palavras presentes na frase (palavras simples ou compostas) na ordem correta, o que pode ser feito mesmo que existam certos erros na detecção dos limites das palavras, como pode ser observado em crianças pequenas (CORREA; DOCKRELL, 2007).

Na presente pesquisa, os experimentos descritos por (LARSEN; CRISTIA; DUPOUX, 2017) foram reproduzidos e comparados com os resultados obtidos pelos quatro algoritmos de segmentação de palavras apresentados a seguir:

2.1.1 DiBS

O primeiro modelo usa pistas estatísticas locais e é o algoritmo *Diphone-Based Segmentation* (DiBS) (DALAND; PIERREHUMBERT, 2011), baseado em propriedades fonéticas, que mantém na memória a frequência de difones (dois fonemas) ocorrendo juntos e coloca um limite entre eles calculando o Teorema de Bayes. Para tanto, o modelo possui várias premissas: o aprendiz conhece as categorias da fonética, é capaz de detectar limites de expressão, assume independência fonológica através de limites de palavras, rastreia a distribuição de difones sem contexto e conhece a frequência relativa das formas das palavras já aprendidas. Todas essas suposições são discutidas em (DALAND; PIERREHUMBERT,

2011) a partir de um ponto de vista psicolinguístico, indicando por que é plausível que os bebês possam atuar como aprendizes do DiBS ao iniciar a segmentação da fala.

O único parâmetro livre do modelo é a probabilidade livre de contexto da fronteira da palavra que é determinada pelo comprimento médio das palavras e pelo número de palavras por frase. Essa configuração é bastante controversa, uma vez que o modelo é parcialmente supervisionado - os limites entre as palavras são dados a um subconjunto da base de dados para ajustar a probabilidade livre de contexto de um limite de palavras - enquanto a segmentação infantil não é parcialmente supervisionada (DALAND; PIERREHUMBERT, 2011).

2.1.2 TPs

Outro modelo estatístico local que é baseado no rastreamento de probabilidades transacionais, chamado de *Transitional Probabilities* (TPs), postula uma ligação entre duas sílabas se a probabilidade de coocorrência for localmente mais baixa (limiar relativo) ou se for menor que um limiar absoluto geralmente calculado tomando o valor médio de pares de sílabas (SAFFRAN; ASLIN; NEWPORT, 1996). Como é altamente improvável que os bebês calculem esse limiar absoluto, o relativo é escolhido para ser utilizado, o que também implica que as palavras monossílabas só podem ser encontradas na extremidade do enunciado.

Em comparação com o DiBS, os TPs exigem uma memória maior, pois o número de todas as sílabas possíveis encontradas é muito maior do que o número de todos os fonemas possíveis. Nesse sentido, os TPs se apresentam menos cognitivamente plausíveis. No entanto, vários estudos experimentais usando linguagens artificiais sugeriram que as crianças podem montar sequências estatísticas altas de sílabas e ligá-las com sequências estatísticas mais baixas (SAFFRAN; ASLIN; NEWPORT, 1996). A estratégia baseada em léxico requer que os aprendizes tenham uma memória maior e um conhecimento mais amplo sobre sua linguagem do que estratégias estatísticas locais.

2.1.3 PUDDLE

O algoritmo *Phonotactics from Utterances Determine Distributional Lexical Elements* (PUDDLE) (MONAGHAN; CHRISTIANSEN, 2010), constrói de forma incremental um léxico usando informações sobre limites de expressão e deduzindo restrições fonéticas. Mais precisamente, cada vez que uma sequência de fonemas é encontrada, se uma correspondência com uma palavra no proto-léxico é encontrada e se as restrições fonéticas são respeitadas, então o pedaço de fonemas é adicionado ao proto-léxico, os pares de fonemas de início e fim são adicionados, respectivamente, a uma lista feita de pares de fonemas iniciais e finais encontrados anteriormente. Assim, as restrições fonéticas são as seguintes: quando uma correspondência é encontrada, o par de fonemas anteriores deve pertencer à lista de pares de fonemas finais e o par de fonemas a seguir deve pertencer à lista de pares

de início. Essas restrições evitam a segmentação excessiva ao nível dos fonemas e tornam o proto-léxico mais próximo do léxico real da base.

Além disso, o modelo promove palavras frequentes contando as ocorrências de palavras adicionadas no proto-léxico e, em seguida, classificando a lista dessas palavras por frequência. Para o PUDDLE, como os requisitos de tempo (número de operações matemáticas) e espaço (memória necessária) eram pesados, foi criado por (LARSEN; CRISTIA; DUPOUX, 2017) uma versão em python usando os módulos de coleção, resultando em uma redução de 120 vezes no tempo de computação e otimização uso do espaço.

2.1.4 AGu

O último algoritmo usado é o *Unigram Adapter Grammar* (AGu) (JOHNSON; GRIFFITHS; GOLDWATER, 2007), que modela um aprendiz ideal, ou seja, um aprendiz com uma memória infinita e um processo em lote (*batch*), analisando todo a base de dados antes de segmentar. A estrutura consiste em dois módulos: um gerador de léxico - palavras existentes em uma determinada língua - e um adaptador - verificando a frequência dos itens. O primeiro gera um léxico de itens que podem ser encontrados na base e o segundo atribui as frequências dos itens. É importante ressaltar que o AGu assume que os itens do léxico são gerados independentemente uns dos outros e que o processo estocástico é escolhido de modo que as frequências dos itens sigam uma distribuição da lei de potência, como é feito na linguagem natural.

Uma sequência de palavras, apresentadas a qualquer um desses algoritmos citados na presente seção, pode ser considerada como uma Série Temporal. Por isso, as Séries Temporais serão definidas e exemplificadas na próxima seção.

2.2 SÉRIES TEMPORAIS

Uma Série Temporal (ST) é um conjunto de observações s_t que são capturadas em um instante de tempo específico t . Uma ST \mathbf{S} pode ser descrita por uma lista $\mathbf{S} = (s_1, s_2, \dots, s_n)$, onde n é o tamanho total da série \mathbf{S} , ou seja, a quantidade de pontos existentes na série, conforme Brockwell e Davis (2013).

Um exemplo de ST pode ser visto na Figura 3, que é o histórico do preço da moeda virtual Bitcoin em Dólares Americanos desde o mês de Janeiro de 2009 até o mês de Julho de 2017 da moeda virtual Bitcoin. Conforme a definição, o valor da moeda virtual ao longo do tempo forma uma Série Temporal.

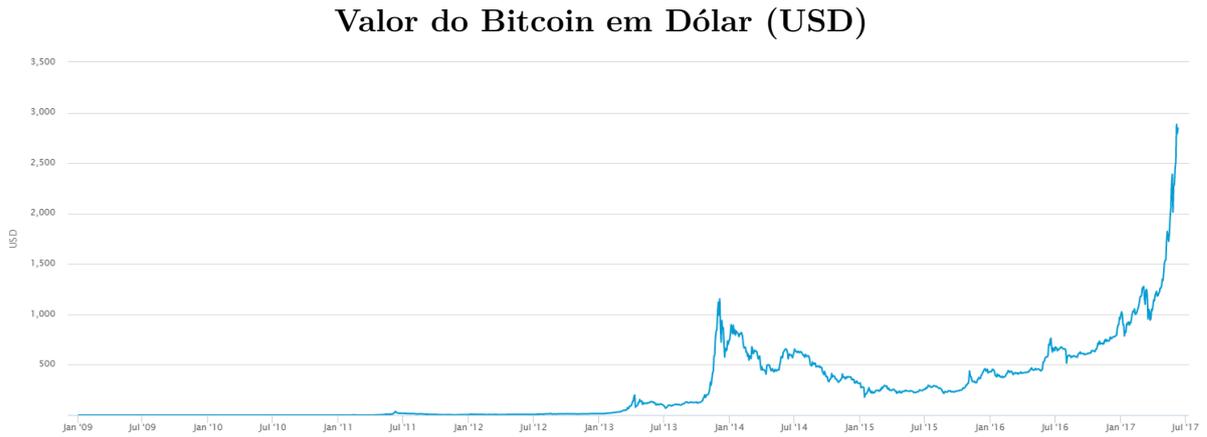


Figura 3 – Exemplo de Série Temporal. Histórico do valor do Bitcoin ao longo do tempo (TAR, 2017).

A Figura 4 mostra uma pequena amostra de 0,1 segundo (1020 pontos) de fala para a frase aaa . . . hhh, e é possível visualizar a natureza repetitiva do sinal e as periodicidades bastante regulares. Um problema atual de grande interesse é o reconhecimento de computador da fala, o que exigiria a conversão deste sinal particular para a frase gravada aaa . . . hhh (SHUMWAY; STOFFER, 2000).

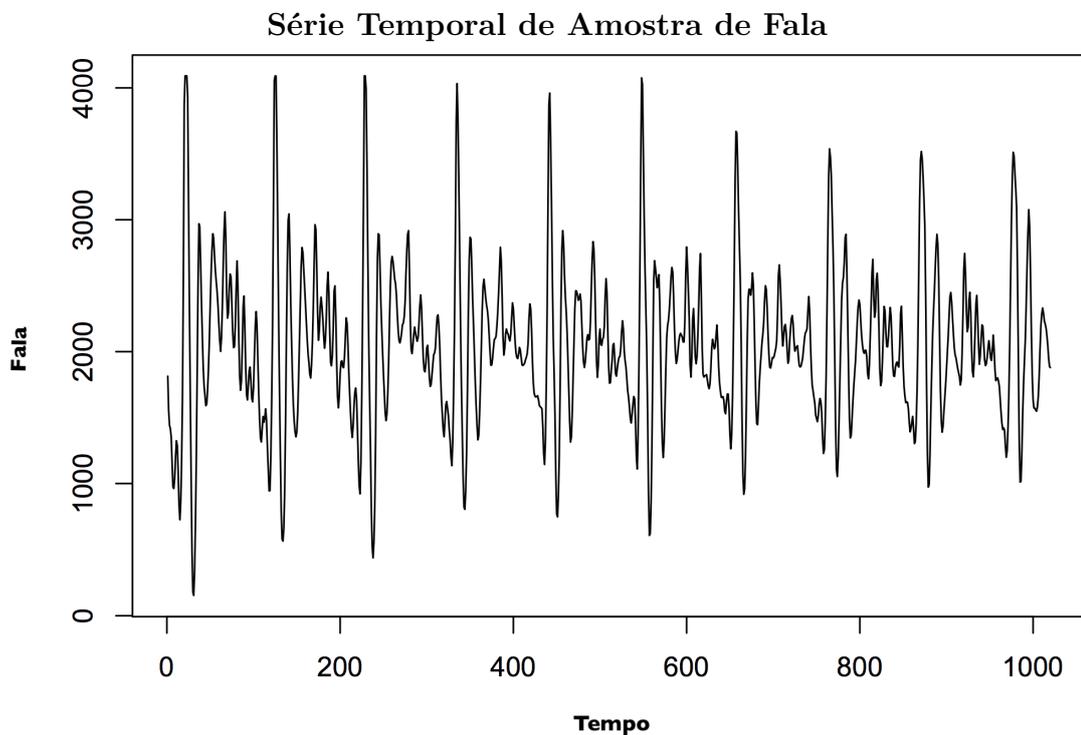


Figura 4 – Gravação de fala da sílaba aaa . . . hhh amostrada em 10.000 pontos por segundo com $n = 1020$ pontos (SHUMWAY; STOFFER, 2000).

Após a explicação sobre Séries Temporais, na próxima seção, será apresentada a descoberta de padrões previamente desconhecidos em uma ST.

2.2.1 Descoberta de Padrões Previamente Desconhecidos

São chamados de *Motifs* os padrões frequentes em uma Série Temporal, os quais são previamente desconhecidos. A busca por esses padrões é chamada de *Motif Discovery* e as buscas por padrões previamente desconhecidos podem ser chamados de *Time Series Motif Discovery* (TSMD) (LIN et al., 2002). Um exemplo clássico de descoberta de padrões previamente desconhecidos com tamanho fixo é apresentado na Figura 5, que possibilita ver uma série temporal e os seus respectivos padrões com facilidade, proporcionado por um aumento realizado realizado nas regiões onde os padrões se encontram.

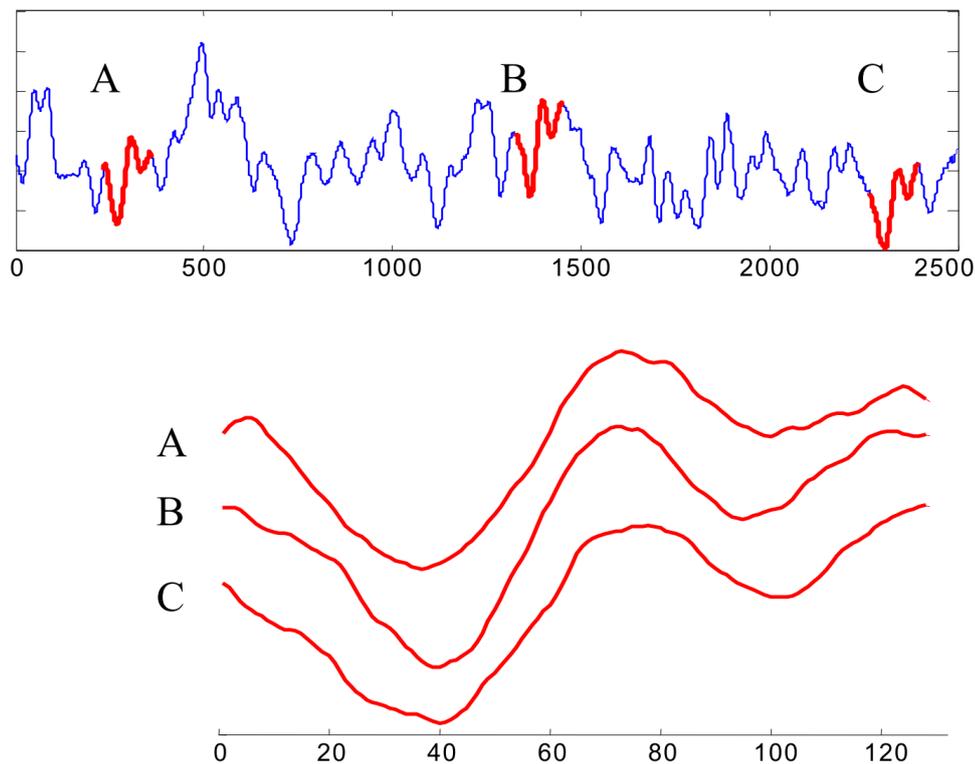


Figura 5 – Uma série temporal astronômica (parte de cima) contém 3 subsequências semelhantes. Uma aproximação na série temporal (parte de baixo) revela quão similares são as 3 subsequências (LIN et al., 2002).

Assumindo R como um limiar que estabelece uma similaridade mínima permitida ou a distância máxima permitida entre duas ocorrências de um *Motif*, são trazidas a tona duas definições principais para o problema da Descoberta de Padrões numa Série Temporal, segundo (MUEEN, 2014):

- *Definição baseada em similaridade:* Dada uma série temporal e seu comprimento, os padrões da ST são os segmentos parecidos, na ordem de suas semelhanças, respeitando o limiar de distância R .
- *Definição Baseada em Suporte:* Dada uma série temporal e seu comprimento, padrões em um ST são os segmentos que possuem o maior número de repetições

respeitando o limiar de distância R .

O método de descoberta de padrões de tamanho variável, conhecido como *Variable-Length Motif Discovery* (VLMD), (NUNTHANID; NIENNATTRAKUL; RATANAMAHATANA, 2011) é um método proposto para encontrar automaticamente um conjunto adequado de padrões de comprimento variável. O VLMD separa iterativamente padrões de tamanhos diferentes em grupos com base na sua similaridade. Dentro de um grupo, é selecionado o padrão representativo, aquele com uma distância mínima normalizada entre pares de subsequências. Assim, o VLMD retorna um conjunto de padrões representativos, um número que é extremamente pequeno em comparação com todas as possibilidades de comprimentos de janela deslizantes.

O VLMD consiste em duas etapas. Em primeiro lugar, encontra um conjunto de *Motifs* que são procurados em todos os comprimentos possíveis de uma janela deslizante para obter padrões de diferentes comprimentos. Se o padrão atual e o padrão anterior se sobrepõem, o padrão atual é adicionado ao mesmo conjunto do padrão anterior; Caso contrário, um novo grupo *Motif* será criado. Então, para cada grupo de padrões, um padrão representativo é selecionado com uma distância normalizada mínima para os outros padrões no grupo. Portanto, o VLMD pode retornar um pequeno conjunto de padrões de uma determinada sequência (ST). Não foi realizada a comparação com o método proposto por (NUNTHANID; NIENNATTRAKUL; RATANAMAHATANA, 2011) (VLMD) porque as informações no trabalho indicado não foram suficientes para reproduzir os resultados experimentais.

Apesar da existência do método VLMD, o uso dos Mapas Auto-organizáveis pode ser um caminho interessante a ser tomado para realização da tarefa de segmentação de palavras que pode ser vista como um problema de *Motif Discovery*, descrito na atual seção. Por isso, na próxima seção, os Mapas Auto-organizáveis serão descritos.

2.3 MAPAS AUTO-ORGANIZÁVEIS - SOM

O *Self Organized Map* (SOM), do inglês *Self-Organizing Map* (Figura 6), apresentado em (KOHONEN, 1990), é uma rede neural que mapeia dados de alta dimensão em uma grade menor, geralmente bidimensional, de N nodos (ou neurônios), que comprime as informações preservando as relações topológicas dos dados originais. Os nodos na grade, cuja posição é $\mathbf{c}_{i=1..N}$, participam de uma competição (*winner-takes-all* - que verifica a distância do padrão de entrada em relação a todos os nodos no mapa) para representar cada estímulo de entrada, \mathbf{x} . O nodo \mathbf{c}_i é dito vencedor, para um determinado estímulo de entrada, se possuir a menor distância Euclidiana (Equação 2.1) entre o seu vetor de pesos sinápticos (\mathbf{w}_j) e o padrão de entrada \mathbf{x} . Logo, \mathbf{c}_i é ligeiramente movido para se aproximar deste padrão. Os vizinhos do nodo vencedor na grade também são ajustados (cooperação), em uma escala menor, para estabelecer uma relação topológica que reflete

o que é observado no espaço de entrada. Após o treinamento, estímulos semelhantes são agrupados no mesmo nodo ou em nós topologicamente próximos.

$$c_i = \arg \min [D(\mathbf{x}, \mathbf{w}_j)^2], \forall j \text{ onde } j = 1..N, \quad (2.1)$$

onde, $D(\mathbf{x}, \mathbf{w}_j)$ é a distância Euclidiana entre \mathbf{x} e \mathbf{w}_j , apresentada na Equação 2.2, onde m é definido como o número de dimensões e o \mathbf{w}_{ji} , representa o peso entre o j -ésimo nodo na camada de saída (camada de organização) com o i -ésimo nodo na camada de entrada.

$$[D(\mathbf{x}, \mathbf{w}_j)]^2 = \sum_{i=1}^m (\mathbf{x}_i - \mathbf{w}_{ji})^2 \quad (2.2)$$

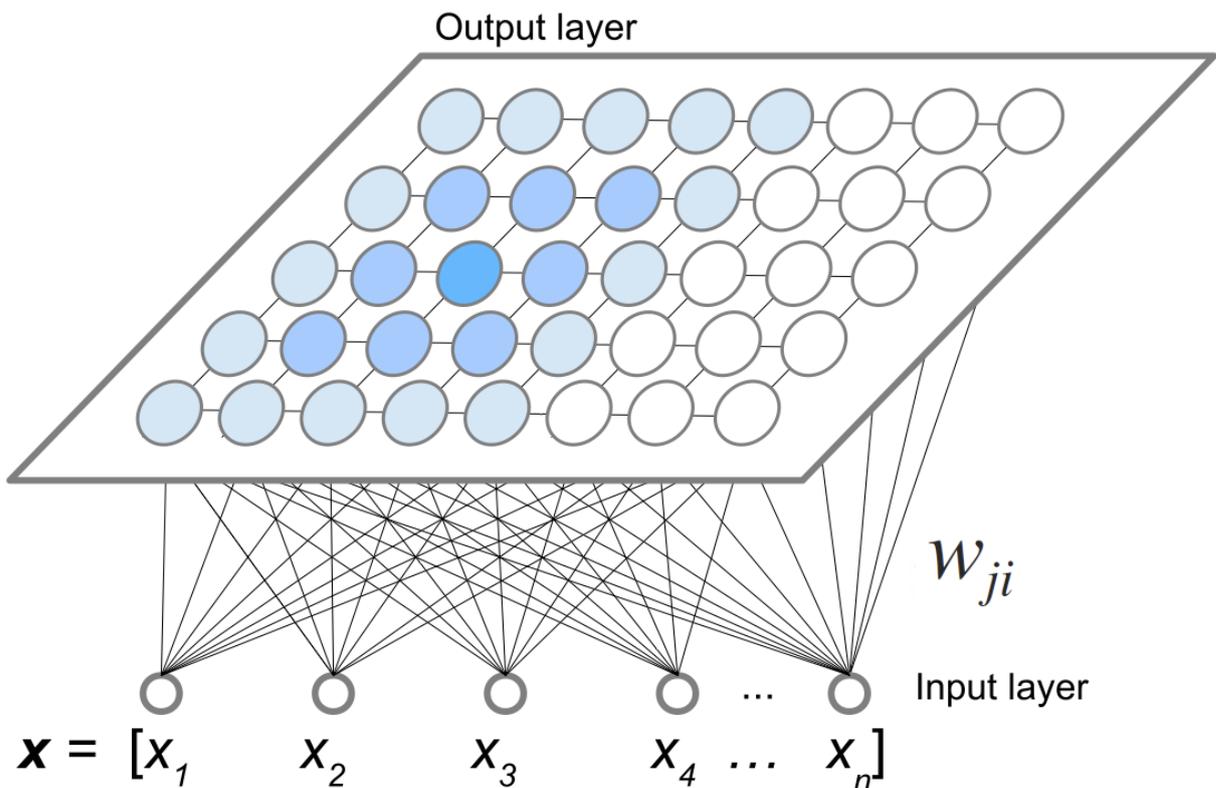


Figura 6 – Estrutura básica de um SOM. Onde \mathbf{x} é o padrão de entrada, \mathbf{x}_i são os valores do nodo da i -ésima camada de entrada. O \mathbf{w}_{ji} , representa o peso entre o j -ésimo nodo na camada de saída (camada de organização) com o i -ésimo nodo na camada de entrada. Nesta configuração, cada nó na camada de saída é conectado diretamente com quatro vizinhos na grade retangular. Os tons de azul significa a influencia que um nodo vencedor (tonalidade mais escura) tem na atualização de sua vizinhança.

O SOM possui uma estrutura básica (Figura 6) que consiste em uma camada de entrada e uma camada de saída. Na camada de entrada recebem-se os estímulos do ambiente que são propagados para todos os nodos da camada de saída. A quantidade de

nodos na camada de entrada é igual ao número de atributos presentes no vetor de entrada $\mathbf{x} = [x_1, x_2, \dots, x_m]^T$. A camada de saída codifica o mapa resultante da aprendizagem e sua topologia é geralmente um gradil bidimensional de nodos, onde, cada nodo na camada de saída é conectado aos seus vizinhos imediatos.

A auto-organização possui um requisito que prega que os pesos sinápticos de um nodo sejam modificados apenas na vizinhança local do nodo vencedor e que todos os pesos que foram modificados assemelhem-se com mais precisão ao estímulo atual do que ao estímulo passado. Diferentes sinais de entrada em diferentes momentos afetam regiões diferentes na grade de nodos. Deste modo, depois de bastante passos de aprendizagem, o mapa possui várias regiões diferentes se tornam especializadas em partes do espaço de entrada (KOHONEN, 1982).

A adaptação dos pesos é realizada como está descrito na Equação 2.3. O termo $\eta(n)$, e a função de decaimento da taxa de aprendizagem, e o termo $h_{ji(x)}(n)$ é a função de decaimento da vizinhança, que decaem a medida que se passam as iterações de treinamento. A função de decaimento da taxa de aprendizagem é apresentada na Equação 2.4 e a função de decaimento da vizinhança é apresentada na Equação 2.5.

$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n)h_{ji(x)}(n)(\mathbf{x} - \mathbf{w}_j(n)) \quad (2.3)$$

$$\eta(n) = \exp\left(-\frac{n}{\tau_2}\right), n = 0, 1, 2, \dots \quad (2.4)$$

Na Equação 2.4, que apresenta a função de decaimento da taxa de aprendizagem, existe um termo, τ_2 , que é uma constante de tempo que ajusta a velocidade do decaimento.

$$h_{ji(x)}(n) = \exp\left(-\frac{\|\mathbf{r}_j - \mathbf{r}_i\|^2}{2\sigma^2(n)}\right), n = 0, 1, 2, \dots \quad (2.5)$$

Um ajuste que é feito nos nodos vizinhos ao vencedor em direção do estímulo de entrada é dito como a etapa de cooperação entre os nodos. O nível de adaptação aplicado a cada nodo da vizinhança do vencedor apresentado pela função $h_{ji(x)}(n)$ (Equação 2.5), onde, \mathbf{r}_i e \mathbf{r}_j são as posições vetoriais do vencedor i e de seu vizinho j na grade, e $\sigma(n)$ corresponde a largura ou raio da função de vizinhança. Conforme a Equação 2.6, o valor de $\sigma(n)$ se inicia em $\sigma(0)$ e decresce monotonicamente com o decorrer das iterações. Além disso, o parâmetro τ_1 , é uma constante de tempo que ajusta a velocidade do decaimento (KOHONEN, 1982).

$$\sigma(n) = \sigma_0 \exp\left(-\frac{n}{\tau_1}\right), n = 0, 1, 2, \dots \quad (2.6)$$

Um experimento realizado em (KOHONEN; SUMERVUO, 1997) usou o SOM para sequências de fonemas. As sequências de fonemas usadas no experimento foram coletadas de 20 pessoas (15 do sexo masculino e 5 do sexo feminino). As classes de fonemas representavam 22 palavras de comando de máquinas finlandesas mostradas na Tabela 1. A Figura

7 apresenta o SOM de sequência de caracteres unitários. Os tons de cinza representam distâncias entre os vetores de referência vizinhos; mais escuros representam maiores distâncias e menores distâncias são os brancos. Com isso é possível verificar que no mapa mostrado na Figura 7 que palavras semelhantes são localizadas próximas umas das outras.

Tabela 1 – Alguns comandos utilizados por (KOHONEN; SUMERVUO, 1997) no experimento com sequências de fonemas.

YLOS	(para cima)
ALAS	(para baixo)
VASEMMALLE	(esquerda)
OIKEALLE	(direita)
SULJE	(fechar)

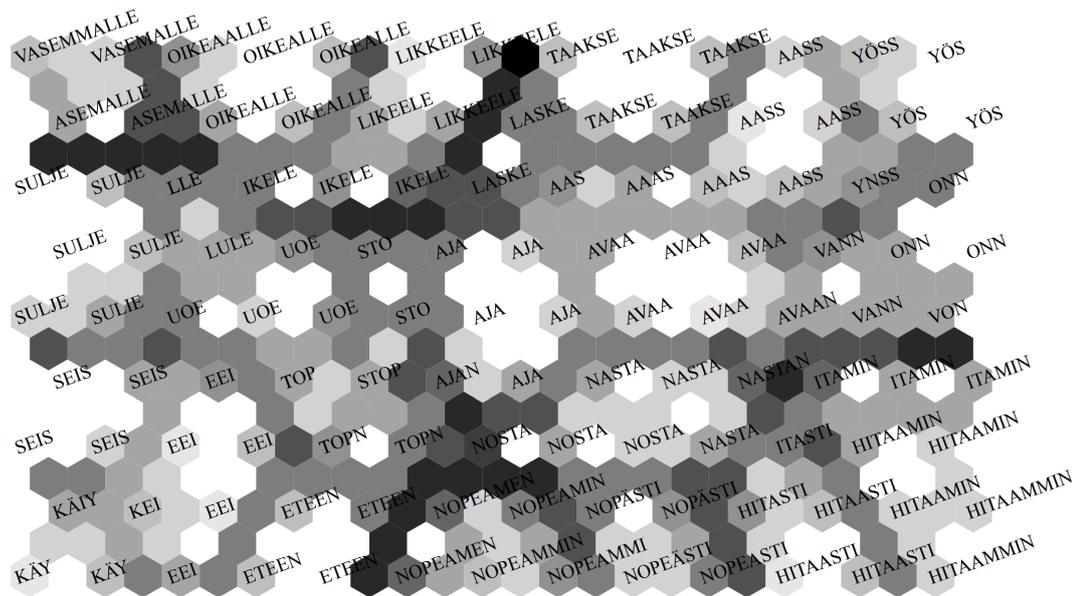


Figura 7 – Mapa gerado após o experimento com comandos de máquinas com palavras finlandesas executado por (KOHONEN; SUMERVUO, 1997).

O SOM é bastante interessante para o contexto desta Dissertação, por ser um modelo neural biologicamente inspirado, com aprendizagem não-supervisionada e incremental, que produz como resultado protótipos que podem ser considerados como padrões frequentes em uma série temporal. Todavia, o SOM possui limitações que inviabilizam a realização de algumas tarefas, como a aprendizagem a partir de entradas de tamanhos variáveis — aquelas que possuem dimensões diferentes. Algumas destas limitações do SOM estão listadas a seguir: topologia fixa, inviabilizando o ajuste topológico do mapa ao longo do tempo; utilizada a distância Euclidiana, impossibilitando que os padrões possuam importância distinta nas dimensões de entrada.

Na próxima seção, serão apresentados alguns modelos que são baseados em SOM, possuem estrutura variante no tempo e podem aplicar diferentes relevâncias para cada dimensão.

2.3.1 Mapas Auto-Organizáveis Adaptativos com Dimensões Relevantes

O *Dimension Selective Self-Organizing Map* (DSSOM), apresentado por (BASSANI; ARAÚJO, 2012), é uma extensão do SOM que apresenta uma das características que é procurada para a solução do problema desta Dissertação, pois esse modelo é um Mapa Auto-organizável seletivo a dimensões, ou seja, capaz de ajustar a relevância de cada uma das dimensões no cálculo da unidade vencedora para cada nodo do mapa, um exemplo que apresentou bons resultados para a tarefa de agrupamento em subespaços. O modelo consegue realizar o agrupamento em subespaços aplicando pesos diferentes para cada dimensão de entrada.

Com isso, o modelo permite que o peso de algumas dimensões seja praticamente anulado, fazendo com que estas dimensões não influenciem no agrupamento estabelecido por um dado nodo, e esta é uma característica relevante para esta Dissertação porque com isso, podem ser colocadas diferentes relevâncias quando os nodos existentes crescem em dimensão (Capítulo 3), por exemplo. O ajuste destes pesos é realizado de forma adaptativa durante o treinamento. Apesar disso ainda se faz necessário um modelo que possa ajustar sua topologia ao longo do tempo para que possa ser possível a geração de nodos de qualquer tamanho, independentemente do tamanho inicial. Por isso, está apresentado com maiores detalhes o LARFDSSOM que é um modelo baseado em SOM e que também herda as características do DSSOM.

O *Local Adaptive Receptive Field Dimension Selective Self-organizing Map* (LARFDSSOM), proposto em (BASSANI; ARAUJO, 2015), também é um exemplo que apresentou bons resultados para a tarefa de agrupamento em subespaços, possuindo topologia variante, não sendo necessário especificar a quantidade de agrupamentos existentes. Isto permitem uma nova gama de aplicações relacionadas ao agrupamento além do agrupamento no subespaço, no qual a descoberta de *Motifs* está incluída.

O LARFDSSOM possui três fases: a fase de organização, a fase de convergência e a fase de agrupamento (BASSANI; ARAUJO, 2015). Na fase de organização, os nós são dinamicamente inseridos, removidos e ajustados no mapa para cobrir as regiões do espaço de entrada nas quais os padrões de entrada são encontrados da melhor maneira possível. Quando um padrão de entrada é apresentado para a rede, um nível de ativação é calculado para cada nodo do mapa, e o nodo com a ativação mais alta é considerado o vencedor da competição. Este nível de ativação é uma função inversa da distância entre o centro do nodo e o padrão de entrada. Um limite de ativação, a_t , é usado para determinar quando o nodo vencedor está próximo o suficiente para ser ajustado ou se é necessário inserir um novo nodo no mapa. Os nodos que não agrupam uma porcentagem significativa de padrões de entrada são periodicamente removidos, fazendo com que os nodos removidos

sejam desconectados da sua vizinhança.

Cada nodo j no LARFDSSOM representa um grupo associado com três vetores m -dimensionais, onde m é o número atual de dimensões de entrada, são eles: a primeira dimensão é o centro do vetor $\mathbf{c}_j = \{c_{ji}, i = 1, \dots, m\}$ que representa o grupo do protótipo j no espaço de entrada; o segundo é o vetor de relevâncias $\boldsymbol{\omega}_j = \{\omega_{ji}, i = 1, \dots, m\}$, que armazena os pesos (variando entre $[0, 1]$) que o nodo j aplica a i -ésima dimensão de entrada; e o terceiro é o vetor de distância $\boldsymbol{\delta}_j = \{\delta_{ji}, i = 1, \dots, m\}$, que armazena a média móvel das distâncias obtidas a partir dos padrões de entrada \mathbf{x} e do centro do vetor $|\mathbf{x} - \mathbf{c}_j(n)|$, que é apenas utilizado para calcular o vetor de relevâncias, como apresenta Bassani e Araujo (2015).

Uma função de base radial de uma distância ponderada (Equação 2.7) é usada para calcular a ativação do nodo no LARFDSSOM. Os campos receptivos dos nodos são ajustados em função da distância ponderada entre o protótipo, o padrão de entrada, e o somatório do vetor de relevâncias $\sum_{i=1}^m \omega_{ji}$. Distâncias pequenas e altas relevâncias resultam em uma ativação alta. A equação da ativação é

$$ac(D_\omega(\mathbf{x}, \mathbf{c}_j), \boldsymbol{\omega}_j) = \frac{\sum_{i=1}^m \omega_{ji}}{\sum_{i=1}^m \omega_{ji} + D_\omega(\mathbf{x}, \mathbf{c}_j) + \epsilon}, \quad (2.7)$$

onde ϵ é um valor muito pequeno para prevenir a divisão por zero e $D_\omega(x, c_j)$, é a distância ponderada, mostrada na Equação 2.8, como proposto em Bassani e Araújo (2012).

$$D_\omega(\mathbf{x}, \mathbf{c}_j) = \sqrt{\sum_{i=1}^m \omega_{ji} (x_i - c_{ji})^2} \quad (2.8)$$

Quando um nodo é criado, o seu centro c_j é inicializado na posição do último padrão de entrada apresentado ao mapa. O vetor de relevâncias $\boldsymbol{\omega}_j$ é inicializado como um vetor de uns e o vetor de distância é inicializado como um vetor de zeros. Esses vetores são atualizados nas etapas de adaptação e cooperação, apresentados na Seção 3.2.

No LARFDSSOM, o vencedor da competição $s(\mathbf{x})$, é o nodo que apresenta o maior valor de ativação em relação ao padrão de entrada, como é definido na Equação 2.9.

$$S(\mathbf{x}) = \operatorname{argmax}[ac(D_\omega(\mathbf{x}, \mathbf{c}_j), \boldsymbol{\omega}_j)], \forall j \text{ onde } j = 1..N \quad (2.9)$$

Na fase de convergência, os nodos também são atualizados e removidos quando necessário, como na fase de organização. No entanto, não há inserção de novos nodos. Essa fase termina quando o número de nodos no mapa para de diminuir.

Após a fase de convergência, as informações armazenadas no mapa podem ser usadas para o agrupamento de novos padrões de entrada (fase de agrupamento). Todos os nós com uma ativação igual ou maior que o limiar a_t para um padrão de entrada específico são considerados como agrupados.

Finalmente, assim como no SOM, a camada de entrada do LARFDSSOM tem um comprimento fixo, não sendo possível o aumento ou diminuição dos tamanhos das amostras durante o processo de auto-organização.

No entanto, na descoberta de *Motifs*, há uma demanda por métodos que podem trabalhar com tamanhos diferentes de amostras, como apresenta (NUNTHANID; NIENNAT-TRAKUL; RATANAMAHATANA, 2011), para permitir a descoberta de padrões de tamanhos distintos. O LARFDSSOM não foi desenvolvido com este objetivo e, de acordo com o conhecimento adquirido com os estudos, não existe um método baseado em SOM que seja capaz de agrupar dados de entrada de tamanhos diferentes.

Com os conceitos apresentados na fundamentação teórica, no próximo capítulo, o método proposto (VILMAP) será apresentado visando solucionar os problemas, como por exemplo a segmentação de palavras.

3 VARIABLE INPUT LENGTH MAP - VILMAP

O VILMAP¹, ou Mapa com Tamanho de Entrada Variável, é um modelo baseado no LARFDSSOM, proposto por Bassani e Araujo (2015), capaz de receber estímulos com quantidade de dimensões desconhecidas. O método proposto herda algumas características importantes do LARFDSSOM que são apresentadas a seguir:

- A primeira é a capacidade do modelo adaptar sua estrutura topológica a medida que novos padrões são apresentados ao longo do tempo;
- A capacidade que os nodos possuem de aplicar relevâncias distintas em cada dimensão de entrada;
- O campo receptivo dos nodos pode ser ajustado durante a fase de auto-organização do modelo.

Uma das principais características herdadas ao VILMAP é a capacidade de aplicar relevâncias distintas em cada atributo de cada nodo, pois, com isso, em um modelo onde as entradas não possuem tamanho fixo, as relevâncias ponderam a parte da entrada que é mais relevante para agrupar um determinado tipo de estímulo.

Estando o VILMAP apto a receber tamanhos de entrada desconhecidos, os dados de entrada podem aumentar ou diminuir de dimensão que ainda assim os nodos representativos serão capazes de se ajustar para agrupar padrões de entrada, sendo assim, um nodo pode retornar uma ativação alta estando alinhado com o padrão de entrada. Além disso, até mesmo partes dos nodos podem retornar um valor de ativação alto, por conta do tamanho dos padrões que podem ser maiores, menores ou iguais ao do nodo em questão.

O VILMAP é composto por duas fases: a fase de Auto-organização, mostrada no Algoritmo 1, e a fase de agrupamento, mostrada no Algoritmo 3. A fase de convergência existente no LARFDSSOM, não está presente no VILMAP já que um dos objetivos do modelo criado é de trabalhar de maneira online. Assim como acontece no SOM padrão, a fase de organização do VILMAP é composta por três etapas que são listadas a seguir:

1. Etapa de competição
2. Etapa de adaptação
3. Etapa de cooperação

De maneira similar ao LARFDSSOM, quando um novo padrão de entrada é apresentado, uma competição acontece entre os nodos, e o nodo vencedor é aquele que possui o

¹ Disponível em: <<https://github.com/RaphaBrito/VILMAP>>

maior nível de ativação de acordo com uma função de base radial (ver linha 5 do Algoritmo 1). Além disso, uma operação de deslocamento foi construída para comparar o sinal do padrão de entrada com a informação armazenada no protótipo, viabilizando o cálculo da distância entre o sinal do padrão de entrada e todas as possíveis posições de cada protótipo do mapa, deslocando a entrada de um em um por todas as posições do nodo (ver linha 6 do Algoritmo 1). Logo ao término deste procedimento, um nodo vencedor é obtido, e para ser o vencedor, ele precisa ter o maior valor de ativação em pelo menos um dos deslocamentos realizados.

Algoritmo 1 Fase de Auto-organização

- 1: Inicialização dos parâmetros a_t , e_b , e_n , β , ϵ_{ds} , n_{max} , d_{min} , d_{max} e $minwd$
 - 2: Inicializa o mapa: o nodo c_j é inicializado com o primeiro estímulo recebido na entrada, $\delta_j \leftarrow 0$ e $\omega_s \leftarrow 1$
 - 3: **enquanto** *existe_padrao_de_entrada*(\mathbf{x}) **faça**
 - 4: \mathbf{x} é apresentado ao mapa
 - 5: A ativação entre \mathbf{x} e todos os nodos é calculada de acordo com a Equação 2.7
 - 6: Encontra o nodo vencedor s que possui a maior ativação (a_s) conforme a Equação 2.9
 - 7: **se** *tamanhoDe*(s) < *tamanhoDe*(\mathbf{x}) **então** \triangleright “*tamanhoDe*” retorna a dimensão do vetor passado como argumento
 - 8: *atualizaDimensao*(s , \mathbf{x}) \triangleright descrita no Algoritmo 2
 - 9: **fim se**
 - 10: **se** $a_s < a_t$ **e** $n < n_{max}$ **então**
 - 11: Criação de um novo nodo j e defina: $c_j \leftarrow \mathbf{x}$ e $\delta_j \leftarrow 0$
 - 12: Conexão do novo nodo j aos outros nodos
 - 13: $n \leftarrow n + 1$
 - 14: **senão se** $a_s \geq a_t$ **então**
 - 15: Atualiza δ_s , ω_s e c_s do nodo vencedor e de seus respectivos vizinhos
 - 16: Atualiza ω_s do nodo vencedor e de seus respectivos vizinhos
 - 17: Atualiza c_s do nodo vencedor e de seus respectivos vizinhos
 - 18: **fim se**
 - 19: **fim enquanto**
-

Assim, é realizada a verificação a fim de saber se a ativação está acima ou abaixo de um limiar (parâmetro a_t). Quando a ativação está abaixo do limiar de ativação, um novo nodo é criado com o centro na mesma posição do padrão de entrada. Por outro lado, quando a ativação está acima do limiar (a_t), o nodo vencedor pode ser maior ou igual ao tamanho do padrão de entrada ou menor.

Se o nodo vencedor for maior ou igual ao estímulo, a etapa de adaptação e a etapa de cooperação são realizadas normalmente, caso contrário, se o nodo vencedor for menor do que o padrão de entrada, antes das etapas de adaptação e cooperação, é realizada a atualização da quantidade de dimensões do nodo que venceu (ver linha 8 do Algoritmo 1), para que ele possua a mesma quantidade de dimensões do padrão de entrada, como é

descrito no Algoritmo 2. Na atualização da dimensão do nodo alguns passos precisam ser feitos, aumentar alguns vetores (ver linhas 4, 6 e 8 do Algoritmo 2) e preencher as novas partes criadas (ver linhas 5, 7 e 9 do Algoritmo 2), após o processo de aumento.

Na fase de adaptação o nodo vencedor é atualizado tornando-o mais próximo do padrão de entrada e na fase de cooperação, todos os vizinhos do nodo são atualizados. Todo este processo está detalhado na Seção 3.2.

Algoritmo 2 Atualização da Dimensão do Nodo

- 1: **Entrada:** O nodo vencedor j e o estímulo atual \mathbf{x} .
 - 2: **Saída:** O nodo vencedor j com sua dimensão atualizada.
 - 3: **função** ATULIZADIMENSAO(\mathbf{s} , \mathbf{x})
 - 4: Aumenta δ_j até ele possuir a mesma dimensão do padrão de entrada \mathbf{x} ;
 - 5: A nova parte do vetor δ_j é preenchida com o valor 0.0;
 - 6: Aumenta ω_j até ele possuir a mesma dimensão do padrão de entrada \mathbf{x} ;
 - 7: A nova parte do vetor ω_j é preenchida com o valor 0.5; \triangleright um valor intermediário de relevância
 - 8: Aumenta \mathbf{c}_j até ele possuir a mesma dimensão do padrão de entrada \mathbf{x} ;
 - 9: Nas respectivas posições, a nova parte do vetor \mathbf{c}_j é preenchida com os mesmos valores do padrão de entrada \mathbf{x} ;
 - 10: **fim função**
-

Na Figura 8, pode ser observado um exemplo prático de como irá acontecer a atualização da dimensão do nodo. Utilizando-se da palavra *STAND*, na língua inglesa, partindo da suposição que o protótipo vencedor é o apresentado na Figura 8 e que suas relevâncias estão dispostas logo a baixo de cada dimensão, e na última parte da Figura 8, é apresentado o nodo vencedor com uma dimensão acrescida a ele. No decorrer deste capítulo será melhor explicado o processo de atualização de tamanho do nodo vencedor e no Capítulo 4, será apresentado um experimento aplicado na área de aquisição de linguagem natural utilizando a fonética.

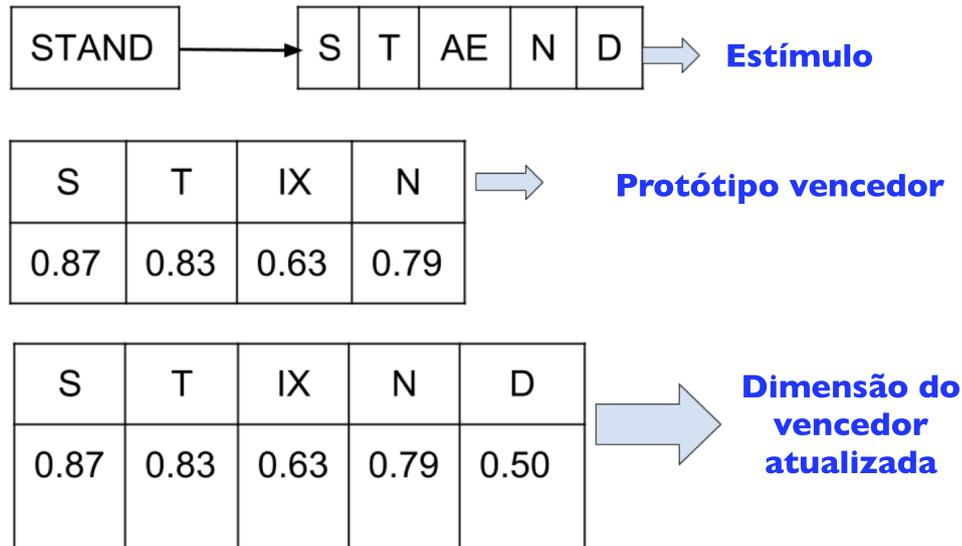


Figura 8 – Exemplo que ilustra a atualização da dimensão do nodo vencedor sendo atualizada. Estão ilustrados o vetor de protótipos e o vetor de relevâncias.

Após apenas uma execução da fase de Auto-organização, a fase de agrupamento já pode ser executada, já que em um modelo online, a fase de agrupamento pode ocorrer em paralelo com a fase de organização. No VILMAP, o processo de agrupamento associa o padrão de entrada a apenas um grupo, cujo é representado pelo nodo do mapa que retornou a maior ativação (ver linha 4 do Algoritmo 3). A fase de agrupamento será apresentada com mais detalhes na Seção 3.3.

3.1 COMPETIÇÃO E INSERÇÃO DOS NODOS

O VILMAP possui um limiar de ativação a_t . Se o nodo vencedor possui ativação abaixo do limiar a_t , um novo nodo é criado na mesma posição do padrão de entrada e o nodo vencedor não é modificado. Assim, o limiar de ativação a_t afeta diretamente na quantidade de nodos inseridos no mapa.

O VILMAP pode receber padrões de entrada de diferentes dimensões, para isso, é necessário desenvolver um novo procedimento para calcular a distância, originalmente apresentada na Equação 2.8, entre os padrões de entrada e os nodos no mapa, visto que não é trivial calcular distância entre vetores com tamanhos distintos, pois nesse caso, devem ser tratadas três situações distintas:

1. **Comparação regular:** se o tamanho do nodo em comprimento é igual ao tamanho do padrão de entrada, então o cálculo da distância é calculado diretamente como no LARFDSSOM (Equação 2.8), com o nodo e o padrão de entrada completamente alinhados.
2. **Comparação em janela deslizante:** se o nodo vencedor é maior em comprimento do que o padrão de entrada, a informação armazenada no nodo será comparada

com cada parte do padrão de entrada, como acontece em uma abordagem de janela deslizante, mudando uma posição de cada vez. Então a ativação é calculada em cada posição de acordo com a Equação 2.9 e o maior valor de ativação é retornado como resultado no final do processo. Isto é ilustrado na Figura 9 em que dois deslocamentos são possíveis onde o comprimento do padrão é igual a 4 e o comprimento do nodo é igual 5, portanto, duas ativações são calculadas. Neste exemplo, o nodo alcança a maior ativação no deslocamento B.

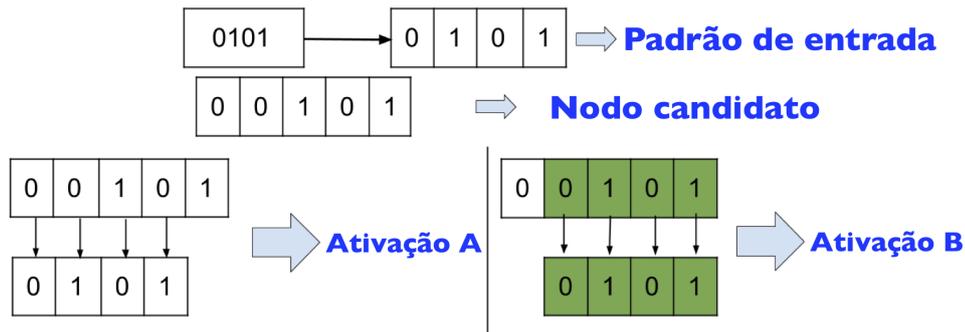


Figura 9 – Exemplo de como a ativação é calculada em um conjunto de dados binário para um padrão de entrada com um comprimento menor que o nodo.

3. **Comparação truncada:** Se o nodo for menor em comprimento que o padrão de entrada, a ativação é calculada somente na parte inicial do padrão de entrada e a parte final do estímulo é desconsiderada como ilustrado na Figura 10. Neste caso, se um nodo que passou por este procedimento se tornar o nodo vencedor e tiver uma ativação acima a_t , terá sua dimensão atualizada (Seção 3.2).

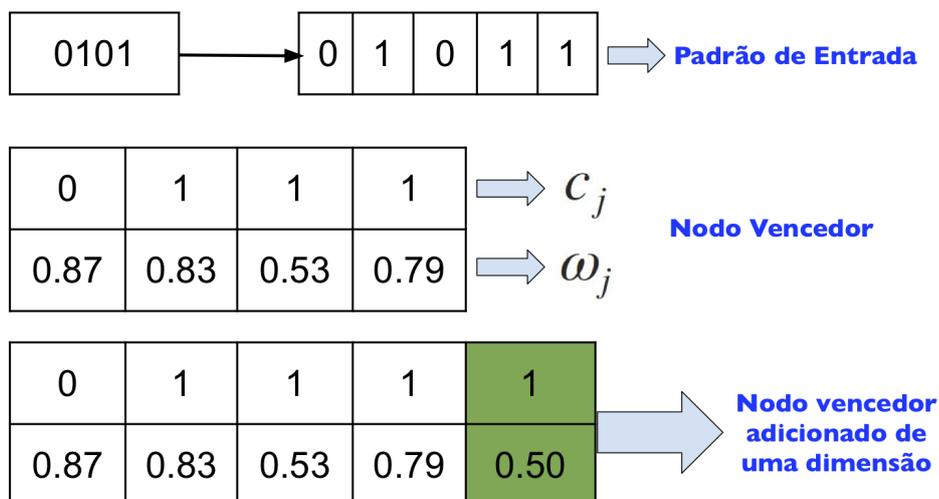


Figura 10 – Exemplo que ilustra a atualização da dimensão dos vetores do nodo.

Por outro lado, quando a ativação do nodo vencedor está acima do limiar a_t , o nodo vencedor e os seus vizinhos são atualizados, como descrito na próxima seção e mostrado no Algoritmo 1.

3.2 ATUALIZAÇÃO DO NODO VENCEDOR E DA SUA VIZINHANÇA

No VILMAP, estímulos de diferentes tamanhos podem ser recebidos a qualquer momento. Consequentemente, o vetor m -dimensional do nodo vencedor pode ter um tamanho diferente do padrão de entrada. Aqui novamente, três situações são possíveis:

1. **Atualização regular:** se o tamanho do nodo em comprimento é igual ao tamanho do padrão de entrada, o nodo é atualizado diretamente como é feito no LARFDSSOM, com o nodo e o padrão de entrada completamente alinhados.
2. **Atualização deslocada:** se o nodo é maior em comprimento do que o padrão de entrada, a atualização é feita primeiramente alinhando o padrão com o nodo vencedor precisamente na posição que produziu a ativação mais alta no nodo vencedor. Então, finalmente os vetores são atualizados como no LARFDSSOM, desconsiderando as partes que estão fora do alinhamento com o nodo vencedor.
3. **Atualização truncada:** se o nodo é menor em comprimento do que o padrão de entrada, então, primeiramente, as dimensões dos vetores associadas com o nodo vencedor são atualizadas como mostrado a seguir: todos os vetores associados ao nodo vencedor aumentam até que tenham as mesmas dimensões que o padrão de entrada. Então, os novos elementos do vetor c_j são inicializados com os mesmos valores do padrão de entrada; as novas posições do vetor δ_j são inicializados com 0,0; e os novos elementos do vetor ω_j são inicializados com um valor intermediário (0,5 assumindo dados normalizados no intervalo de 0-1). A Figura 10 também apresenta tal procedimento de atualização de dimensão. Finalmente, os vetores alinhados são atualizados como no LARFDSSOM.

3.3 AGRUPAMENTO COM O VILMAP

Como o VILMAP opera com fluxo contínuo de dados, após uma certa quantidade de padrões apresentados ao mapa, a informação armazenada em cada nodo do VILMAP pode ser utilizada para agrupar um padrão de entrada teste. O Algoritmo 3 apresenta a fase de agrupamento. Cada nodo no mapa define um grupo e todos os padrões de teste possuem um nodo vencedor, onde o padrão de teste é agrupado. Na fase de agrupamento, a ativação dos nodos é calculada da mesma forma que é descrita na Seção 3.2.

Algoritmo 3 Fase de Agrupamento

- 1: **enquanto** *existe_padrao_de_entrada* **faça**
 - 2: o padrão de entrada \mathbf{x} é apresentado ao mapa
 - 3: A ativação dos nodos é calculada de acordo com a Equação 2.7
 - 4: Encontra o nodo vencedor \mathbf{s} com a maior ativação (a_s) (Equação 2.9)
 - 5: **se** $a_s \geq a_t$ **então**
 - 6: Atribui \mathbf{x} ao cluster com o índice do nodo vencedor \mathbf{s}
 - 7: **fim se**
 - 8: **fim enquanto**
-

Após a apresentação do VILMAP, no presente capítulo, é preciso realizar a validação do modelo, que será mostrada no chaptername 4, onde serão apresentados experimentos que envolvem tanto dados reais, bem como dados simulados.

4 VALIDAÇÃO DO MODELO

Nesta seção será apresentada a validação do VILMAP, cujos experimentos serão introduzidos e descritos para viabilizar a reprodução de todos os resultados obtidos nesta pesquisa. Os experimentos realizados foram divididos em duas partes:

1. Experimentos que usam bases de dados simulados, mostrados na (Seção 4.3);
2. Experimentos que usam bases de dados reais, descritos na (Seção 4.4).

Os experimentos de dados simulados (Seção 4.3) foram realizados com o principal objetivo de avaliar as características do VILMAP. Já os experimentos realizados com as bases de dados reais (Seção 4.4) possuem um foco maior em apresentar aplicações realizadas utilizando o VILMAP, ou seja, verificar se o modelo é capaz de reconhecer dados vistos em treinamento (com tamanhos de entrada variável) sem que os dados que não foram vistos os confundam. Um roteiro com os objetivos de cada experimento realizado está apresentado a seguir:

- **Reconhecimento de Padrões de Tamanhos Diferentes** - Os bebês precisam reconhecer palavras de tamanhos diferentes;
- **Reconhecimento de Padrões com Tamanhos Crescentes sem esquecer os Padrões Menores Previamente Aprendidos** - Os bebês precisam reconhecer palavras cada vez maiores sem esquecer as palavras aprendidas anteriormente;
- **Descoberta de Padrões Previamente Desconhecidos em Séries temporais** - Este é um problema tipicamente avaliado na área de reconhecimento de padrões em séries temporais;
- **Aprendizagem de Sequência de Fonemas de tamanhos variáveis** - O modelo precisa reconhecer assim como os bebês reconhecem fonemas;
- **Reconhecimento de Palavras** - Depois do reconhecimento de fonemas, o VILMAP precisa reconhecer palavras, que é o nosso quinto experimento;
- **Coerência Topológica do Mapa** - Por fim, foi verificado se vizinhos mais próximos do nodo são mais parecidos com ele.

4.1 AJUSTE PARAMÉTRICO DO VILMAP

Para alcançar bons resultados, nos experimentos realizados nesta Dissertação, foram amostrados 100 conjuntos de parâmetros usando os intervalos descritos na Tabela 2, de acordo com a técnica *Latin Hypercube Sampling* (LHS) (HELTON; DAVIS; JOHNSON, 2005). O LHS

é empregado para garantir cobertura completa do intervalo de cada parâmetro. O intervalo de cada parâmetro é dividido em 100 intervalos de igual probabilidade e um único valor é selecionado aleatoriamente de cada intervalo, como pode ser visto na Figura 11. Assim, a junção de cada valor gerado de cada parâmetro, forma o conjunto de parâmetros a ser utilizado para teste.

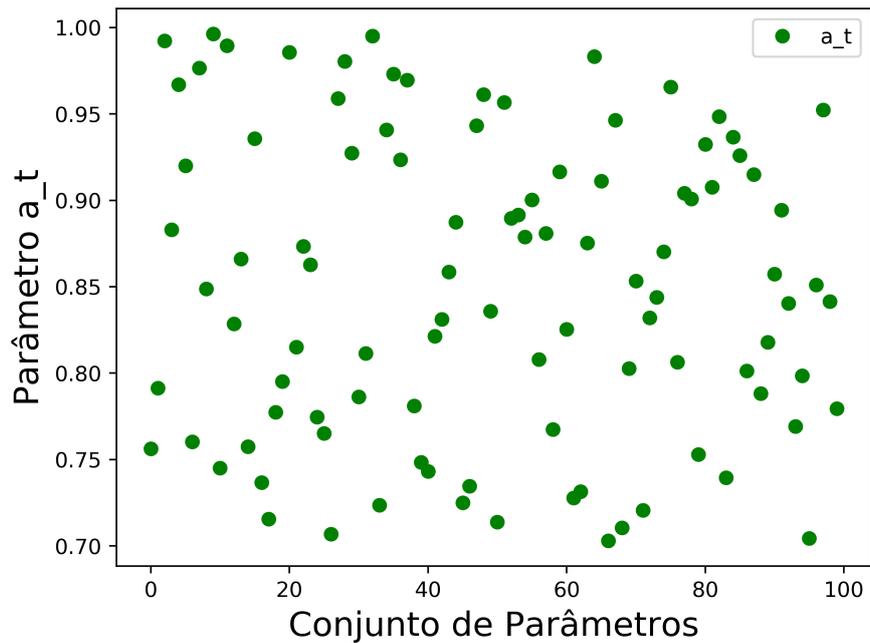


Figura 11 – Todos os valores gerados pelo LHS para os 100 conjuntos de parâmetros. Com isso, pode-se observar que toda a região do intervalo definido foi ocupada pelos pontos.

O ajuste de parâmetros é feito seguindo como base a metodologia estabelecida em (BASSANI, 2014). A Tabela 2 mostra os intervalos utilizados para cada parâmetro. O parâmetro mais importante para o VILMAP é o a_t porque este parâmetro influencia diretamente no número de nodos que o mapa criará, assim como acontece no LARFDSSOM. Com um valor baixo de a_t , menos nodos serão inseridos no mapa, por outro lado, com um limiar mais alto, os nodos reconhecerão menos padrões de entrada, acarretando em mais nodos criados no mapa.

Tabela 2 – Intervalo de parâmetros VILMAP

Parâmetros	min	max
Limiar de ativação (a_t)	0.70	0.999
Taxa de relevância (β)	0.001	0.5
Taxa de aprendizado do vencedor (e_b)	0.0001	0.01
Taxa de aprendizado dos vizinhos (e_n)	0.002	$1.0 \times e_b$
Limiar de conexão ($minwd$)	0.001	0.5
Suavização da curva de relevância (ϵ_{ds})	0.01	0.1

4.2 MÉTRICAS

As métricas utilizadas nesta Dissertação para avaliar o modelo proposto são: Precisão, Revocação e a Medida-F. Na Figura 12, pode-se visualizar, de maneira intuitiva, como são calculadas cada uma das três métricas. Em linhas gerais, a precisão (Equação 4.1) é o cálculo dos verdadeiros positivos (vp) divididos pela soma dos verdadeiros positivos (vp) com os falsos positivos (fp); já a revocação (Equação 4.2) é calculada pela divisão entre os verdadeiros positivos (vp) e a soma dos verdadeiros positivos (vp) com os falsos negativos (fn). Assim, a precisão consegue responder a seguinte pergunta: Quantos, dos elementos selecionados, são relevantes?; e com o cálculo da revocação conseguimos responder a seguinte pergunta: Quantos elementos relevantes foram selecionados?

$$\text{Precisão} = \frac{vp}{vp + fp}, \quad (4.1)$$

$$\text{Revocação} = \frac{vp}{vp + fn}. \quad (4.2)$$

Com as medidas de precisão e revocação calculadas, podemos então calcular a Medida-F (Equação 4.3). A medida-F é a medida que combina precisão e revocação, sendo a média harmônica entre elas. Em geral, a Medida-F é calculada utilizando-se do quadrado da média geométrica dividido pela média aritmética, neste caso ela é chamada de F1.

$$\text{Medida - F} = 2 * \frac{\text{Precisão} * \text{Revocação}}{\text{Precisão} + \text{Revocação}} \quad (4.3)$$

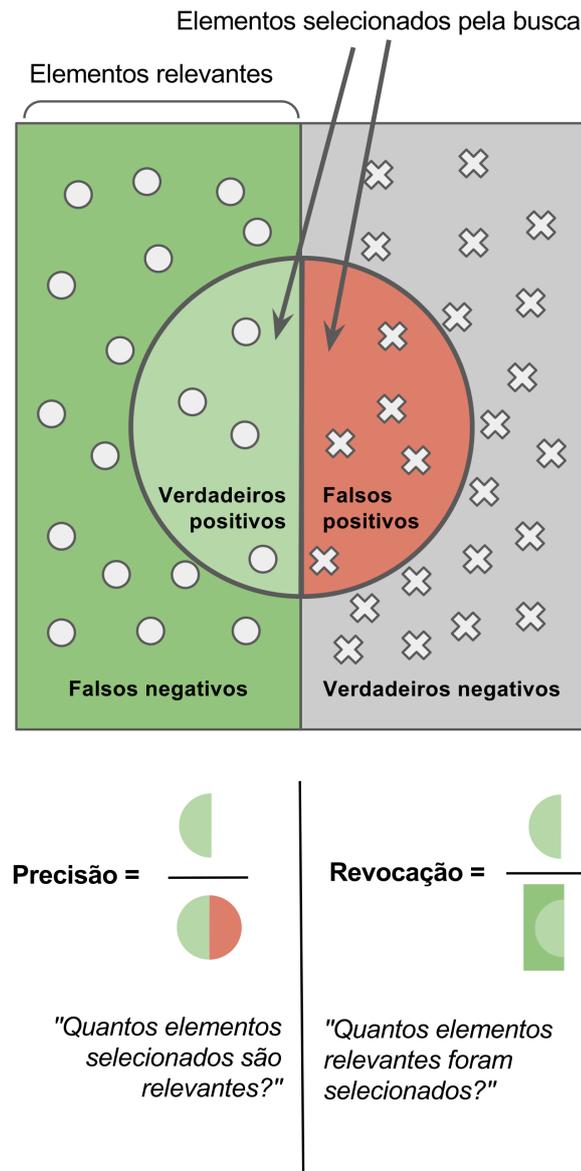


Figura 12 – Ilustra o cálculo, de maneira intuitiva, da medida de precisão e da medida de revocação (Fonte: Wikipédia, acessado em Junho/2018).

4.3 EXPERIMENTAÇÃO COM DADOS SIMULADOS

Experimentos usando dados simulados foram realizados para que as características do modelo fossem visualizadas com mais facilidade, em relação aos dados reais.

4.3.1 Bases de Dados Simulados

Os Dados simulados foram gerados utilizando-se do pacote Python, chamado Numpy, disponibilizado por (OLIPHANT, 2006). Primeiramente, o vetor de médias com dimensão D foi criado e preenchido com um valor aleatório com distribuição uniforme no intervalo $[0 \dots 1]$. Após o vetor de médias, foi gerada a matriz de covariância, de dimensão $D \times D$, sendo completamente preenchida com zeros possuindo apenas a sua diagonal principal

preenchida com um valor amostrado uniformemente dentro do intervalo $[0 \dots 0, 25]$. Com o vetor de médias e a matriz de covariância criados, foi gerada uma normal multivariada de dimensão $D = 100$ e quantidade de padrões $p = 30$.

A cada repetição do processo explicado no parágrafo anterior, pode-se gerar classes de dados para formar uma base de dados com dados simulados, onde cada classe criada possuirá trinta padrões.

Na Figura 13 pode ser observadas as duas primeiras dimensões de uma base de dados criada contendo quatro classes que mais a diante serão utilizadas como os dados verdadeiros positivos, ou seja, serão apresentadas a rede para que o modelo possa aprendê-las. Note o alto nível de sobreposição entre as classes, que é típico quando observadas em apenas duas dimensões.

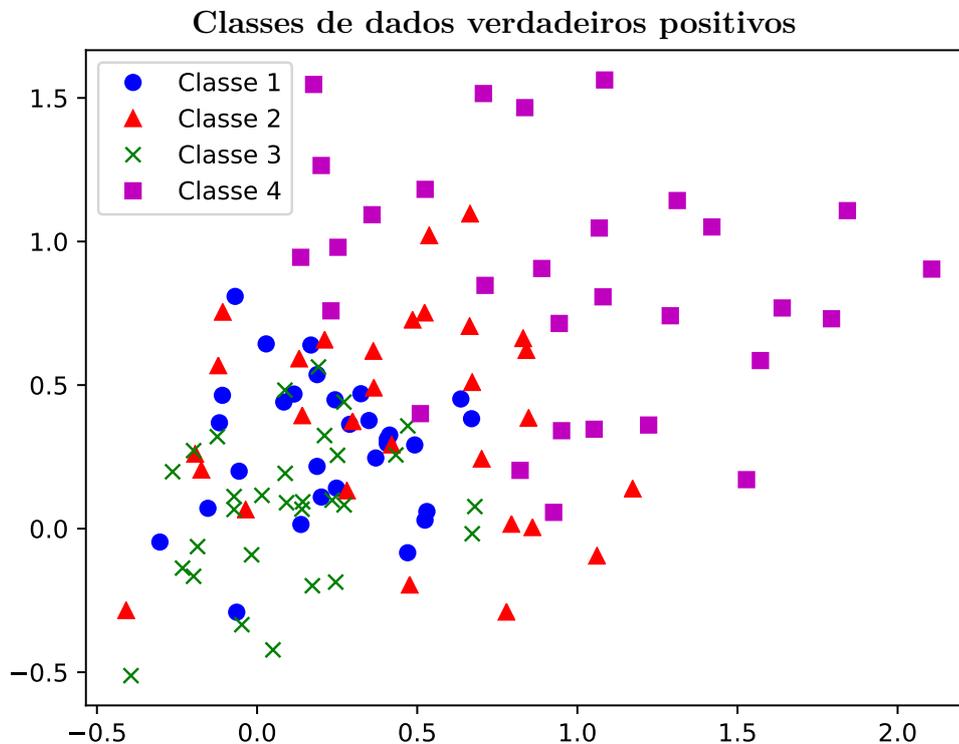


Figura 13 – Primeira e segunda dimensões das quatro classes geradas para serem verdadeiros positivos.

Já na Figura 14, podem ser vistas as duas primeiras dimensões das quatro classes, também geradas aleatoriamente, neste caso com o objetivo de serem os falsos positivos, ou seja, que serão apresentadas ao modelo apenas para testá-lo.

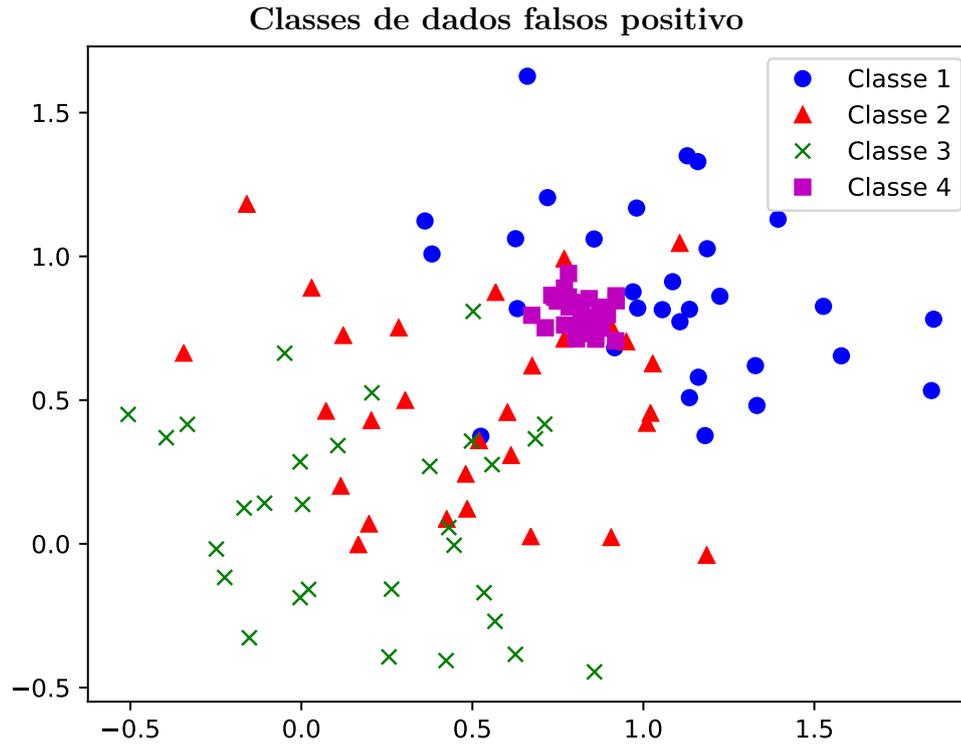


Figura 14 – Primeira e segunda dimensões das quatro classes geradas para serem falsos positivos.

Na Figura 15, pode ser observado a alta sobreposição de dados entre verdadeiros positivos e falsos negativos. No momento em que os falsos positivos forem apresentados ao modelo, com o intuito de testá-lo, pode ser que alguns pontos estejam suficientemente próximos de dados verdadeiros positivos, a ponto de serem confundidos. Assim, podendo acarretar em categorização como verdadeiro positivo, quando o modelo deveria apontar falso positivo.

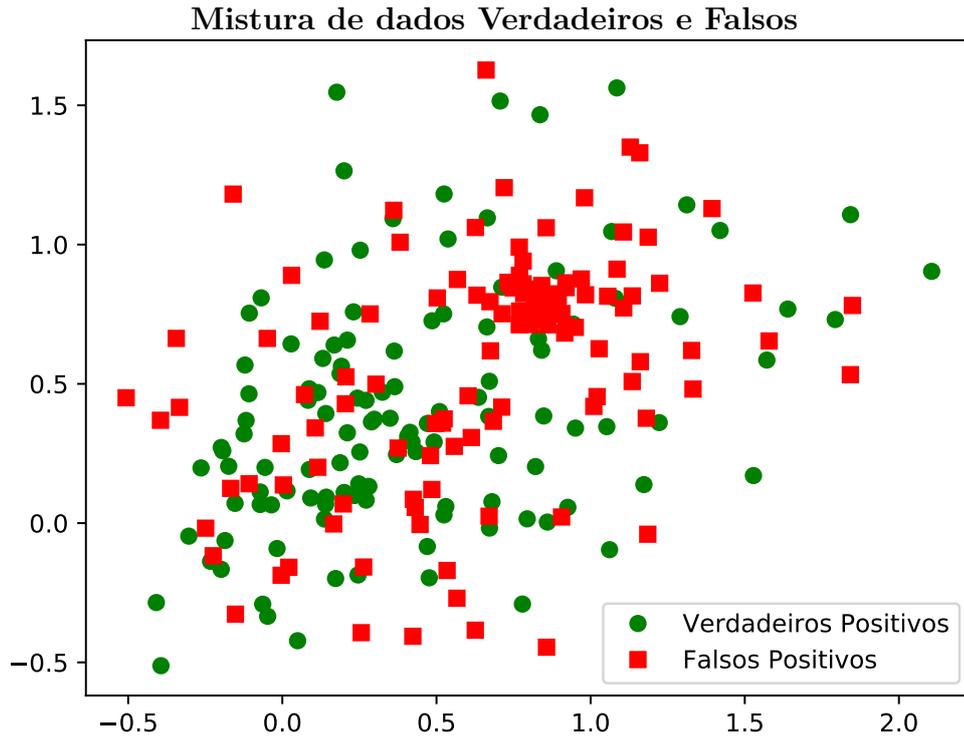


Figura 15 – Visualização da primeira e segunda dimensões da mistura de dados gerados para treinamento e teste do modelo.

Nas Seções 4.3.2 e 4.3.3 serão apresentados dois experimentos realizados utilizando-se da base de dados simulados apresentada na presente seção.

4.3.2 Reconhecimento de Padrões de Tamanhos Diferentes - Experimento 1

A Figura 16 apresenta os passos executados de maneira sequencial com o objetivo de verificar se o VILMAP consegue lidar com padrões de tamanhos variáveis, reconhecendo-os.

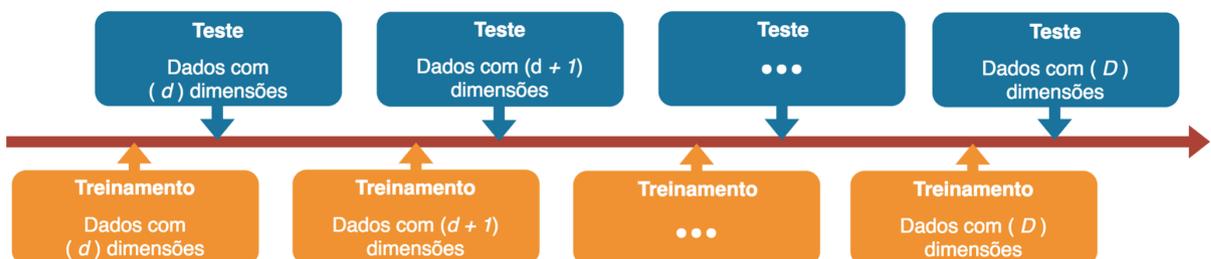


Figura 16 – Primeira sequência de treinamentos e testes. Sequência executada com o objetivo de verificar a capacidade do modelo em aprender padrões de tamanhos variáveis.

O experimento ilustrado na Figura 16 inicia-se com o treinamento de padrões de entrada que possuem d dimensões, seguido pelo teste que usa os mesmos padrões de treinamento (para verificar se o modelo é capaz de reconhecer os verdadeiros positivos)

e outros padrões que não foram apresentados anteriormente (para que o modelo possa verificar a existência de verdadeiros negativos). Assim, os treinamentos e testes vão sendo executados de maneira alternada com incrementos de uma em uma dimensão até a maior dimensão $D = 100$.

Após a realização do experimento, explicado com o auxílio da Figura 16, pode-se visualizar o resultado na Figura 17 expressado utilizando a Precisão, a Revocação e a Medida-F.

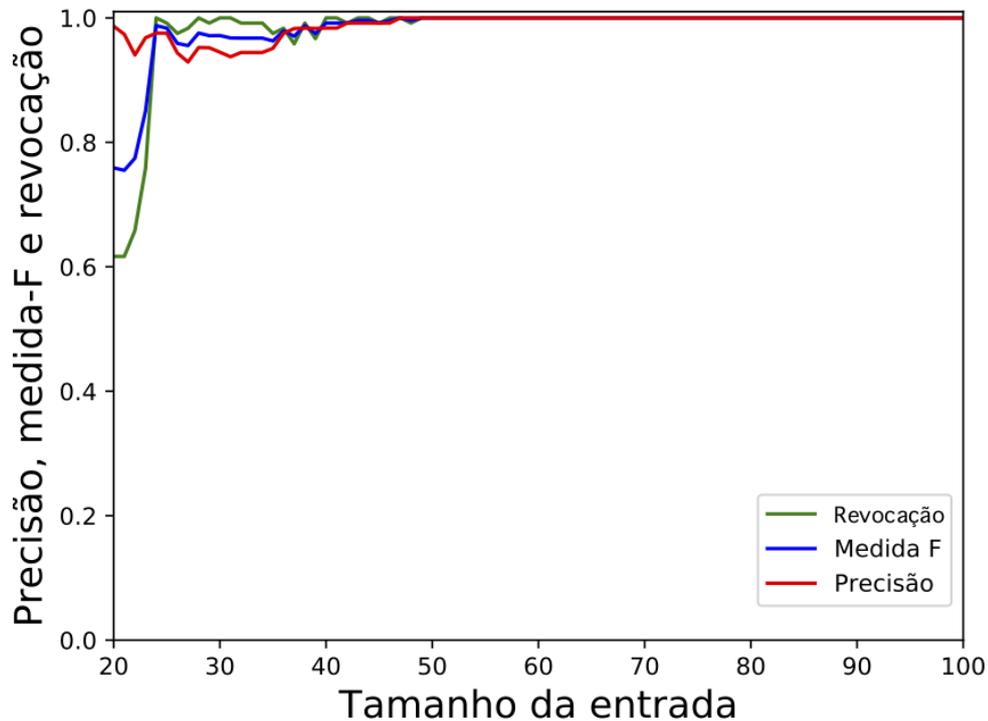


Figura 17 – Precisão, Medida-f e Revocação resultantes do experimento com o melhor conjunto de parâmetros encontrado.

Na Seção 4.3.3, será apresentada uma continuação do atual experimento onde será possível verificar como o modelo lida com o esquecimento catastrófico.

4.3.3 Reconhecimento de Padrões com Tamanhos Crescentes sem Esquecer os Padrões Menores Previamente Aprendidos - Experimento 2

Na Figura 18 podem ser vistos os passos executados sequencialmente com o objetivo de testar a capacidade do modelo aprender padrões de tamanhos crescentes sem esquecer os padrões treinados no início do treinamento, evitando assim o esquecimento catastrófico.

O experimento descrito na Figura 18 inicia-se com o treinamento de padrões de entrada que possuem d dimensões. Logo em seguida, a dimensão d é incrementada em 1 e o treinamento continua até que o tamanho da entrada seja igual a D . Terminado o treinamento, inicia-se o teste que usa os mesmos padrões de treinamento (para verificar

os verdadeiros positivos) e outros padrões que não foram apresentados anteriormente (que o modelo deve rejeitar).

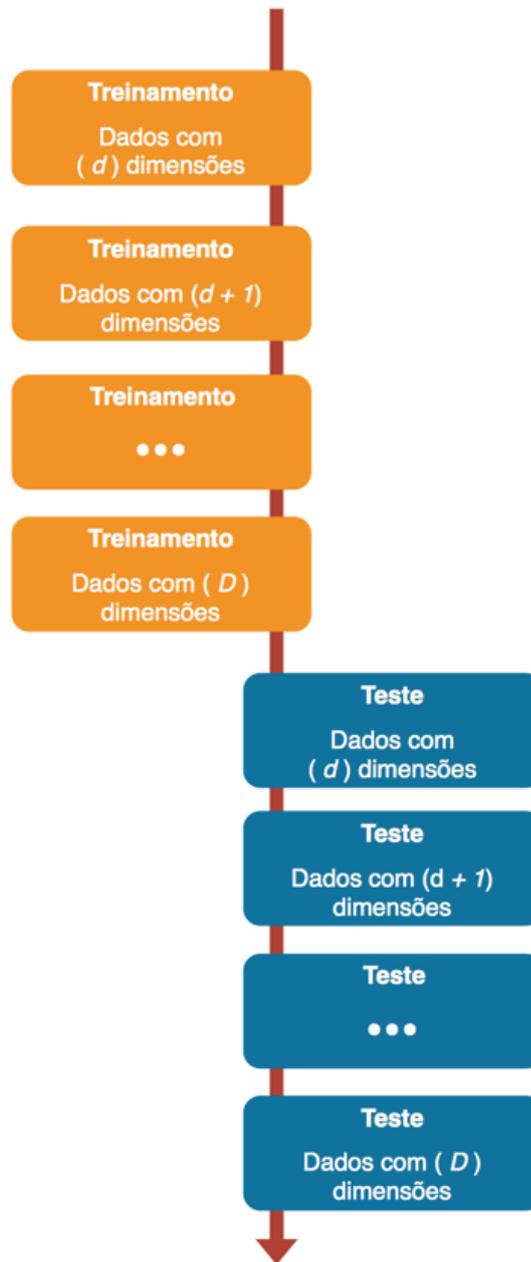
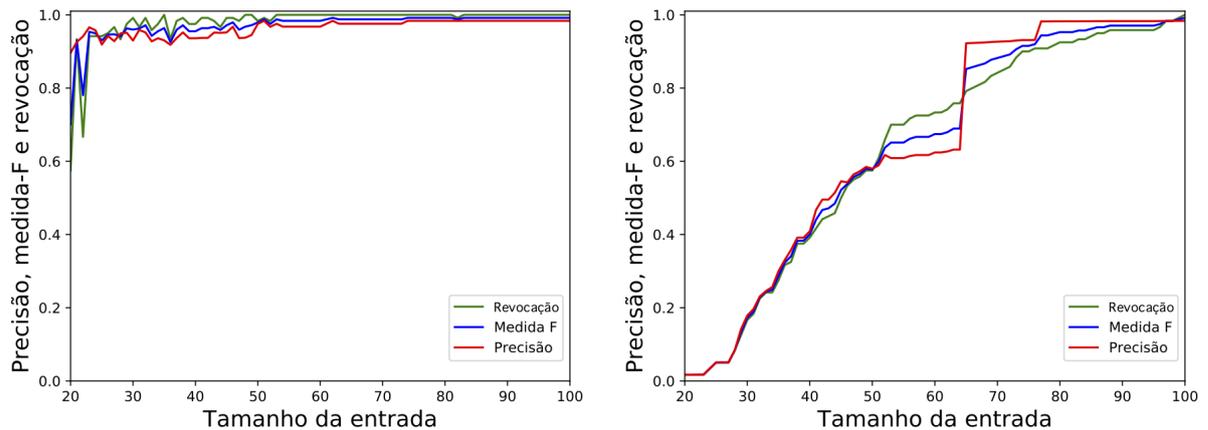


Figura 18 – Segunda sequência de treinamentos e testes. Sequência executada com o objetivo de verificar a capacidade do modelo em aprender padrões de tamanhos crescentes sem esquecer os padrões de tamanhos menores.



(a) Experimento executado utilizando a primeira sequência de treinamento e teste. (b) Experimento executado utilizando a segunda sequência de treinamento e teste.

Figura 19 – Resultados do experimento com o melhor conjunto de parâmetros encontrado usando as duas formas de treinamento e teste (a) e (b). O objetivo, neste caso, foi verificar o esquecimento catastrófico gerado pelo modelo.

Na Figura 19, pode-se analisar o resultado do modelo após a execução do experimento descrito nesta seção. Na Figura 19a, é apresentado o resultado do experimento executado utilizando o primeiro método de treinamento e teste criado (Figura 16) e na Figura 19b, é mostrado o resultado da execução do experimento utilizando o segundo método de treinamento e teste criado (Figura 18).

Por fim, pode-se verificar na Figura 20, a curva de recordação do modelo gerada relacionando os dois resultados apresentados na Figura 19. Quatro pontos ao longo da Figura 20 foram destacados. O primeiro é o ponto que apresenta entradas de tamanho 20 quando o modelo estava treinado com entradas de tamanho de 20 até 100 com saltos de um em um, e o valor é de 2,4% de recordação. Já quando o modelo foi testado com padrões de tamanho 70, a taxa de recordação subiu para 89,3%.

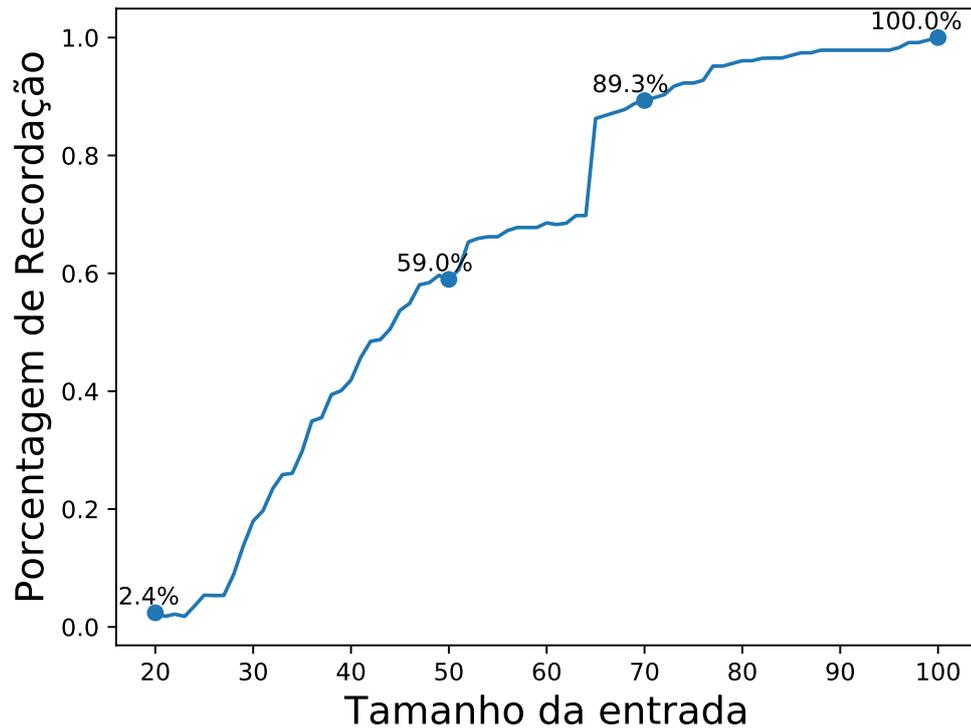


Figura 20 – Curva da porcentagem de recordação do modelo. Por exemplo, depois que a rede foi treinada até o tamanho 100, se for apresentada ao VILMAP padrões de tamanho 70 é possível que 89,3% desses padrões de tamanho 70 ainda sejam reconhecidos/recordados.

Com isso, este resultado foi considerado um resultado satisfatório já que o modelo não esquece totalmente os dados aprendidos anteriormente, e além disso, ainda é necessário levar em consideração, que todas as entradas são apresentadas apenas uma vez ao mapa, ou seja, se as entradas menores fossem apresentadas mais vezes, após o mapa já ter sido treinado com o tamanho de entrada indo de 20 até 100, o modelo poderia lembrar de mais padrões menores.

Assim, após os experimentos executados utilizando a base de dados simulados, apresentada na Seção 4.3.1, serem avaliados, na Seção 4.4 são explicados os exemplos executados utilizando bases de dados reais.

4.4 EXPERIMENTAÇÃO COM DADOS REAIS

Três experimentos serão apresentados utilizando bases de dados reais. Apesar de o modelo funcionar com dados de tamanhos variáveis, o primeiro experimento utiliza uma base de dados de tamanho fixo com o objetivo maior de mostrar que mesmo com as alterações feitas ao modelo base, o LARFDSSOM, o VILMAP continua resolvendo problemas da literatura de tamanho fixo. O segundo experimento utiliza sequências de fonemas de tamanhos variáveis para que possa ser testada a aprendizagem destas sequências ao longo do tempo. E por fim, o terceiro experimento, tem um foco na segmentação de palavras de tamanhos variáveis.

4.4.1 Descoberta de Padrões Previamente Desconhecidos em Séries Temporais - Experimento 3

O objetivo deste experimento é mostrar que mesmo com todas as alterações feitas ao LARFDSSOM, o modelo proposto nesta pesquisa, VILMAP, ainda consegue resolver problemas padrão de tamanho fixo da literatura. Este experimento consiste em receber como entrada os padrões da base GunPoint (CHEN et al., 2015) e fazer o treinamento. Ao término do treinamento, os padrões gerados pelo modelo são comparados com as a média e desvio das duas classes da base de dados GunPoint.

4.4.1.1 Bases de Dados

A base de séries temporais utilizada para este experimento foi a GunPoint (CHEN et al., 2015) que é um conjunto de dados que envolve uma atriz e um ator fazendo um movimento com a mão, como mostra a Figura 21. Esta base é dividida em duas classes: *Gun-Draw* e *Point*. Na classe *Gun-Draw*, os atores começam com as mãos ao lado do corpo, puxam uma réplica de uma arma pendurada em um coldre no quadril, apontam para um alvo por aproximadamente um segundo e devolvem a arma para o coldre. Já na segunda classe, *Point*, os atores deixam suas armas ao lado deles e fazem um movimento apontando os seus dedos indicadores para um alvo por aproximadamente um segundo e por fim retornam suas mãos para a lateral do corpo. Para ambas as classes, são rastreados os centróides das mãos direitas dos atores nos eixos X e Y. Os pontos no arquivo da base de dados são apenas do rastreamento do eixo X.



Figura 21 – Exemplo da base de dados GunPoint (RATANAMAHATANA; KEOGH, 2005).

4.4.1.2 Análise dos resultados

O campo receptivo dos nodos gerados no VILMAP foi ajustado de maneira tal que com apenas dois nodos fosse possível reconhecer as classes existentes na base de dados GunPoint. Assim, após a fase de treinamento do VILMAP, foram gerados dois nodos.

A Figura 22 apresenta o resultado para a primeira classe da base de dados GunPoint em relação ao conteúdo armazenado no centróide do nodo vencedor para os padrões desta classe gerado na pelo modelo. Em azul, pode ser observada a média e o desvio padrão da primeira classe, e em laranja, é mostrado graficamente conteúdo do nodo gerado para representar a primeira classe.

Já na Figura 23, pode ser observado o resultado para a segunda classe da base de dados GunPoint em relação ao nodo gerado pelo modelo, representada da mesma forma.

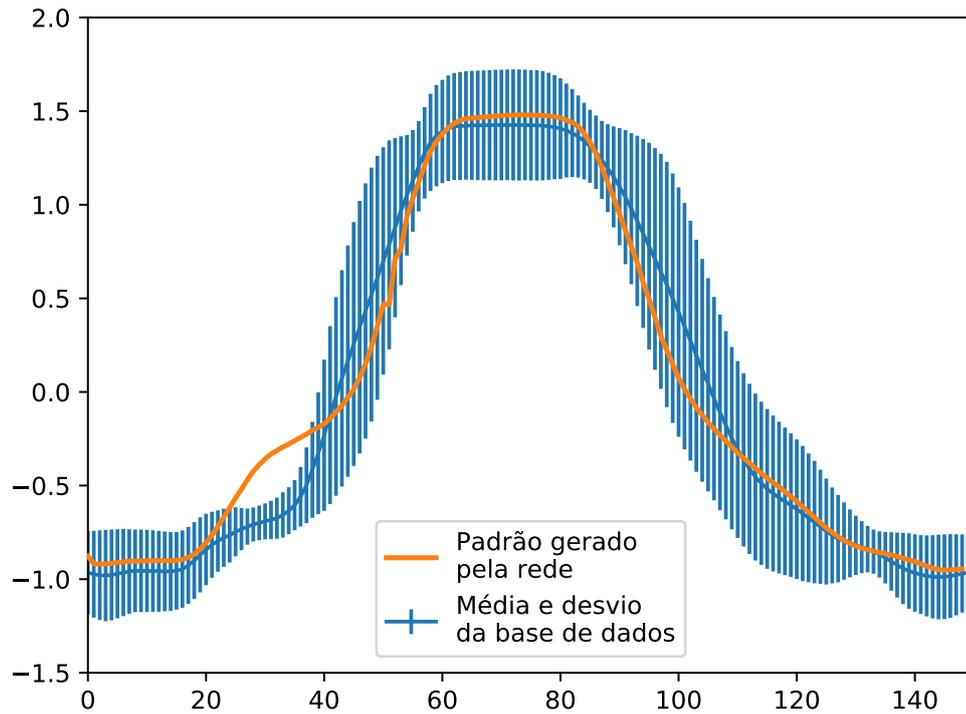


Figura 22 – Resultado para a primeira classe da base de dados GunPoint, executados com o modelo VILMAP.

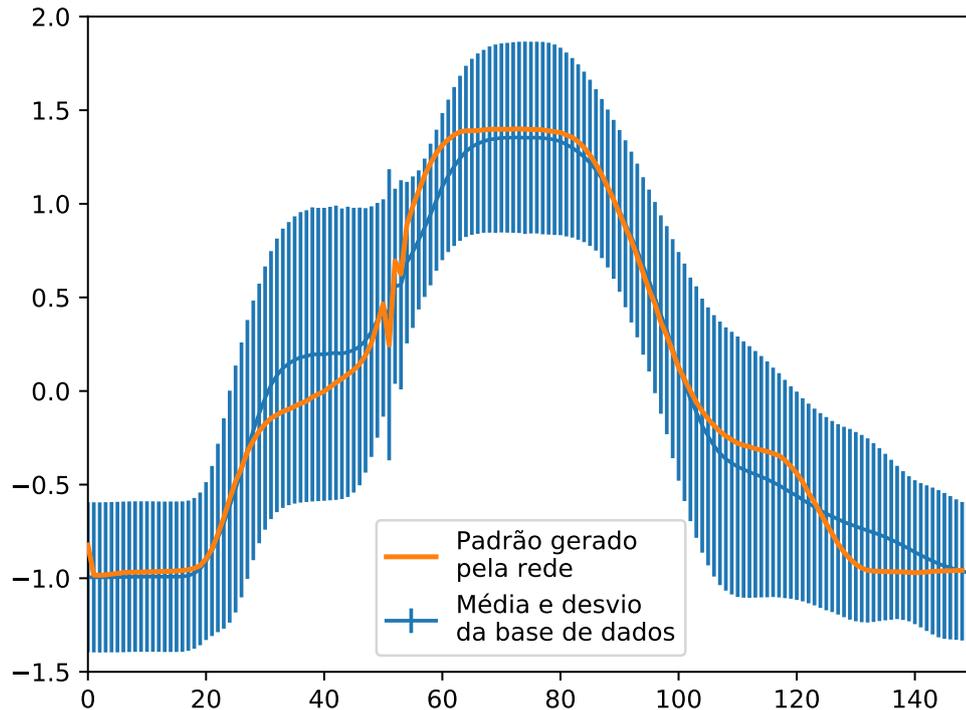


Figura 23 – Resultado para a segunda classe da base de dados GunPoint, executados com o modelo VILMAP.

Nas Figuras 22 e 23, que apresentam os resultados para o experimento atual, é perceptível a influência das duas classes existentes nos dois nodos gerados para representá-las. Um exemplo pode ser visto entre os pontos 20 e 40 da Figura 22, onde a curva gerada pelo nodo sai um pouco do referencial da classe em questão, que é causado pelo fato que o nodo gerado para representar a primeira classe da base de dados GunPoint aceitou alguns padrões da segunda classe, como mostram a média e o desvio padrão da segunda classe na Figura 23.

O experimento realizado na atual seção, sobre descoberta de padrões desconhecidos em séries temporais, também foi executado usando o LARFDSSOM com o objetivo de mostrar que mesmo após o VILMAP receber todas as modificações para estar apto a trabalhar com problemas com tamanho de entrada variáveis, ainda assim o modelo continua funcionando bem para problemas que possuem a dimensão de entrada fixa. Nas Figuras 24 e 25, são apresentadas as duas classes geradas após o treinamento usando o LARFDSSOM. Pode-se observar que, comparando as Figuras 22 com a 24 e 23 com a 25, entre si, os resultados são semelhantes. Com isso, considera-se que, o VILMAP, funciona bem para problemas com entrada fixa, assim como o LARFDSSOM que é um método especializado em problemas de dimensão fixa.

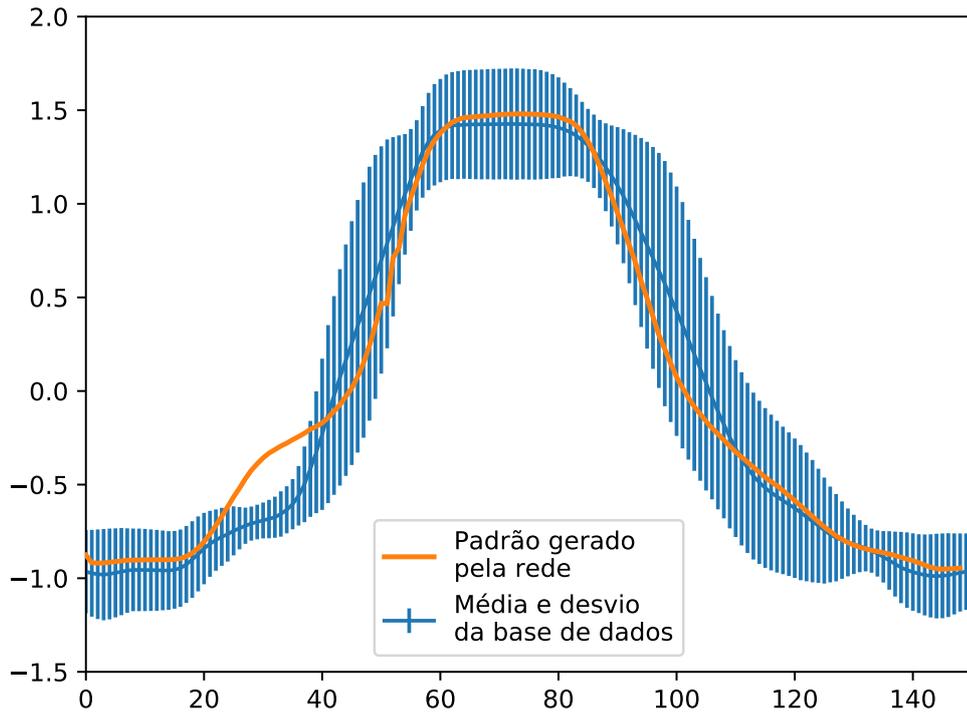


Figura 24 – Resultado para a segunda classe da base de dados GunPoint, executados com o modelo LARFDSSOM.

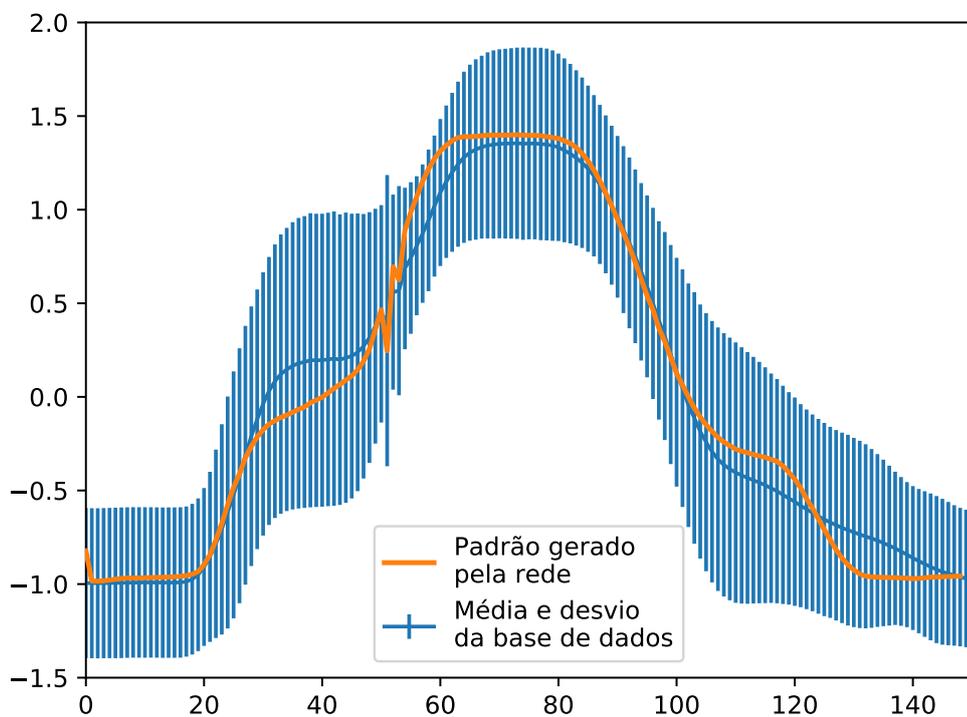


Figura 25 – Resultado para a segunda classe da base de dados GunPoint, executados com o modelo LARFDSSOM.

Contudo, o experimento apresentado na seção atual conseguiu atingir as expectativas de verificar se o VILMAP ainda consegue resolver problemas de tamanho fixo presentes na literatura.

4.4.2 Aprendizagem de Sequência de Fonemas de Tamanhos Variáveis - Experimento 4

Na presente seção, o VILMAP será aplicado ao aprendizado não-supervisionado de palavras, utilizando características extraídas de fonemas como entrada da rede. Espera-se que o modelo se adapte de maneira tal, conseguindo reconhecer palavras vistas e rejeitar palavras não vistas.

4.4.2.1 Bases de Dados

Os dados utilizados para o treinamento foram o texto transcrito da fala com 130 sentenças da base de dados *timit* extraída do NLTK (*Natural Language Toolkit*), disponibilizado por (BIRD; LOPER, 2004).

A conversão de palavras para fonemas é feita utilizando a ferramenta *LOGIOS Lexicon Tool* (CMU, 2011) que gera um dicionário de palavras versus fonemas a partir de um arquivo de texto, escrito em inglês. Em seguida, é preciso fazer a conversão de fonemas em características. A extração de características a partir dos fonemas será feita assim como sugerido por (BASSANI, 2014), que levou em consideração as propriedades distintivas de fonemas dependendo da articulação dos fonemas quando são pronunciados.

Como é apresentado na Figura 26, as palavras da série temporal de entrada são traduzidas para fonemas e cada fonema possui 12 características, onde cada característica é uma entrada para a rede neural. No exemplo apresentado ainda na Figura 26, o tamanho da janela de tempo utilizado é igual a 4 fonemas, logo, o tamanho mínimo da entrada da rede neural será igual a 48, pois as características de cada fonema são concatenadas em um vetor único que serve como entrada para a rede.

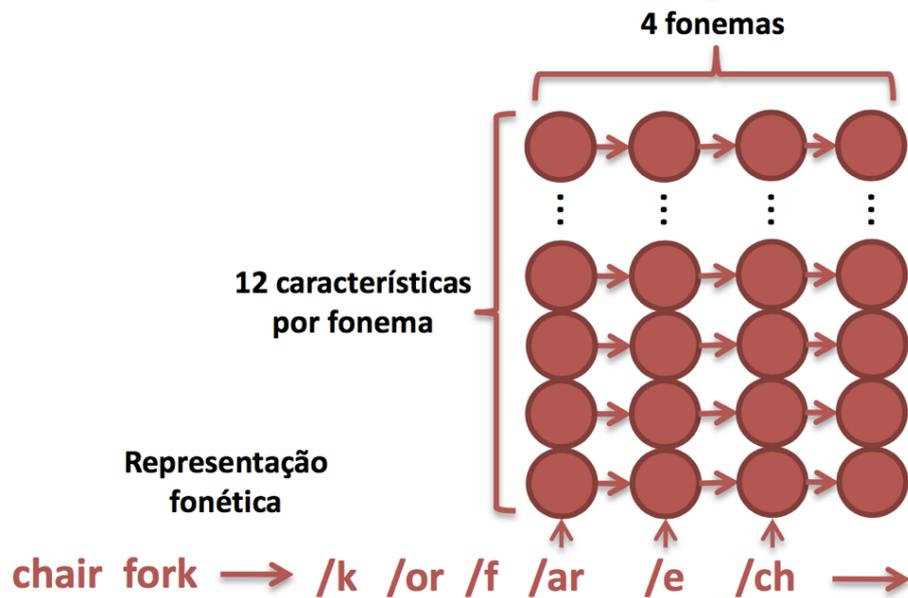


Figura 26 – Tradução das palavras em características. Exemplo que mostra o tamanho da janela de tempo igual a 4 fonemas, logo, o tamanho mínimo da entrada do modelo será de 48, já que as características de cada fonema são concatenadas em um vetor único que serve como entrada para a rede (BASSANI, 2014).

Na Tabela 3, pode ser observado um exemplo da representação fonética gerada de acordo com o dicionário fonético da Carnegie Mellon, através da ferramenta *LOGIOS Lexicon Tool* (CMU, 2011).

Palavras	Representação fonética
COMPUTER	K AH M P Y UW T ER
CENTER	S EH N T ER
FEDERAL	F EH D ER AH L
UNIVERSITY	Y UW N AH V ER S AH T IY

Tabela 3 – Exemplos de palavras da língua inglesa traduzidas para fonemas utilizando o dicionário fonético disponibilizado pela Carnegie Mellon (CMU, 2011).

Na Figura 27, pode-se observar um exemplo de tradução de uma palavra, neste caso *CENTER*, para as suas respectivas características, usando o dicionário definido em (BASSANI, 2014).

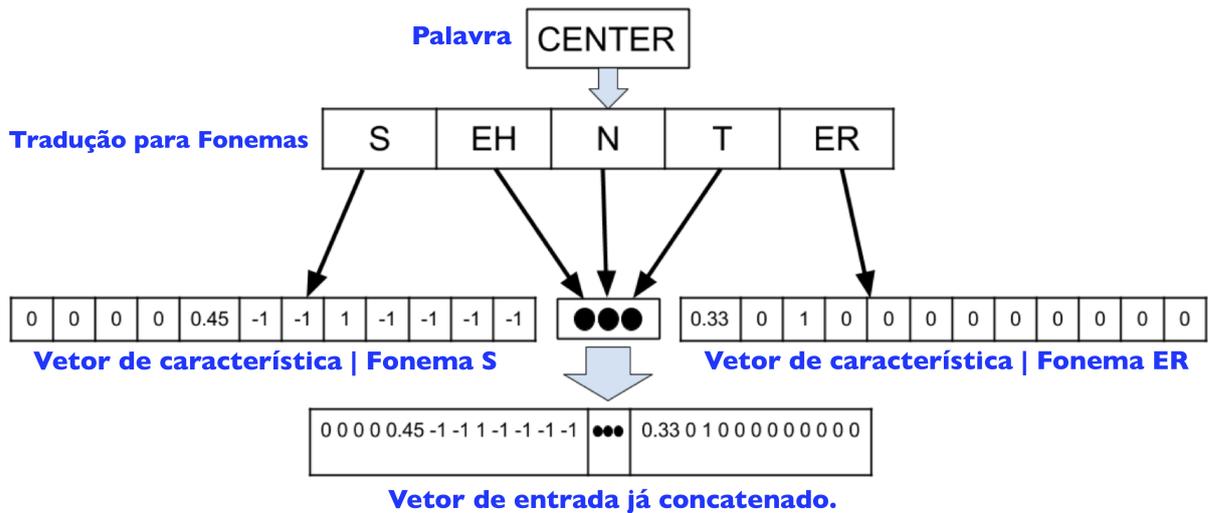


Figura 27 – Exemplo de tradução de uma palavra até o seu respectivo vetor de características, que neste caso, possui tamanho 60.

Para o experimento atual, o modelo foi treinado de duas maneiras diferentes para que a comparação visual pudesse ser exibida com clareza. Assim, foi escolhida uma pequena parte do resultado para ser apresentada a diferença da rede treinada com o deslocamento e sem o deslocamento no momento do treinamento e do teste, explicados no Capítulo 3.

4.4.2.2 Análise dos resultados

Um visualizador utilizando cores foi criado para deixar mais clara a interpretação do resultado. As cores que aparecem nos padrões de entrada e no cálculo da ativação, nas Figuras 28 e 29, seguem a ideia de mapa de calor, onde, o padrão de entrada é colorido da seguinte maneira: quanto mais relevante a dimensão, mais próxima da cor vermelha ela será; e quanto menos relevante for a dimensão, mais próxima do azul ela estará. A mesma ideia de cores serve para o valor da ativação, logo, quanto maior a ativação da entrada em relação ao protótipo, mais vermelho será o valor da ativação; e quanto mais baixa for a ativação, mais próxima do azul será o valor do cálculo da ativação.

Na Figura 28 apresenta-se a visualização dos padrões de entrada, do número identificador do nodo, valor da ativação, e protótipos da rede, respectivamente. Os padrões de entrada estão coloridos dependendo da relevância que cada dimensão tem para que este fosse agrupado ao nodo. Nos dois, os padrões foram agrupados em nodos maiores que ele. Na primeira linha, é visto que a entrada e o protótipo estão iguais e com relevâncias altas, ou seja, existe um nodo que representa muito bem esse padrão de entrada. Já na segunda linha, o protótipo está bastante diferente e com relevância baixa, porque, neste caso, não foi utilizado o deslocamento na fase de agrupamento. O segundo padrão de entrada se encaixaria perfeitamente no nodo de número 4391, pois o padrão de entrada é exatamente o protótipo da primeira linha com o deslocamento de uma coluna para frente.

Padrão de Entrada			Protótipo vencedor e suas relevâncias					
IY S ER V AY	4391	1.00	1.00	1.00	1.00	1.00	1.00	1.00
S ER V AY V	3919	0.99	0.42	0.64	0.63	0.57	0.53	0.64
			JH	OW	Z	OY	L	OY

ID do nodo
Ativação resultante

Figura 28 – Agrupamento feito sem o deslocamento do VILMAP. Na primeira coluna, estão os estímulos, traduzidos para fonemas, recebidos pelo mapa; na segunda coluna, estão os códigos identificadores do nodo para o mapa; na terceira coluna, está o resultado do cálculo da ativação entre o estímulo e o nodo (na mesma linha); e na colunas seguintes, estão: acima as relevâncias e abaixo o protótipo traduzido para fonemas.

Já na Figura 29, pode se observar que com os mesmos dados submetidos, agora utilizando o deslocamento do VILMAP (explicado no Capítulo 3), o resultado é totalmente diferente e agora o padrão de entrada pode ser reconhecido pelo nodo de número 4391, pois a parte final deste nodo responde com relevância alta ao padrão de entrada, que agora tem o comportamento esperado.

Padrão de Entrada			Protótipo vencedor e suas relevâncias					
IY S ER V AY	4391	1.00	1.00	1.00	1.00	1.00	1.00	1.00
S ER V AY V	4391	1.00	1.00	1.00	1.00	1.00	1.00	1.00
			IY	S	ER	V	AY	V
			IY	S	ER	V	AY	V

ID do nodo
Ativação resultante

Figura 29 – Agrupamento feito com o deslocamento do VILMAP. Neste caso, está em negrito, no protótipo, em qual parte do nodo a entrada foi reconhecida.

Com isso, é perceptível que, com o deslocamento criado para o agrupamento no VILMAP, o padrão de entrada consegue se encaixar não apenas no início de um nodo candidato a vencedor, mas sim, ao trecho do nodo que mais se ativar ao padrão de entrada.

4.4.3 Reconhecimento de Palavras - Experimento 5

O último experimento a ser apresentado nesta dissertação está focado no Reconhecimento de Palavras. Uma comparação será realizada com os experimentos executados por (LARSSEN; CRISTIA; DUPOUX, 2017). Neste experimento, o VILMAP é aplicado ao papel de

reconhecer palavras na transcrição de fala fluente. Considerando que o modelo proposto pode não ser capaz de identificar precisamente a separação entre as palavras, mas é capaz de reconhecer as palavras quando são apresentadas como entradas, para possibilitar a comparação do método proposto com os métodos apresentados em (LARSEN; CRISTIA; DUPOUX, 2017), é preciso reconhecer as palavras rejeitando as pseudo-palavras.

As pseudo-palavras são obtidas através dos deslocamentos realizados nos dados. Este procedimento faz com que o conjunto de pseudo-palavras geradas tenha uma distribuição de fonemas bastante semelhante à distribuição de fonemas das palavras reais. Todos os deslocamentos possíveis são colocados na rede como estímulos de entrada, para que o VILMAP identifique os padrões que aparecem com mais frequência e que podem ser de tamanhos diferentes, ou seja, que são de fato palavras. Para testar o mapa gerado pelo VILMAP, foram apresentadas à rede as palavras originais da base de dados para contar os verdadeiros positivos e os falsos negativos; e foi apresentado à rede o conjunto de pseudo-palavras para conseguirmos obter os falsos positivos e os verdadeiros negativos. Finalmente, como nos primeiros experimentos apresentados nesta dissertação, pode-se calcular a precisão, a revocação e a medida-F.

4.4.3.1 Bases de Dados

O conjunto de dados utilizado por (LARSEN; CRISTIA; DUPOUX, 2017) foi o mesmo utilizado nesta dissertação, de forma a permitir uma comparação justa, ou seja, a base de dados Brent-Siskind (BRENT; SISKIND, 2001). Esta base, é a maior do repositório CHIELDS (MACWHINNEY, 2000) e contém a transcrição ortográfica de mais de 100 horas de gravação de 16 mães de língua inglesa interagindo com crianças entre 9 e 15 meses de idade, no momento da gravação.

4.4.3.2 Análise dos resultados

Os resultados obtidos com o método proposto em comparação com os resultados apresentados em (LARSEN; CRISTIA; DUPOUX, 2017) são mostrados em Tabela 4. A partir dessa tabela, podemos ver que o VILMAP obteve uma medida F relativamente boa, perdendo apenas para o algoritmo AGu; uma precisão muito boa, superando todos os outros algoritmos; e, finalmente, uma revocação intermediária. Considera-se como um resultado promissor, pois o AGu modela um aluno com memória infinita e um processo em lote, enquanto nosso método representa um aluno on-line que passa pelos dados apenas uma vez.

Algoritmo	Medida-F	Precisão	Revocação
VILMAP	0,750	0,856	0,667
PUDDLE	0,706	0,682	0,733
DiBS	0,236	0,234	0,240
AG _u	0,782	0,787	0,777
TP _s	0,468	0,432	0,512

Tabela 4 – Resultados de Medida-F, precisão e revocação para o experimento de segmentação de palavras.

4.4.4 Coerência Topológica do Mapa - Nodos Vizinhos são Semelhantes - Experimento 6

O sexto experimento realizado foi focado em apresentar graficamente uma parte da vizinhança de um nodo selecionado aleatoriamente após a rede ter recebido como entrada a seguinte frase: *two hand first like new*, traduzida apresentar os fonemas obtidos após a tradução utilizando, assim como nos experimentos 4 e 5, o dicionário da Carnegie Mellon (CMU, 2011). Na Figura 30, são mostrados alguns vizinhos do nodo, selecionados aleatoriamente, que se encontra no centro da figura.

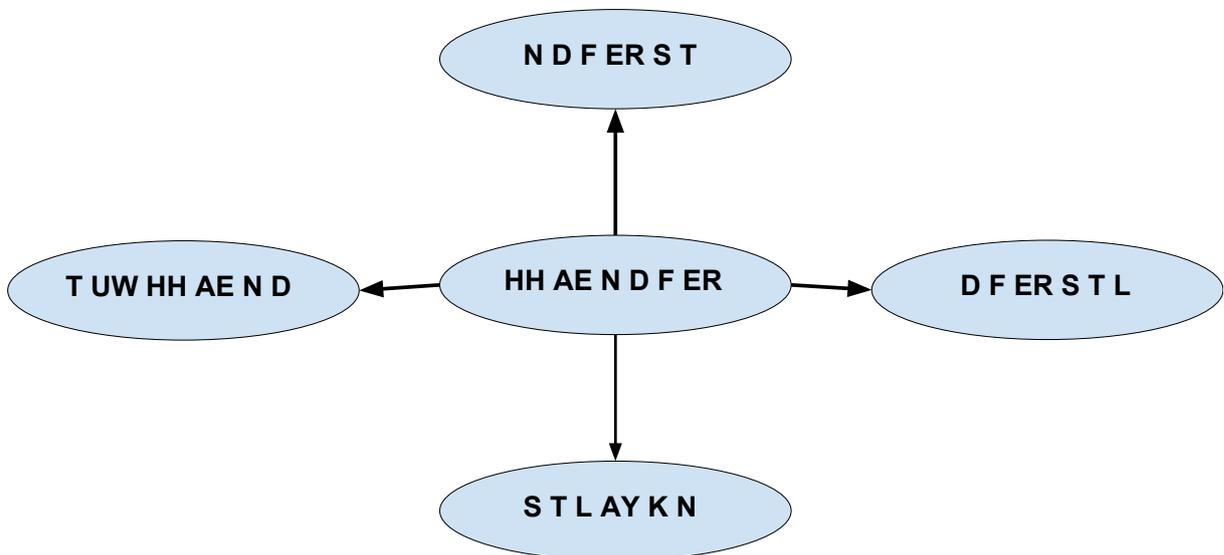


Figura 30 – Vizinhança do nodo localizado no centro da figura.

Pode-se perceber, utilizando a Figura 30, que a topologia do VILMAP mantém a característica que determina que os nodos e suas vizinhanças possuem uma semelhança. A diferença da topologia gerada pelo VILMAP, em relação ao SOM, é que no modelo proposto a vizinhança pode ser parecida em relação a qualquer parte do nodo, por conta do

deslocamento implementado no mapa proposto. Além disso, nodos de tamanhos diferentes podem ser vizinhos. Desta forma, conseguimos apresentar um mapa onde nodos estão conectados com outros nodos que são semelhantes a ele, não apenas na parte inicial, mas sim em qualquer das partes do nodo.

Os três vizinhos do nodo central que se encontram mais acima, possuem uma similaridade visualmente perceptível com o nodo central. Já o nodo que está abaixo do nodo central não apresenta uma similaridade tão perceptível, em relação ao nodo central, como os outros nodos vizinhos possuem. Todavia, se os detalhes da Figura 30 forem analisados, pode-se perceber que o nodo abaixo do nodo central possui uma visível semelhança com a vizinhança do nodo central. Logo, percebe-se que existe uma continuidade nos dados da vizinhança, onde, cada nodo vai aprendendo partes das palavras colocadas como entrada.

Após apresentar todos os experimentos neste capítulo, estão apresentadas as nossas considerações finais no próximo capítulo.

5 CONSIDERAÇÕES FINAIS

Neste capítulo será apresentada uma análise da modelagem proposta, as principais contribuições para a ciência, o trabalho publicado, as limitações do modelo serão expostas e por fim, serão mostrados trabalhos futuros e possíveis aplicações.

5.1 ANÁLISE DA MODELAGEM PROPOSTA

O modelo proposto (VILMAP), explicado no Capítulo 3 e validado no Capítulo 4, possui algumas características interessantes em relação ao SOM. Dos objetivos apresentados no Capítulo 1, quase todos foram alcançados.

O primeiro deles foi que o mapa deveria ser capaz de ajustar a relevância de cada uma das dimensões para permitir que as amostras de entrada sejam variáveis em relação ao número de dimensões de entrada, este objetivo foi alcançado.

O segundo objetivo foi o de reconhecer padrões com tamanhos crescentes sem esquecer os padrões menores previamente aprendidos, nesta situação o mapa teve um bom resultado, como pode ser observado no Experimento representado pela Figura 18. Analisando a Figura 20, foram destacados quatro pontos ao longo da curva de recordação. Um dos pontos é o que apresenta entradas de tamanho 50 quando o modelo estava treinado com entradas de tamanho de 20 até 100 com saltos de um em um, e o valor é de 59,3% de recordação. Com isso, pode-se concluir que o método consegue lembrar bem de parte do que foi apresentando com dimensões menores. Além disso, deve-se levar em consideração que os padrões pequenos são apresentados apenas no início, e depois não foram apresentados em nenhum momento. Existe a possibilidade de melhorar o esquecimento se os padrões menores fossem apresentados mais vezes ao mapa.

O terceiro e último objetivo elencado no Capítulo 1 foi o de realizar um agrupamento topológico, de forma que padrões semelhantes sejam agrupados em nodos próximos no mapa. Pode ser verificado através do Experimento apresentado na Seção 4.4.4.

Outro ponto importante a ser discutido é o fato que o VILMAP não é capaz de identificar precisamente o limite entre as palavras, mas mesmo assim, consegue reconhecer as palavras sem segmentá-las. Assim também acontece com as crianças que erram na segmentação mas ainda assim conseguem reconhecer algumas palavras.

No decorrer da pesquisa várias dificuldades foram encontradas, como por exemplo, o fato que o modelo não consegue segmentar palavras. Com estudos e com a melhoria de conjuntos paramétricos, é possível perceber que o modelo consegue reconhecer as palavras mesmo sem ser capaz de segmentá-las, ou seja, mesmo sem ser capaz de encontrar os limites iniciais e finais das palavras.

5.2 CONTRIBUIÇÕES PARA A CIÊNCIA

A principal contribuição que esta dissertação possui para a ciência é a modelagem de um Mapa Auto-organizável que é capaz de receber estímulos de tamanhos diferentes e desconhecidos. Além disso, a colocação do problema de reconhecimento de palavras como um problema de reconhecimento de *Motifs* em séries temporais também é uma contribuição importante deste trabalho.

5.2.1 Trabalhos Publicados

Durante a execução do Mestrado, foi submetido o artigo referente à modelagem do VILMAP. O artigo foi aceito para o *International Joint Conference on Neural Networks (IJCNN 2018)*.

BRITO, Raphael C.; BASSANI, Hansenclever F. Self-Organizing Maps with Variable Input Length for Motif Discovery and Word Segmentation. In: **2018 International Joint Conference on Neural Networks (IJCNN)**. IEEE, 2018.

5.3 LIMITAÇÕES DO MODELO

Nesta seção estão listadas algumas limitações do modelo proposto:

- O modelo não é capaz de reconhecer totalmente os padrões com tamanhos crescentes sem esquecer parte dos padrões menores previamente aprendidos;
- O modelo não é capaz de identificar precisamente o limite entre as palavras;
- O modelo não é capaz de reconhecer como iguais padrões que se apresentem ao mapa em escala diferente, como por exemplo: áudio de fala lenta e áudio de fala rápida.

5.4 TRABALHOS FUTUROS E APLICAÇÕES

Os trabalhos futuros serão focados, inicialmente, na criação de um modelo, inspirado no VILMAP, pois este viabiliza a construção de mapas recorrentes com topologia variante para aprendizagem de conceitos hierárquicos, para que o modelo seja capaz de aprender padrões cada vez maiores sem esquecer os padrões aprendidos anteriormente, corrigindo assim uma das principais limitações do modelo proposto nesta Dissertação.

Além das aplicações apresentadas neste trabalho, o VILMAP, pode ser aplicado para a resolução de vários problemas, como por exemplo: agrupamento de características extraídas de imagens, de texto (a partir de fonemas ou sílabas), também pode ser aplicado a uma variedade de problemas envolvendo processamento sensorial, incluindo o visual e o auditivo.

REFERÊNCIAS

- ARAÚJO, A. R.; COSTA, D. C. Local adaptive receptive field self-organizing map for image color segmentation. *Image and Vision Computing*, Elsevier, v. 27, n. 9, p. 1229–1239, 2009.
- BASSANI, H. F. Modelos neurais modulares para aquisição de linguagem natural. Universidade Federal de Pernambuco, 2014.
- BASSANI, H. F.; ARAÚJO, A. F. Dimension selective self-organizing maps for clustering high dimensional data. In: IEEE. *Neural Networks (IJCNN), The 2012 International Joint Conference on*. [S.l.], 2012. p. 1–8.
- BASSANI, H. F.; ARAUJO, A. F. R. Dimension Selective Self-Organizing Maps With Time-Varying Structure for Subspace and Projected Clustering. *IEEE Transactions on Neural Networks and Learning Systems*, v. 26, n. 3, p. 458–471, mar 2015. ISSN 2162-237X. Disponível em: <<http://ieeexplore.ieee.org/document/6803941/>>.
- BIRD, S.; LOPER, E. Nltk: the natural language toolkit. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*. 2004. p. 31. Disponível em: <<http://www.nltk.org>>.
- BRENT, M. R.; SISKIND, J. M. The role of exposure to isolated words in early vocabulary development. In: *Cognition*. [s.n.], 2001. v. 81, p. 31–44. Disponível em: <<https://childes.talkbank.org/access/Eng-NA/Brent.html>>.
- BROCKWELL, P. J.; DAVIS, R. A. *Time series: theory and methods*. [S.l.]: Springer Science & Business Media, 2013.
- CHEN, Y.; KEOGH, E.; HU, B.; BEGUM, N.; BAGNALL, A.; MUEEN, A.; BATISTA, G. The UCR Time Series Classification Archive. July 2015.
- CMU. The carnegie mellon university pronouncing dictionary - a machine-readable pronunciation dictionary for north american english. on-line. 2011.
- CORREA, J.; DOCKRELL, J. E. Unconventional word segmentation in brazilian children's early text production. *Reading and Writing*, v. 20, n. 8, p. 815–831, Nov 2007. ISSN 1573-0905.
- CORRÊA, L. M. S. O que, afinal, a criança adquire ao adquirir uma língua? *Letras de Hoje*, v. 42, n. 1, 2007.
- DALAND, R.; PIERREHUMBERT, J. B. Learning diphone-based segmentation. *Cognitive science*, Wiley Online Library, v. 35, n. 1, p. 119–155, 2011.
- HELTON, J. C.; DAVIS, F.; JOHNSON, J. D. A comparison of uncertainty and sensitivity analysis results obtained with random and latin hypercube sampling. *Reliability Engineering & System Safety*, Elsevier, v. 89, n. 3, p. 305–330, 2005.
- JOHNSON, M.; GRIFFITHS, T. L.; GOLDWATER, S. Adaptor grammars: A framework for specifying compositional nonparametric bayesian models. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2007. p. 641–648.

- JUSCZYK, P. W. How infants begin to extract words from speech. *Trends in Cognitive Sciences*, v. 3, n. 9, p. 323 – 328, 1999. ISSN 1364-6613. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1364661399013637>>.
- KANGAS, J. Phoneme recognition using time-dependent versions of self-organizing maps. In: [*Proceedings*] *ICASSP 91: 1991 International Conference on Acoustics, Speech, and Signal Processing*. [S.l.: s.n.], 1991. p. 101–104 vol.1. ISSN 1520-6149.
- KEOGH, E.; MUEEN, A. Curse of dimensionality. In: *Encyclopedia of machine learning*. [S.l.]: Springer, 2011. p. 257–258.
- KOHONEN, T. Self-organized formation of topologically correct feature maps. *Biological cybernetics*, Springer, v. 43, n. 1, p. 59–69, 1982.
- KOHONEN, T. The self-organizing map. *Proceedings of the IEEE*, IEEE, v. 78, n. 9, p. 1464–1480, 1990.
- KOHONEN, T. The self-organizing map. *Neurocomputing*, v. 21, n. 1-3, p. 1–6, nov 1998. ISSN 09252312. Disponível em: <<http://linkinghub.elsevier.com/retrieve/pii/S0925231298000307>>.
- KOHONEN, T.; SUMERVUO, P. Self-organizing maps of symbol strings with application to speech recognition. In: *Proceedings of WSOM*. [S.l.: s.n.], 1997.
- KRIEGEL, H.-P.; KRÖGER, P.; ZIMEK, A. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM Trans. Knowl. Discov. Data*, ACM, New York, NY, USA, v. 3, n. 1, p. 1:1–1:58, mar. 2009. ISSN 1556-4681. Disponível em: <<http://doi.acm.org/10.1145/1497577.1497578>>.
- LARSEN, E.; CRISTIA, A.; DUPOUX, E. *Relating unsupervised word segmentation to reported vocabulary acquisition*. Open Science Framework, 2017. Disponível em: <osf.io/wa6tq>.
- LIN, J.; KEOGH, E.; LONARDI, S.; PATEL, P. Finding motifs in time series. p. 53–68, 10 2002.
- MACWHINNEY, B. *The chldes project: Tools for analyzing talk*. [S.l.]: Psychology Press, 2000. v. 2.
- MCCLOSKEY, M.; COHEN, N. J. Catastrophic interference in connectionist networks: The sequential learning problem. In: BOWER, G. H. (Ed.). *Academic Press*, 1989, (Psychology of Learning and Motivation, v. 24). p. 109 – 165. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0079742108605368>>.
- MIKKULAINEN, R.; BEDNAR, J. A.; CHOE, Y.; SIROSH, J. *Computational Maps in the Visual Cortex*. [S.l.]: Springer, 2005. v. 1.
- MONAGHAN, P.; CHRISTIANSEN, M. H. Words in puddles of sound: Modelling psycholinguistic effects in speech segmentation. *Journal of child language*, Cambridge University Press, v. 37, n. 3, p. 545–564, 2010.
- MUEEN, A. Time series motif discovery: dimensions and applications. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, John Wiley & Sons, Inc., v. 4, n. 2, p. 152–159, 2014. ISSN 1942-4795. Disponível em: <<http://dx.doi.org/10.1002/widm.1119>>.

- NUNTHANID, P.; NIENNATTRAKUL, V.; RATANAMAHATANA, C. A. Discovery of variable length time series motif. In: IEEE. *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2011 8th International Conference on*. [S.l.], 2011. p. 472–475.
- OLIPHANT, T. E. A guide to NumPy: Open source scientific tools for Python. USA: Trelgol Publishing, 2006. [Online “<http://www.numpy.org/>”; acessado <17-05-2018>].
- PARSONS, L.; HAQUE, E.; LIU, H. Subspace clustering for high dimensional data: A review. *SIGKDD Explor. Newsl.*, ACM, New York, NY, USA, v. 6, n. 1, p. 90–105, jun. 2004. ISSN 1931-0145. Disponível em: <<http://doi.acm.org/10.1145/1007730.1007731>>.
- RÄSÄNEN, O. Computational modeling of phonetic and lexical learning in early language acquisition: Existing models and future directions. *Speech Communication*, Elsevier, v. 54, n. 9, p. 975–997, 2012.
- RATANAMAHATANA, C. A.; KEOGH, E. Three myths about dynamic time warping data mining. In: SIAM. *Proceedings of the 2005 SIAM international conference on data mining*. [S.l.], 2005. p. 506–510.
- REGO, R. L. M. E. do; ARAUJO, A. F. R.; NETO, F. B. de L. Growing self-organizing maps for surface reconstruction from unstructured point clouds. In: *2007 International Joint Conference on Neural Networks*. [S.l.: s.n.], 2007. p. 1900–1905. ISSN 2161-4393.
- SAFFRAN, J. R.; ASLIN, R. N.; NEWPORT, E. L. Statistical learning by 8-month-old infants. *Science*, American Association for the Advancement of Science, v. 274, n. 5294, p. 1926–1928, 1996.
- SAFFRAN, J. R.; NEWPORT, E. L.; ASLIN, R. N. Word segmentation: The role of distributional cues. *Journal of memory and language*, Elsevier, v. 35, n. 4, p. 606–621, 1996.
- SHUMWAY, R. H.; STOFFER, D. S. Time series analysis and its applications. *Studies In Informatics And Control*, Springer, v. 9, n. 4, p. 375–376, 2000.
- TAR, A. Valor de Mercado em Dólar (USD) do Bitcoin - <<https://cointelegraph.com/storage/uploads/view/ab9f11afda0158fd5debbca4ef25e11a.png>>. 2017. Acessado em: 01-05-2018.
- TORKAMANI, S.; LOHWEG, V. Survey on time series motif discovery. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Wiley Online Library, v. 7, n. 2, 2017.
- YINGCHAREONTHAWORNCHAI, S.; SIVARAKS, H.; RAKTHANMANON, T.; RATANAMAHATANA, C. Efficient proper length time series motif discovery. *2013 IEEE 13th International Conference on Data Mining*, p. 1265–1270, 2013.