UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

MARCELY ZANON BOITO

Unsupervised Word Discovery Using Attentional Encoder-Decoder Models

Monografia apresentada como requisito parcial para a obtenção do grau de Bacharel em Ciência da Computação

Trabalho realizado no Institut National Polytechnique de Grenoble - Ensimag dentro do acordo de dupla diplomação UFRGS - INP Grenoble

Orientador: Prof. Dr. Aline Villavicencio
Co-orientador: Prof. Dr. Laurent Besacier

Porto Alegre
2018

RESUMO

Os sistemas neurais "sequência-para-sequência" de tradução automática baseados em atenção (*attentional seq2seq NMT systems*) são capazes de traduzir e alinhar frases em uma língua fonte diretamente em frases em uma língua alvo. Neste trabalho de conclusão de curso, nós utilizamos sequências de símbolos não-segmentados (grafemas ou fonemas) como língua de origem, com o objetivo de explorar um produto do treino desses sistemas, as matrizes probabilísticas de alinhamento suave (*soft-alignment probability matrices*). Investigamos se essas matrizes são capazes de automaticamente descobrir fronteiras de palavras.

Se essa abordagem for bem sucedida, ela pode ser útil como auxílio para a documentação de línguas sem forma escrita e/ou com perigo de desaparecimento. Entretanto, para isso ser possível, os modelos de atenção precisam ser robustos a um cenário escarso de dados, composto por coleções de frases na ordem dos poucos milhares. Neste trabalho, nós utilizamos um corpus paralelo entre uma língua verdadeiramente em perigo de desaparecimento (Mboshi, língua falada em Congo-Brazaville) e o Francês, enfrentando o desafio de treinar uma rede neural com apenas 5.157 frases alinhadas.

Os nossos resultados (BOITO; BESACIER; VILLAVICENCIO, 2017; BOITO et al., 2017) demonstraram que, modificando a ordem de entrada entre as línguas, e assim criando um modelo "palavras-para-símbolos' de tradução automática do Francês para o Mboshi, nós fomos capazes de descobrir automaticamente quase 28% do léxico presente nas frases em Mboshi. Além disso, as matrizes probabilísticas de alinhamento suave geradas apresentaram características interessantes, que podem ser exploradas futuramente para auxílio na descrição de línguas.

Palavras-chave: Descoberta não supervisionada de palavras. documentação computacional de linguas. modelos neurais de tradução automática. modelos neurais de atenção.

Resumo Estendido

Este é um resumo estendido (páginas 5 a 21) em português para a Universidade Federal do Rio Grande do Sul do trabalho original que segue. O trabalho de conclusão original, em inglês, foi apresentado no Institut National Polytechnique de Grenoble, através do programa de dupla diplomação BRAFITEC - EcoSud entre as duas universidades.

1 INTRODUÇÃO

O consenso entre especialistas é que existem aproximadamente 7.000 línguas aos redor do mundo, e entre 50 a 90% dessas línguas estarão extintas até o ano 2100 (AUSTIN; SALLABANK, 2011). Mesmo com o esforço conjunto da comunidade de linguistas, documentar manualmente todas essas línguas antes de sua extinção não é possível.

Assim, o objetivo da documentação computacional de línguas (DCL) é criar ferramentas e metodologias capazes de automatizar a extração de informação léxica, morfológica e sintática de línguas, reduzindo a carga manual do processo de documentação. Exemplos de iniciativas recentes de DCL são a proposição de tarefas coletivas (VERSTEEGH et al., 2016; JANSEN et al., 2013) e os pipelines que partem diretamente do áudio (BESACIER; ZHOU; GAO, 2006; BARTELS et al., 2016; BANSAL et al., 2016; LIGNOS; YANG, 2010; ANASTASOPOULOS; CHIANG, 2017).

É importante que as metodologias propostas considerem a natureza dos dados coletados. Línguas em perigo de desaparecimento são caracterizadas pelo pequeno número de falantes e frequentemente pela falta de uma forma escrita padrão, o que torna a sua documentação desafiadora. Uma alternativa popular é coletar gravações na língua alvo e alinhá-las com suas traduções em uma língua bem documentada. Entretanto, devido à dificuldade de encontrar falantes nativos bilíngues, os corpora coletados tendem a ter tamanho reduzido.

Uma das tarefas envolvidas no processo de documentação é a **segmentação de palavras**. Esta consiste em, dada uma sequência não segmentada de símbolos, encontrar fronteiras capazes de transformá-la em uma sequência de possíveis palavras. A entrada desse sistema pode ser uma sequência de caracteres (chamados de grafemas), fonemas ou até mesmo áudio.

Sistemas de segmentação de palavras podem ser um recurso precioso para linguistas, os auxiliando no processo inicial de transcrição e documentação. Por exemplo, um linguista pode utilizar a saída do sistema como um vocabulário inicial para língua, a partir do qual ele manualmente valida as palavras geradas.

Soluções populares para as tarefas de segmentação e descoberta de palavras inclui Modelos Bayesianos Não Paramétricos (MBNP) (GOLDWATER; GRIFFITHS; JOHNSON, 2009a; ELSNER et al., 2013; ADAMS et al., 2015; ADAMS et al., ; LEE; O'DONNELL; GLASS, 2015) e, mais recentemente, redes neurais (BARTELS et al., 2016; ANASTASOPOULOS; CHIANG, 2017; WANG et al., 2016). Redes neurais também têm sido foco

em tarefas como tradução a partir do áudio (BÉRARD et al., 2016; DUONG et al., 2016) e descoberta não supervisionada de fonemas (FRANKE et al., 2016).

1.1 CONTRIBUIÇÃO

Nesse trabalho, nós apresentamos a primeira tentativa de aplicar uma rede neural de tradução automática baseada em atenção em um cenário de DCL utilizando uma língua verdadeiramente em perigo de desaparecimento. Nós mostramos que é possível, utilizando uma quantidade reduzida de dados, aplicar um pipeline de descoberta não supervisionada de palavras, obtendo um F-score levemente inferior a MBNP, conhecidos por se adequarem muito bem a cenários de escassez de dados.

Além disso, um outro produto do nosso pipeline é o alinhamento de estruturas na língua de interesse com palavras traduzidas, uma informação que pode ser utilizada como ponto inicial na construção de um dicionário bilíngue. Por causa disso, nós preferimos o termo **descoberta de palavras** para definir o nosso método, já que não apenas descobrimos fronteiras entre palavras (como em um método de segmentação), mas também alinhamos os segmentos com os seus possíveis equivalentes em uma língua bem documentada.

Por fim, um outro motivo para o nosso interesse nos modelos neurais utilizados nesse trabalho é a versatilidade desses sistemas. Os modelos sequência-para-sequência são facilmente modificáveis para trabalhar diretamente do áudio, um cenário ideal para DCL.

1.2 MÉTODO

Nós treinamos um modelo neural de tradução automática baseado em atenção e extraímos as matrizes probabilísticas de alinhamento suave geradas pelo mecanismo de atenção. Esses alinhamentos são pós-processados para segmentar uma sequência de símbolos em uma língua desconhecida (no nosso caso, o Mboshi) em palavras.

Nós exploramos três melhorias para o nosso pós-processamento de matrizes: suavização de alinhamento, como apresentado em (DUONG et al., 2016), redução de vocabulário, como discutido em (GODARD et al., 2016), e simetrização do alinhamento gerado. Nós também propomos a inversão da direção de tradução do modelo, utilizando

como fonte as palavras em uma língua bem documentada, e como destino os símbolos em uma língua desconhecida.

Por último, nós também investigamos um cenário semi-supervisionado, onde possuímos algum conhecimento a priori. Nós emulamos esse cenário informando ao nosso sistema a segmentação das 100 palavras mais frequentes em Mboshi.

2 TRABALHOS RELACIONADOS

Modelos Bayesianos Não Paramétricos (MBNP) (GOLDWATER, 2007; JOHNSON; GOLDWATER, 2009) são abordagens estatísticas que podem ser utilizadas para a segmentação de palavras e análise morfológica. O modelo é definido como "não paramétrico" pois não possui limitação quanto ao número de parâmetros, que se adaptam a quantidade de dados. Variantes recentes desses modelos são capazes de trabalhar diretamente do áudio (LEE; O'DONNELL; GLASS, 2015), ou com tradução alinhadas a nível de frase (ADAMS et al., 2015).

A maior vantagem de MBNP para DCL é a robustez de seus modelos em cenários com escassez de dados. Recentemente, Godard et al. (2016) alcançou seus melhores resultados utilizando um MBNP em um subset (1.200 frases) do mesmo corpus utilizado no nosso trabalho. Aplicando o sistema *dpseg*[1] (GOLDWATER; GRIFFITHS; JOHNSON, 2009a), eles recuperaram 23,10% do vocabulário total, alcançando um F-score de 30,48%.

Entretanto, embora MBNP sejam bem estabelecidos na área da descoberta não supervisionada de palavras, nós desejamos explorar como abordagens baseadas em redes neurais poderiam contribuir nesta área de pesquisa. Em particular, modelos estilo **Encoder-Decoder** baseados em atenção têm sido bem sucedidos na tradução automática (BAHDANAU; CHO; BENGIO, 2014), e abordagens fim-a-fim para traduação partindo do áudio têm obtido resultados promissores (BÉRARD et al., 2016; WEISS et al., 2017). Quanto a última, essa abordagem é especialmente interessante para o cenário de DCL, no qual frequentemente corporas são compostas de gravações alinhadas com traduções (sem uma transcrição intermediária).

Assim, como as matrizes probabilísticas geradas pelo modelo de atenção oferecem informação sobre o alinhamento de palavras em redes neurais de tradução automática (BAHDANAU; CHO; BENGIO, 2014; BÉRARD et al., 2016), nós investigamos se isso também é reprodutível em cenários onde a quantidade de dados é limitada. Essa investigação é necessária pois uma desvantagem de abordagens que utilizam redes neurais está nas grandes quantias de dados que são necessários para efetuar o treino (KOEHN; KNOWLES, 2017).

Nós estamos cientes de apenas um outro trabalho utilizando redes neurais de tradução de máquina para a descoberta não supervisionada de palavras no cenário de escassez

[1] Disponível em http://homepages.inf.ed.ac.uk/sgwater/resources.html.

de dados. O trabalho de Duong et al. (2016) usou 18.300 frases em um corpus paralelo entre o Espanhol e o Inglês para emular o comportamento de um corpus de língua com perigo de desaparecimento. A abordagem de descoberta não supervisionada deles é a mais similar à nossa.

Entretanto, nós vamos um passo além: nós aplicamos a técnica em uma língua verdadeiramente em perigo de desaparecimento. Nós trabalhamos com um pouco mais de cinco mil frases em uma língua africana (o Mboshi) sem forma escrita padrão, e nós acreditamos que esse cenário é mais representativo do que linguistas podem encontrar no processo de documentação de línguas.

3 METODOLOGIA

3.1 Corpus paralelo Mboshi-Francês

Nós utilizamos 5.157 frases do corpus paralelo em Mboshi (Língua Bantu C25), uma língua africana sem forma escrita[1], alinhadas a nível de frase com traduções em Francês. Mboshi é uma língua falada no Congo-Brazzaville, e possui 32 fonemas diferentes (25 consoantes e 7 vogais) e dois tons (alto e baixo). O corpus foi gravado utilizando a ferramenta LIG-AIKUMA (BLACHON et al., 2016) durante o projeto BULB (ADDA et al., 2016).

Para cada frase, possuímos uma transcrição de grafemas (padrão ouro para a segmentação), a versão não segmentada dessa transcrição, a tradução em Francês, a forma lematizada[2] dessa tradução, e o áudio. É importante mencionar que para esse trabalho, nós utilizamos a transcrição de grafemas (próximos à fonologia da língua) como a nossa fonte, deixando o uso de áudios para trabalhos futuros.

O corpus foi separado entre conjunto de treino e desenvolvimento (ou validação), onde apenas 10% foram reservados para o conjunto de desenvolvimento. A Tabela 3.1 provê um resumo de *types* (palavras únicas) e *tokens* (número total de palavras) para cada porção do corpus paralelo.

Tabela 3.1: Organização do corpus em conjuntos desenvolvimento (Dev, 514 frases) e treino (Train, 4.643 frases).

	# types	**#tokens**	**# tokens por frase (média)**
Dev Mboshi	1.324	3.133	6,0
Dev French	1.343	4.321	8,2
Train Mboshi	6.245	27.579	5,9
Train French	4.903	38.226	8,4

3.2 Arquitetura Neural

Nós utilizamos o sistema **encoder-decoder** LIG-CRIStAL NMT[3], usando texto não segmentado como entrada de treino. Nossos modelos seguem Bahdanau, Cho and

[1] Mesmo não possuindo uma forma escrita padrão, linguistas providenciaram transcrições de grafemas próximas à fonologia da língua.

[2] Para tokenização e lematização, foi utilizada a ferramenta TreeTagger (SCHMID, 2013).

[3] Disponível em https://github.com/eske/seq2seq.

Bengio (2014), e podem facilmente ser estendidos para trabalhar diretamente do áudio (BÉRARD et al., 2016). A intuição por traz de sistemas **encoder-decoder** é: a rede neural **encoder** vai aprender a codificar a entrada na língua origem, enquanto a rede neural **decoder** vai aprender a decodificar essa informação codificada, gerando uma saída na língua destino.

Um **encoder** bidirecional lê a sequência de entrada $x_1, ..., x_A$ e produz uma sequência de estados (encoder states) $\mathbf{h} = h_1, ..., h_A \in R^{2\times n}$, onde n é o tamanho da célula do encoder. O **decoder** usa seu estado atual s_t e o mecanismo de atenção para gerar o próximo símbolo de saída z_t.

A cada *time step* t, o decoder calcula a distribuição de probabilidade sobre o vocabulário destino (Equações 3.1 e 3.2). Então, ele gera o símbolo z_t cuja probabilidade é a maior, parando apenas ao encontrar um token especial de final de frase (Equação 3.3). O decoder atualiza o seu estado s_t com o token gerado z_t (Equação 3.4). No nosso caso, como as traduções de referência estão sempre disponíveis (mesmo no momento de teste), nós utilizamos um processo de *force feeding* do símbolo de referência anterior w_t, ao invés de utilizar o último símbolo z_t gerado (esse método é conhecido como *teacher forcing*).

$$c_t = \text{attn}(\mathbf{h}, s_{t-1}) \tag{3.1}$$

$$y_t = \text{output}(s_{t-1} \oplus E(w_{t-1}) \oplus c_t) \tag{3.2}$$

$$z_t = \arg\max y_t \tag{3.3}$$

$$s_t = \text{LSTM}(s_{t-1}, E(w_t) \oplus c_t) \tag{3.4}$$

Nas equações acima, $\oplus$ é o operador de concatenação, s_0 é inicializado com o último estado do encoder (após uma transformação não linear), $z_0 =$ <BOS> (token especial de início de frase) e $E \in R^{|V|\times n}$ é a matriz de *embeddings* alvo. A função de $output$ usa uma camada *maxout*, seguida de uma projeção linear no tamanho do vocabulário dado por $|V|$.

A atenção (Equação 3.1) é definida como segue:

$$c_t = \text{attn}(\mathbf{h}, s_t) = \sum_{i=1}^{A} \alpha_i^t h_i \tag{3.5}$$

$$\alpha_i^t = \text{softmax}(e_i^t) \tag{3.6}$$

$$e_i^t = v^T \tanh\left(W_1 h_i + W_2 s_t + b_2\right) \tag{3.7}$$

Onde v, W_1, W_2, e b_2 são variáveis cujo aprendizado acontece simultaneamente ao apren-

dizado dos outros parâmetros do modelo. A cada time step t, um valor de pontuação e_i^t é computado para cada estado h_i do encoder, utilizando para isso o estado atual s_t do decoder. Essas pontuações são então normalizadas utilizando a função $softmax$, que as transforma em uma distribuição de probabilidade com respeito à sequência de entrada $\sum_{i=1}^{A} \alpha_i^t = 1$ e $\forall i, 0 \leq \alpha_i^t \leq 1$.

O vetor de contexto c_t utilizado pelo decoder é uma soma ponderada dos estados do encoder, e podemos interpretá-lo como uma forma de destacar as partes do encoder que o modelo considera **relevantes** para a geração do próximo símbolo de saída z_t. Os pesos α_i^t podem ser vistos como um **alinhamento suave** entre a entrada x_i e a saída z_t.

Nossos modelos são treinados utilizando o algorítmo Adam, com uma *learning rate* de 0.001 e *batch size* N de 32 (Equação 3.8). Nós minimizamos a *cross-entropy loss* entre a distribuição de probabilidade de saída p_t (Equação 3.10) e a tradução de referência w_t (Equação 3.9).

$$L = \frac{1}{N} \sum_{i=1}^{N} \text{loss}(s_i = w_1, ..., w_T \mid \mathbf{x_i}) \tag{3.8}$$

$$\text{loss}(w_1, .., .w_T \mid \mathbf{x_i}) = -\sum_{t}^{T} \sum_{j}^{|V|} \log p_{tj} \times 1(w_t = V_j) \tag{3.9}$$

$$p_{tj} = softmax(y_t) = \frac{e^{y_{tj}}}{\sum_{k}^{|V|} e^{y_{tk}}} \tag{3.10}$$

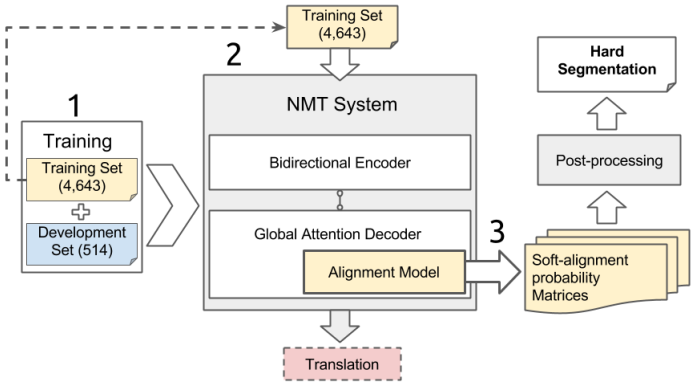
3.3 Modelos Neurais de Descoberta Não Supervisionada de Palavras

O nosso pipeline completo de descoberta de palavras está ilustrado na Figura 3.1. Primeiramente, nós treinamos o sistema NMT (seção 3.2) utilizando o corpus paralelo Mboshi-Francês (seção 3.1), sem o uso de fronteiras entre os símbolos em Mboshi. Isso é representado pelo passo 1 na figura.

Nós treinamos o modelo utilizando um número fixo de *epochs*, e o modelo selecionado é aquele que possui melhor *BLEU score* (PAPINENI et al., 2002), que é avaliado sobre o conjunto de desenvolvimento (ou validação). Através de experimentação, observamos que esse setup é o que fornece informações mais precisas de alinhamento entre línguas fonte e destino. Após o treino do modelo, nós reinserimos o conjunto de treino, utilizando o modelo para extrair matrizes probabilísticas de alinhamento suave que são calculadas pelo modelo de atenção (passo 2).

Por último, nós pós-processamos as matrizes geradas no passo 2 e inferimos ali-

Figura 3.1: Pipeline completo da descoberta não supervisionada de palavras.



nhamento a nível de palavra (passo 3). Nós transformamos a informação de alinhamento suave em alinhamento definitivo: alinha-se cada símbolo alvo x_i com a palavra destino w_t tal que $t = \arg\max_i \alpha_i^t$. Então, nós segmentamos a sequência de entrada (Mboshi) de acordo com o resultado da matriz de alinhamento definitivo: se dois símbolos consecutivos estão alinhados com a mesma palavra em Francês, nós consideramos que eles pertencem a mesma palavra em Mboshi.

4 DESCOBERTA NÃO SUPERVISIONADA DE PALAVRAS

Para os experimentos de descoberta não supervisionada de palavras, nós utilizamos transcrições (não segmentadas) em Mboshi[1] feitas por linguistas. Estas estão alinhadas com frases em Francês. Nós experimentamos com as seguintes variações:

1. **Suavização de alinhamento**: Para lidar com a discrepância que existe entre o tamanho das frases na língua fonte (fonemas ou grafemas) e na língua destino (palavras), nós encorajamos alinhamentos do tipo muitos-para-um entre Mboshi e Francês. Esses alinhamentos ajudam a clusterizar símbolos em Mboshi em "unidades de palavra". Nós implementamos a suavização de alinhamento proposta por Duong et al. (2016).

   Neste trabalho, a função $softmax$ utilizada pelo mecanismo de atenção utiliza um parâmetro adicional de temperatura: $\alpha_i^t = \exp(e_i^t/T) / \sum_j \exp(e_j^t/T)$. Quando a temperatura T é maior que um[2], isso resultará em um softmax mais "suave", que encoraja alinhamentos do tipo muitos-para-um. Além disso, as probabilidades das matrizes resultantes também são suavizadas através do uso de um filtro equivalente ao *low-pass*. A probabilidade α_i^t passa a ser a média entre o seu valor e a sua vizinhança imadiata à direita e à esquerda: $\alpha_i^t \leftarrow (\alpha_{i-1}^t + \alpha_i^t + \alpha_{i+1}^t)/3$.

2. **Arquitetura inversa**: Em modelos NMT, os alinhamentos suaves são criados de uma maneira a forçar as probabilidades de cada palavra destino t a somar para um (ou seja, $\sum_i \alpha_i^t = 1$). Isso força o modelo de alinhamento a utilizar todas as palavras disponíveis na frase destino. Como discutido em Duong et al. (2016), não existe uma limitação similar para os símbolos fonte.

   Como estamos mais interessados na produção de alinhamentos do que na tradução, nós propomos inverter a arquitetura. O modelo inverso traduz palavras em Francês em símbolos em Mboshi, o que previne que o modelo de atenção ignore símbolos em Mboshi.

3. **Fusão de alinhamentos**: Modelos estatísticos de tradução automática, como o Moses (KOEHN et al., 2007), extraem alinhamentos em duas direções (origem-para-destino e destino-para-origem) e então fundem os dois, criando o modelo de tradução final. Essa fusão de alinhamento é seguidamente chamada de "simetrização".

   Nós investigamos uma simetrização similar: calcula-se a média das matrizes pro-

[1] O conjunto de símbolos diferentes para as transcrições não segmentadas é de 44 símbolos.

[2] Nós utilizamos $T = 10$ como no artigo original (DUONG et al., 2016).

babilísticas de alinhamento suave geradas pelo modelo base (Mboshi-Francês) e inverso (Francês-Mboshi). Cada probabilidade de alinhamento α_i^t é substituída por $\frac{1}{2}(\alpha_i^t + \beta_t^i)$, onde β_t^i é a probabilidade do mesmo alinhamento $i \leftrightarrow t$ na arquitetura inversa.

4. **Redução de vocabulário de língua destino**: Reduzimos o tamanho do vocabulário da língua bem documentada (Francês) substituindo as palavras por seus lemas. A hipótese é que a simplificação da informação de tradução pode facilitar o aprendizado do alinhamento entre as duas línguas. Para a tarefa de descoberta não supervisionada de palavras, essa técnica foi recentemente investigada por Godard et al. (2016).

O modelo base (Mboshi-Francês) utiliza embeddings e tamanho de célula de 12. O encoder empilha duas camadas (LSTM) bidirecionais, e o decoder usa uma única camada (LSTM). O modelo inverso (Francês-Mboshi) utiliza embeddings e tamanho de célula de 64, com uma camada bidirecional no encoder e uma camada no decoder (ambas LSTM).

Na Tabela 4.1, apresentamos os resultados para a tarefa de descoberta não supervisionada de palavras obtidos com o nosso modelo base, inverso, com e sem suavização de alinhamento. Notamos que a técnica de suavização de alinhamento apresentada por Duong et al. (2016) melhorou os resultados, especialmente para types.

Tabela 4.1: Resultados para a descoberta não supervisionada de palavras utilizando 4.643 frases.

	TOKENS			**TYPES**		
	Recall	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
Modelo Base (Mb-Fr)	7,16	4,50	5,53	12,85	6,41	8,55
Modelo Base (Mb-Fr) com suavização	6,82	5,85	6,30	15,00	6,76	9,32
Modelo Inverso (Fr-Mb)	20,04	10,02	13,36	18,62	14,80	16,49
Modelo Inverso (Fr-Mb) com suavização	21.44	16.49	18.64	27.23	15.02	19.36

Assim, demonstramos que a proposição de inverter o modelo melhora consideravelmente a descoberta de types e tokens. Isso parece confirmar a nossa hipótese de que inverter o alinhamento resulta em uma melhor segmentação. Neste cenário, o modelo de atenção alinha cada símbolo em Mboshi a palavras em francês com a soma das probabilidades para esses símbolos resultando em um. Entretanto, isso também pode ser uma consequência do fato do modelo inverso ler palavras e retornar símbolos, o que é normalmente mais fácil do que ler uma sequência de símbolos (LEE; CHO; HOFMANN, 2016).

De toda forma, nós alcançamos o nosso melhor resultado utilizando o modelo inverso com a suavização de alinhamento (última linha da Tabela 4.1). **Nós utilizamos esse modelo para os resultados apresentados no restante desse resumo estendido.**

Para a fusão de alinhamentos, nós testamos três configurações para as matrizes geradas pelos modelos base e inverso. Nós realizamos a fusão das matrizes sem suavização de alinhamento, a fusão de matrizes suavizadas, e a fusão de matrizes suavizadas seguida por mais uma fase de suavização. Todas as configurações resultaram em resultados piores: redução de recall entre 3% e 5% para tokens e entre 1% e 9% para types.

Acreditamos que isso acontece pois a diferença de performance do modelo base e o inverso é grande. O modelo base não possui a restrição quanto ao alinhamento de símbolos em Mboshi, e isso acaba degradando mais os alinhamentos do que adicionando informação.

Por último, nós testamos a redução de vocabulário em Francês. O modelo treinado com lemas teve uma queda de recall de aproximadamente 2% para tokens e types. Acreditamos que isso pode ser devido à natureza do Mboshi, e não necessariamente um resultado generalizável. A língua Mboshi tem uma morfologia rica, criando uma palavra diferente para cada flexão verbal através da modificação do radical. Assim, removendo flexão verbal das traduções em Francês, podemos ter dificultado a tarefa de alinhamento entre as duas línguas.

5 DESCOBERTA SEMI-SUPERVISIONADA DE PALAVRAS

A tarefa de documentação de línguas é raramente completamente não supervisionada, já que linguistas costumam interagir com a comunidade cuja língua eles estão documentando. Para emular esse conhecimento a priori, nós selecionamos as 100 palavras mais frequentes no nosso padrão ouro em Mboshi.

Consideramos essa quantidade de palavras razoável para representar a informação que um linguista poderia adquirir após alguns dias em contato com a comunidade. A intuição é que, provendo essa informação de segmentação para o sistema, podemos melhorar a sua performance na descoberta do restante do vocabulário.

Foram adicionadas 100 palavras (types) mais frequentes na versão não segmentada do Mboshi, criando uma representação mista, onde a entrada é uma coleção de grafemas e palavras. Para a avaliação, como línguas seguem distribuições Zipfianas (POWERS, 1998) e nós informamos o modelo sobre as palavras mais frequentes no corpus, analisamos apenas os resultados de types. Resultos são apresentados na Tabela 5.1.

Tabela 5.1: Resultados da descoberta semi-supervisionada de palavras (100 palavras conhecidas, 4.653 frases).

	Não supervisionado	**Semi-supervisionado**
Recall	27,23	29,49
Precision	15,02	24,64
F-score	19,36	26,85
# types corretos	1.692	1.842
# types gerados	11.266	7.473

Para types, observamos um aumento de 2,4% em recall, o que não é uma grande melhoria, considerando que demos 100 palavras ao modelo. O nosso modelo não supervisionado já era capaz de descobrir 97 entre essas 100 palavras, o que poderia justificar a diferença pequena de performance. Além das 100 palavras já conhecidas, o modelo semi-supervisionado encontrou 50 outras palavras que o modelo não supervisionado foi incapaz de descobrir.

É interessante analisar que, mesmo o aumento de performance tendo sido pequeno, o modelo semi-supervisionado reduziu consideravelmente o número de types gerado (de 11.266 para 7.473). Isso sugere que a informação adicional provida pode ter ajudado o modelo a alcançar uma melhor reprsentação de vocabulário, mais próxima ao vocabulário do padrão ouro.

6 ANÁLISE

6.1 Comparação com Baseline

Como baseline foi utilizado o $dpseg$ (GOLDWATER; GRIFFITHS; JOHNSON, 2006; GOLDWATER; GRIFFITHS; JOHNSON, 2009b). Os hiper-parâmetros utilizados[1] foram os mesmos utilizados por Godard et al. (2016). Eles os obteram após experimentação em um corpus de grande tamanho em Inglês, com aplicação posterior (bem sucedida) na segmentação do Mboshi.

A Tabela 6.1 mostra os nossos resultados para types, os comparando com os resultados do modelo não paramétrico. Embora o nosso método seja capaz de recuperar mais do vocabulário, o modelo bayesiano possui maior precisão, e assim, ambos são bastante próximos em termos de F-score. O nosso método, entretanto, tem a vantagem de produzir indicativos de tradução, enquanto $dpseg$ utiliza apenas a porção Mboshi do corpus paralelo.

Tabela 6.1: Comparação entre dpseg e o modelo inverso com suavização de alinhamento na descoberta não supervisionada de palavras. Os resultados são a média de três execuções.

	Recall	**Precision**	**F-score**
Modelo Inverso (Francês-Mboshi)	27.23	15.02	19.36
dpseg (MBNP)	13.94	38.32	20.45

É interessante notar que o nosso método, baseado em redes neurais e não é especializado para essa tarefa (os alinhamentos suaves são apenas um subproduto do pipeline de tradução), foi capaz de alcançar uma performance próxima ao $dpseg$, que é reconhecido como excelente em cenários de escassez de dados. Isso ilustra o potencial da nossa abordagem para DCL.

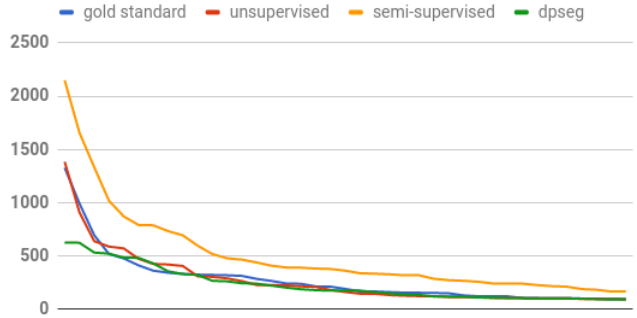
6.2 Análise do Vocabulário Gerado

Para melhor compreendermos o comportamento de segmentação da nossa abordagem, analisamos o vocabulário gerado. Nós comparamos os nossos modelos não supervisionado e semi-supervisionado com o padrão ouro e com o $dpseg$. A primeira característica analisada foi a distribuição da frequência de palavras nos vocabulários gerados.

[1] Também utilizamos inicialização aleatória e 19.600 iterações de amostragem.

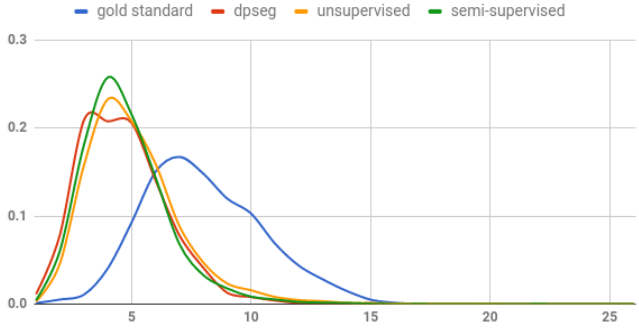
Embora já saibamos que $dpseg$ força o vocabulário gerado a seguir um comportamento exponencial, nós observamos que as nossas abordagens também resultaram nesse comportamento. Nossos métodos produzem curvas tão próximas ao padrão outro quanto o $dpseg$ (ver Figura 6.1).

Figura 6.1: Distribuição da frequência de palavras nos três modelos e no padrão ouro.



Também foi medido o tamanho médio das palavras com o objetivo de identificar sub-segmentação e segmentação excessiva nos diferentes métodos. Para podermos comparar vocabulários de tamanhos diferentes, nós normalizamos as frequências pelo número total de types gerados. As curvas são apresentadas na Figura 6.2. Lendo a legenda da esquerda para a direita, os vocabulários são respectivamente de 6.245, 2.285, 11.266, e 7.473 palavras.

Figura 6.2: Distribuição de tamanho de palavras nos diferentes métodos.



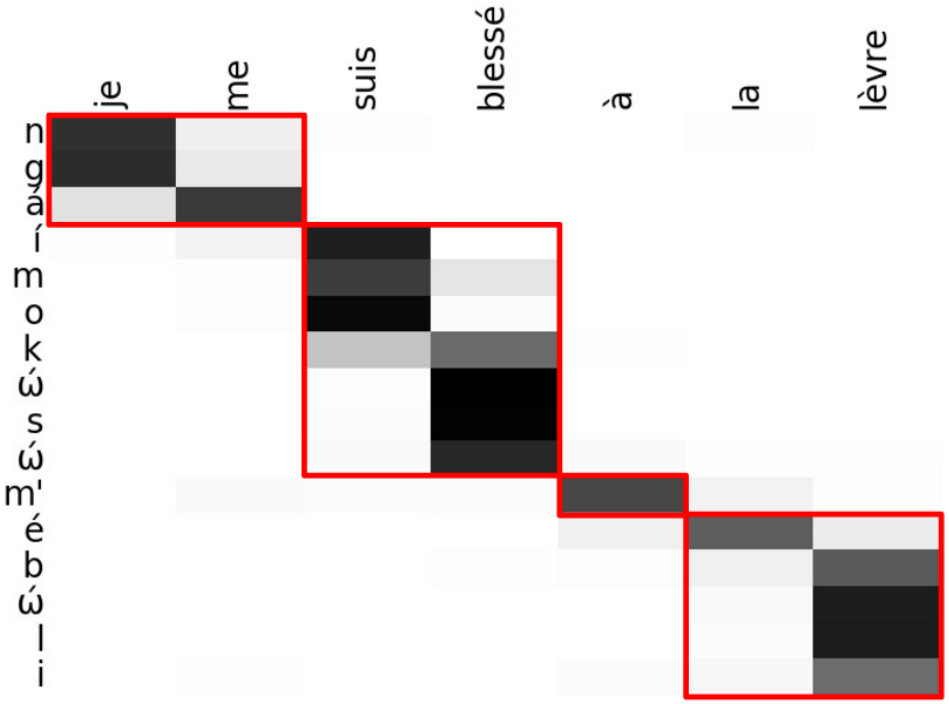
Nosso modelo semi-supervisionado é o mais próximo do vocabulário real em termos de tamanho, com apenas 1.228 palavras a mais. Todas as abordagens (incluindo $dpseg$) segmentam excessivamente a entrada de maneira similar, criando vocabulários com tamanho de palavra média inferior ao padrão ouro e com um maior número de types.

Como tanto $dpseg$ como a abordagem baseada em redes neurais sofrem do mesmo problema de segmentação excessiva, nós acreditamos que isso pode também ser uma consequência do corpus utilizado para treino, e não necessariamente uma característica generalizável do nosso método em cenários de escassez de dados. Para os nossos modelos neurais, outra justificativa é que, com o corpus sendo pequeno e a média de tokens por

sequência sendo maior no lado Francês (mostrado na Tabela 3.1), isso pode resultar numa dispersão dos alinhamentos entre as possíveis traduções, resultando em fronteiras adicionais (segmentação excessiva).

O Mboshi é uma língua aglutinativa, e através de análise manual, nós percebemos que houveram vários casos em que havia um bom alinhamento, mas não uma boa segmentação. Um exemplo está na Figura 6.3, onde percebemos que a palavra "ímokώsώ" teve que ser separada em duas para manter o alinhamento com as duas partes da sua tradução (correta) "suis blessé". Esse também é o caso da última palavra da imagem: o Mboshi não necessita de artigos precedendo seus substantivos, o que causou o desalinhamento.

Figura 6.3: Examplo de alinhamento suave resultante do nosso modelo de descoberta não supervisionada de palavras. Quanto mais escuro o quadrado, maior é a probabilidade associada ao par. A nossa segmentação para esse exemplo foi "ngá ímo kώsώ m' é bώli", e a correta era "ngá ímokώsώ m' ébώli".



Acreditamos que, explorando esse alinhamento com a tradução, podemos restringir o processo de segmentação, criando um modelo de descoberta de palavras mais eficaz. Em resumo, nós fomos capazes de criar um modelo de qualidade razoável, que resulta em segmentação e informação de alinhamento, utilizando apenas 5.157 frases para treino (cenário de escassez de dados).

7 CONCLUSÃO

Neste trabalho, nós apresentamos uma abordagem baseada em redes neurais para realizar a descoberta de palavras em cenários de escassez de dados. Nós utilizamos um modelo NMT com atenção global para recuperar matrizes probabilísticas de alinhamento suave, e nós as utilizamos para segmentar a língua alvo da documentação. Uma abordagem similar foi apresentada em Duong et al. (2016), mas esse trabalho apresenta a primeira tentativa de treinar uma rede neural utilizando uma língua sem forma escrita padrão e um conjunto de dados pequeno, feito de apenas 5.157 frases alinhadas.

Invertendo a entrada do nosso sistema e aplicando a suavização de alinhamento, nós fomos capazes de recuperar 27.23% do vocabulário, o que resultou em um F-score próximo ao do baseline bayesiano $dpseg$, conhecido como robusto em ambientes de escassez de dados. Além disso, a nossa abordagem possui a vantagem de incorporar naturalmente a tradução, o que pode ser utilizado futuramente para melhorar a segmentação e/ou criar um dicionário bilíngue. O nosso sistema também pode ser facilmente estendido para trabalhar diretamente a partir do áudio, uma necessidade para muitos cenários de DCL.

Por fim, como trabalho futuro, o nosso objetivo é descobrir vocabulário diretamente do áudio, inspirado nas arquiteturas encode-rdecoder apresentadas em Bérard et al. (2016) e Weiss et al. (2017). Nós também queremos explorar funções objetivo de treino mais relacionadas com a tarefa em questão.

Master of Science in Informatics at Grenoble
Master Mathématiques Informatique - spécialité Informatique
option Artificial Intelligence and Web

Unsupervised Word Discovery using Attentional Encoder-Decoder Models

Marcely Zanon Boito

June 23, 2017

Research project performed at LIG

Under the supervision of:
Laurent Besacier

Aline Villavicencio
Defended before a jury composed of:
Arnaud Legrand
Jerome Euzenat
Eric Gaussier

June 2017

Abstract

Attention-based sequence-to-sequence neural machine translation systems have been shown to jointly align and translate source sentences into target sentences. In this project, we use unsegmented symbol sequences (unsegmented graphemes or phonemes) as source, aiming to explore if soft-alignment probability matrices generated during training can be used to automatically discover word boundaries.

If successful, such approach could be useful for documenting unwritten and/or endangered languages. However, for this to be feasible, attention models should be robust to low-resource scenarios, composed by several thousand of sentences only. We use a parallel corpus between a truly endangered language (Mboshi spoken in Congo-Brazaville) and French, tackling the challenge of training a neural system with only 5,157 aligned sentences.

Our results, which were partially reported in [36], showed that by reverting the input order and creating a words-to-symbols French-Mboshi translation system, we were able to automatically discover almost 28% of the Mboshi vocabulary. Moreover, the generated soft-alignment probability matrices presented interesting features that could be further exploited for language description.

Résumé

Les systèmes séquence-à-séquence de machine de traduction neurale basés en attention ont démontré leur capabilités de conjointement aligner et traduire des phrases sources en phrases cibles. Dans ce projet, nous utilisons comme source des séquences de symboles non-segmentées (graphèmes ou phonèmes non-segmentés), afin d'explorer les matrices de probabilité de soft-alignment qui sont générées pendant le processus d'entraînement dans le but de découvrir s'elles sont exploitables pour définir automatiquement les frontières entre les mots.

Si possible, cette approche peut être utile pour des fins de documentation de langues non-écrites et/ou en voie de disparition. Néanmoins, la faisabilité de cette approche est restreinte par la condition de robustesse des modèles d'attention dans scénarios de ressources faibles, qui sont scénarios dans l'ordre de milliers de phrases seulement. Pour le cas étudié, un corpus parallèle, entre une vraie langue en voie de disparition (Mboshi, parlé à Congo-Brazaville) et le Français, est utilisé pour relever le défis d'entraîner un système de neurones avec seulement 5,157 phrases alignées.

Nos résultats, qui ont été rapportés partiellement en [36], nous ont montré que, utilisant un système inverse de traduction du type mots-vers-symboles entre Français et Mboshi, nous sommes capables de récupérer automatiquement presque 28% du vocabulaire en Mboshi. De plus, les matrices de probabilité de soft-alignment produites presentent des caracteristiques interessantes, qui peuvent être exploitées pour faire de la description de langage.

Acknowledgement

I would like to express my gratitude to my home institution UFRGS (*Universidade Federal do Rio Grande do Sul*), my host institution Grenoble INP (ENSIMAG), the MoSIG program, the LIG (*Laboratoire d'Informatique de Grenoble*), and these institutions directors for this great opportunity. I'm sincerely grateful to my two supervisors: Laurent Besacier and Aline Villavicencio, for their support and guidance through the project.

I want to thank the amazing support that I received from friends and family during these two years of my double degree exchange program. In special, I leave a huge "thank you" to Francieli Zanon Boito and Laércio Lima Pilla for always being an inspiration source for me, from day one studying computer science.

Contents

— 1 —
Introduction

In this chapter we introduce the scope of this work: unsupervised word discovery for computational language documentation. We describe the project approach and review the achieved results. Finally, we offer a summary of all the information present in the chapters.

1.1 Computational Language Documentation

The general consensus between specialists is that there are approximately 7,000 languages currently spoken in the world, and 50 to 90% of them will become extinct by the year 2100 [6]. Even with a joint effort from the linguistics community, manually documenting all these languages before their extinction is not feasible. Recently, initiatives for tackling this issue include organizing tasks [34, 23] and offering tools and methodologies to help to automate (part of) this documentation process.

Endangered languages are characterized by the small number of speakers and often by the lack of a well-defined written form, which makes their documentation an even more challenging task. To deal with the absence of standard written form, collected speech can be aligned to its translation in a well-documented language. The resulting parallel corpora, though, usually lack size.

Nonparametric bayesian models and neural network systems emerged as popular approaches for phonetic unit discovery, unsupervised word discovery and lexicon discovery, common subtasks of the documentation process. The project presented here is a neural approach for unsupervised word discovery.

The task of unsupervised word discovery[1] consists of: given an unsegmented input, to find the boundaries between the word-like units. This input can be a sequence of characters, phonemes or even speech. For accomplishing this, an unsupervised word discovery system receives unsegmented input and analyzes it, generating the words the system believes to belong to the language vocabulary.

In language documentation scenarios, the focus of this project, we often work with speech or phoneme transcription, and from this information, our goal is to discover words in the language we wish to document. This technique can be an important resource for linguists, since it can help them to start the transcription and documentation process. For instance, the linguist can use the system's output as initial clue for the vocabulary, validating or not the generated words.

[1] We will use alternatively the terms *discovery* or *segmentation* in this thesis.

1.2 Project Definition

Collected data for endangered languages usually lack size, and often rely on translations instead of transcriptions. Because of that, language documentation systems must be able to process input in the form of symbol chains (such as characters or phoneme transcriptions) and/or speech. For compensating the lack of data inherent to these scenarios, it is also desirable for these systems to be able to take advantage of parallel information (usually sentence-level aligned translations).

Meanwhile, in the neural machine translation (NMT) field, attentional encoder-decoder architectures have been shown to jointly learn to align and translate source symbols into target sentences. Their structure makes them naturally fit to deal with parallel corpora, and they are built in a way that allows them to easily work with different input representations.

Aiming to study the possible contributions that these systems could have in the language documentation scenario, in this project we use an NMT system, robust to different input representations, to segment and align an endangered language with its translation. We do not focus on creating a good translation system, since in low-resource scenarios we often lack data for accomplishing this. Instead, we focus on investigating if we have enough data to create a meaningful internal representation of the unsegmented sentences used for training.

Linguistic information can be conveyed by hidden layers or, in the case of attention-based NMT systems, by the alignment models learned. In this project, we will mostly analyze alignment models through the generated soft-alignment probability matrices. We will investigate if these matrices could help us to find boundaries in unsegmented symbol chains, and consequently, to help us to discover words. The proposed approach is illustrated in Figure 1.1.

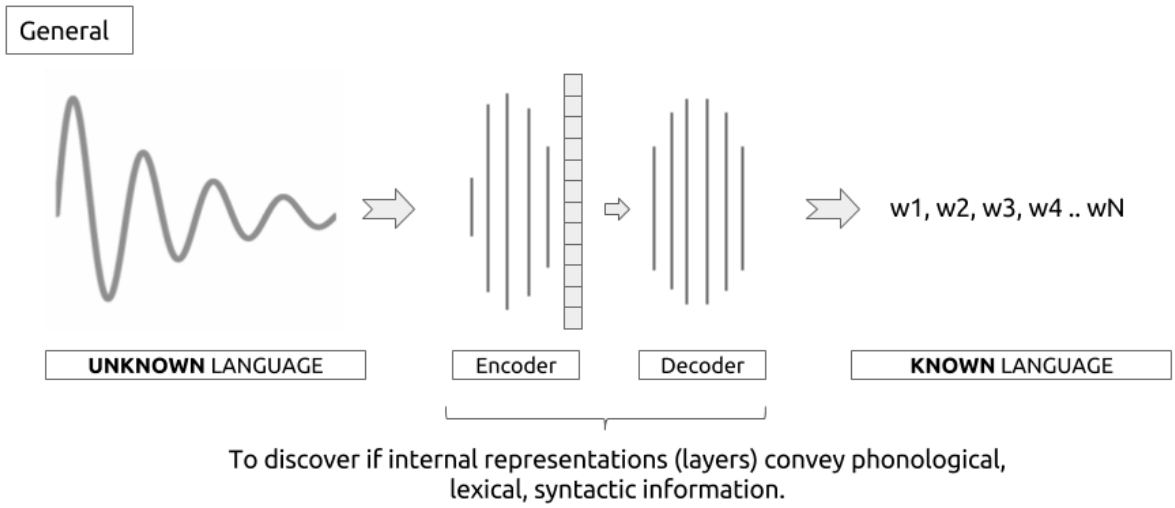


Figure 1.1 – General task definition. This approach starts from speech in an unknown language, and investigates the potential of the internal representations learned by encoder-decoder NMT systems. The ultimate goal is to retrieve words from signal, together with their translations.

For this thesis, we approach the language documentation task using a real endangered language. We use a 5,157 sentences corpus in an unwritten African language called Mboshi, aligned at sentence-level with translations in French. As input, we work with unsegmented

grapheme transcriptions, since for the Mboshi language they are close to phoneme transcriptions. Working directly on speech signal is left as future work.

We investigate if it is possible to discover words from this unsegmented grapheme input supported only by the soft-alignment information generated by the neural system. For accomplishing so, we experimented creating both Mboshi-French symbols-to-words and French-Mboshi words-to-symbols translation systems. The general schema for the task performed is illustrated in Figure 1.2.

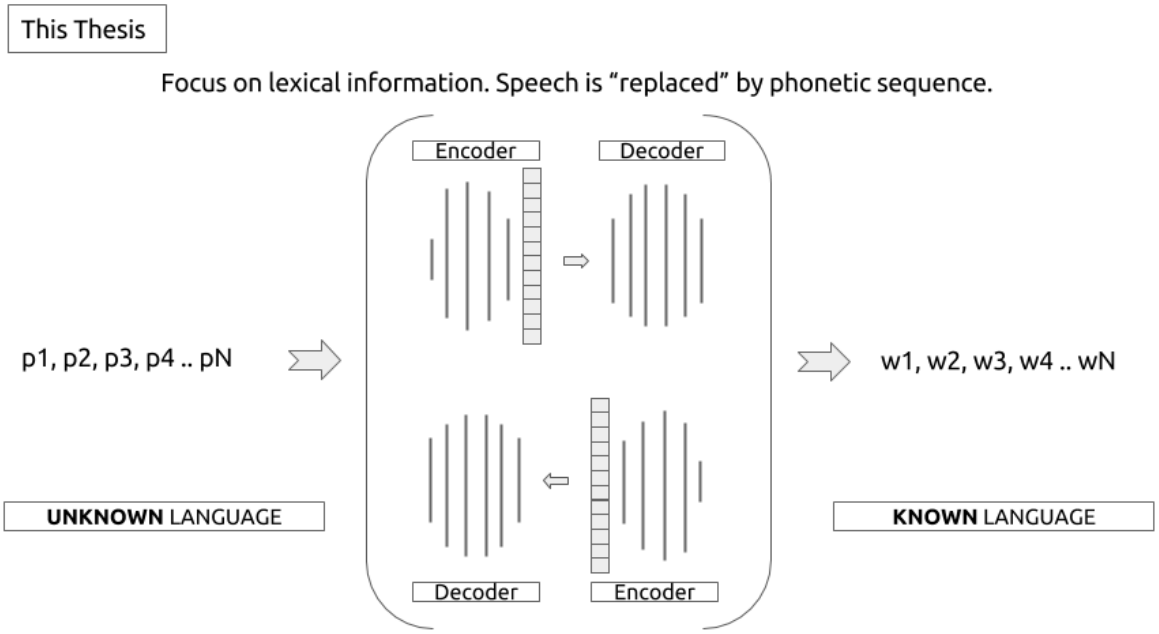


Figure 1.2 – A general schema for the task performed in this thesis. We use the Mboshi-French parallel corpus to train encoder-decoder NMT systems in both directions, aiming to retrieve the soft-alignment information learned during training procedure. Our goal is to use this information to segment our input, discovering words.

If successful, such approach would be useful for documenting unwritten and/or endangered languages. However, for this to be feasible, attention model's soft-alignment matrices should be robust to low resource scenarios of several thousand of sentences only.

To test our hypothesis, we use the LIG encoder-decoder NMT system with global attention [7] whose implementation[2] was made available by Alexandre Bérard in [11]. We use this system to work from unsegmented text input. However, as it is shown in [11], the model is easily extendable for working directly from speech.

We also investigate some optimizations for the unsupervised word discovery system, such as the alignment smoothing presented by Duong et al. (2016) [16], and the target information reduction presented by Godard et al. (2016) [19]. Lastly, we also look at Moses-like symmetrization [26] using the retrieved alignments for both directions (Mboshi-French and French-Mboshi).

[2]Available at https://github.com/eske/seq2seq, branch baseline.

1.2.1 Project Relevance

We already know that soft-alignment probability matrices offer accurate information about word alignments in NMT systems [7, 11]. However, we do not know if this is reproducible in scenarios limited by the amount of data available.

Duong et al. (2016) [16] is the only other work that we are aware of that approached unsupervised word segmentation supported only by the soft-alignment probability matrices. They used an 18,300 sentences corpus in two well-documented languages to simulate the endangered language documentation scenario.

In this project we take one step further, going to a real documentation language scenario, working with only five thousand sentences in an unwritten African language. We believe this is more representative of what linguists may encounter documenting languages. In practice, our goal is to answer if this approach is indeed feasible for language documentation, and to understand its advantages and drawbacks.

Although nonparametric bayesian models have been reaching the best results for the word discovery task in small data sets, they are monolingual techniques. In comparison to that, our approach makes use of the translation information present in our parallel corpus. This way, our model jointly generates segmentation and alignment (translation) information, which could be helpful to decipher languages unknown to the field linguist.

To summarize, this work is the first attempt to jointly segment and align an unsegmented symbol sequence in a truly endangered language, using attentional encoder-decoder models.

1.3 Evaluation and Results

We tested our hypothesis of the soft-alignment probability matrices usefulness by training several encoder-decoder NMT systems using the Mboshi-French parallel corpus. We varied embedding size and number of hidden layers, and we evaluated the trained models by their results for the unsupervised word discovery task (recall, precision and F-score).

By training an initial model using the gold segmentation for the Mboshi sentences, we verified that the approach was able to retrieve useful information from the soft-alignment probability matrices, even with few training examples. The best model had the embedding size of 32, and used one single layer for encoder and decoder networks. From the small test set manually created using our mini Mboshi-French dictionary [9], this model correctly aligned 66.17% of the 68 known word-translation pairs.

After our first attempt to train a model from an unsegmented symbol sequence in Mboshi to a French word sequence, we discovered that our best model was the one with embedding size of 12 and two layers, but its performance was not impressive. It retrieved 7.16% of the tokens and 12.85% of the types present in our corpus. Moreover, we noticed that our best model had low BLEU score results compared to the other tested setups, which hints that this translation metric is inadequate to evaluate the system's capability of discovering words.

Investigating optimizations for improving the results, we noticed that the soft-alignments consistently ignored some of the source symbols. In fact, the alignment model has the constraint of using all the target words (French translations in our case) in the generation of the probability matrices, while no similar constraint is applied to the source units. This could explain the lack of performance in our model. We reversed our translation direction and exper-

imented feeding the French translations as source input and the Mboshi unsegmented grapheme as target output.

The results were considerably better, and the addition of alignment smoothing further improved the performance. The final model was able to retrieve 22.29% of the tokens and 27.09% of the types in the vocabulary. We also verified that the target information reduction, which means replacing the translations by their lemmas, and the Moses-like symmetrization, which means merging the soft-alignment probability matrices created by the normal and reversed models, did not help us to enhance the results.

Although our results fall far from the proposed upper-bounds, they offer alignment information that could be further explored to the task of lexicon discovery. Moreover, the retrieved soft-alignment probability matrices presented the integration of some information about the language morphological structure[3], which means the model was able to learn a little about the Mboshi structure. This is impressive considering we worked with only 5,157 examples.

We followed these experiments by a semi-supervised scenario, where we gave the system the segmentation for the 100 most frequent words in the corpus. This is justified by the fact that such information could be learned by the linguist after some days within the community whose language they wish to document.

Our results were slightly better, resulting in the retrieval of 29.49% of the vocabulary. However, although the performance improvement was not huge, we observed that this semi-supervised model created a smaller vocabulary, closer to the real one. Further analysis showed that both unsupervised and semi-supervised techniques over-segment the input, and could be helped by modifications that encourage alignments of the type one-to-many between French and Mboshi.

1.4 Contents of This Report

This report is organized as follows. In chapter 2 we present the related work, starting by presenting and comparing the two most popular approaches for unsupervised word segmentation: nonparametric bayesian models and neural models. Nonparametric bayesian models are flexible statistical approaches for word discovery and morphological analysis, dealing well with small data sets. In comparison to that, neural models need more data for achieving a good performance, but they have the advantage of being bi-lingual techniques that easily work from signal. We continue the chapter by reviewing the basic Recurrent Neural Network (RNN) definition, and we use it to present its variant used in this project: Long short-term memory (LSTM) RNN. Finally, we describe the baseline architecture for encoder-decoder NMT systems with global attention, proposed by Bahdanau et al. (2014) [7]. For our approach, the important aspect of these neural architectures lays on their internal alignment model, which is a feed-forward network inside the attention-based decoder. This alignment model produces soft-alignment probability matrices as a consequence of the translation learning procedure, and in this project we explore them for discovering words.

In chapter 3 we describe our small Mboshi-French parallel corpus — which consists in aligned audio, transcription and sentence-level translation — and we present general metrics such as vocabulary size and number of tokens. We follow this by a detailed description of our

[3]For instance, the model consistently assigned low probabilities for alignments containing French articles, showing that the model learned that in Mboshi we do not need to precede the nouns by articles.

methodology. In this description, we present our setup for the NMT systems trained in this work, and we discuss embeddings dimensionality and number of hidden layers for scenarios where we lack of data. Finally, we define our maximum *a posteriori* algorithm for transforming the soft-alignment probability matrices in hard segmentation. For evaluating our word segmentation, we use precision, recall and F-score metrics, while we use BLEU score as proxy for evaluating the learning behavior of the system. Finally, we present the two upper-bounds for our work. The first is the work of Duong et al. (2016) [16], which uses a similar segmentation process, but with a corpus more than three times the size of ours. The second upper-bound is the results presented by Godard et al. (2016) [19] using nonparametric bayesian models. They use a small portion of the same corpus used here, but with a system we judge more robust to small data sets.

In chapter 4, we present our three main experiments and results. Firstly we validated the architecture, verifying that our amount of data is enough to generate meaningful alignment information in the ideal scenario where we already have the word segmentation. Approaching the unsegmented scenario, we investigated the following optimizations: alignment smoothing, reverse architecture, target information reduction and alignment fusion. Our best model was the one created by using the reverse architecture combined with the alignment smoothing. This model has the same embedding size from the best model in the architecture validation section, and two layers in the decoder network. Lastly, we approached the semi-supervised scenario, considering that usually the documentation process does not start from zero. Our results using the segmentation for the 100 most frequent words were slightly better, but more importantly, they created a better vocabulary representation. However, further modifications are necessary in order to attenuate the over-segmentation effect caused by the neural approach.

In chapter 5, we conclude that this technique has potential, but further architecture optimization is needed in order to achieve more competitive results. We suggest the study of the BLEU score impact in the alignment model's overall quality and the replacement of LSTM by gate recurrent units (GRU) as different possible optimizations. As future work, we plan to approach speech, exploring these models capability of clustering speech fragments and their translations. We are also interested in working with the pseudo-phone machine transcriptions that we have. By approaching speech, pseudo-phone and grapheme representations, we wish to investigate their impact in the word discovery model's quality. A different perspective is to study the data impact for the soft-alignment probability matrices, investigating where lays the balance between alignment quality and data requirement.

— 2 —
Related Work

In this chapter we review the concepts we judge important for understanding this project. We start by making an overview on unsupervised word discovery models, and we follow it by a review of some important neural network concepts. Finally, we end the chapter with a description of the baseline technique for neural machine translation using attention-based encoder-decoder systems, used in our project.

2.1 Unsupervised Word Discovery Models

In the following sections, we describe two popular approaches for unsupervised word discovery: nonparametric bayesian models and neural networks systems. In this project we use a neural approach, but it is essential to understand nonparametric bayesian models in order to compare the advantages and drawbacks of these different approaches for the language documentation scenario.

2.1.1 Nonparametric Bayesian Models

Nonparametric bayesian models [21, 24] are statistical approaches that can be used for word segmentation and morphological analysis. The model is said "nonparametric" because it has no limitation in the number of parameters which can grow together with the data size.

This model major advantage for word segmentation lays on its flexibility, since it is defined by two different components [20]: a lexicon generator and an adaptor. The lexicon generator, which is task-dependent, models the kind of items that are likely to be found in the lexicon. The adaptor assigns frequencies to the generated lexical items. It does so by using power-law functions inspired on Zipf's Law [32] to adapt the words frequencies into a power-law distribution.

Recent modifications made these models able to work directly from raw speech, as exploited by Lee et al. (2015) [27] for the task of lexicon discovery. Adams et al. (2015) [2] studied the use of this technique combined with parallel information (sentence-level aligned translation). They extracted bilingual lexicons using bayesian learning with Inversion Transduction Grammar (ITG) trees, with which they achieved their best results.

Adams et al. (2016) [3] approached word discovery from phoneme lattices. They also used sentence-level aligned translations to enhance the audio transcription for an endangered language scenario. Finally, Elsner et al. (2013) [17] used nonparametric bayesian models to

study the benefits of executing the tasks of word segmentation, lexical acquisition and phonetic variability together, since they can help to constraint each other during training.

Most of the presented work execute their tasks using small quantities of data, what is in part due to the computational complexity of this technique. Despite the fact that having a model able to achieve good results with small data sets is interesting, difficulties to scale them have been pointed as a drawback. The work presented by Ondel et al. (2016) [30] suggests that replacing Gibbs Sampling (part of the lexicon generator component) by Variational Bayes Inference could solve this problem and make the approach scalable.

2.1.2 Neural Network Approaches

Although nonparametric bayesian models are successful techniques for language documentation, new research has been exploiting the contribution that neural approaches could add to this field. That is due to the fact that in the last few years we have experienced an expansion in the use of deep learning, and these neural systems were shown to be able to outperform some more specialized systems by using the correct combination between different layers types and data size.

In 2014, Bahdanau et al. (2014) [7] introduced encoder-decoder neural translation architectures using an attentional mechanism. This attentional feature not only increased the system capability to deal with different sentence lengths, but it also showed that the system jointly learns to align and translate. This alignment information, present in the form of soft-alignment probabilities matrices, is easily extracted from the trained system.

The work presented in [11] extends this architecture, presenting an end-to-end translation system working directly from signal. The possibility of using these systems to work directly from raw signal makes them worth investigating for language documentation. This is because working with endangered languages scenarios often means dealing with audio recordings aligned with translations. Furthermore, these networks also have the advantage of allowing for native incorporation of parallel information.

Studying cases without transcription, Duong et al. (2016) [16] worked both at phoneme and speech level. They compared their alignment results for phoneme level against 3 baselines (GIZA++, pialign and Model 3P), and their neural approach achieved the best results. At speech level, by creating a hierarchical structure for processing the input in the encoder network, they were able to achieve results close to the state of the art for phone-error-rate. We consider their approach for phonemes the most similar to our work.

A notable drawback of these neural approaches, compared to nonparametric bayesian models, rests in their need of very large data sets. This data is needed to train the numerous network parameters. Such amount of data is usually not available in low-resource scenarios.

Bartels et al. (2016) [8] dealt with this lack of data by training a phone recognizer with seven non-endangered languages. Then they added language specific knowledge to make their system more capable to perform lexical unit discovery in two endangered languages. Although their results were not better than other available approaches, the authors emphasize their merit of not using manual transcriptions for achieving their results.

2.2 Neural Networks and Translation

In this section we introduce recurrent neural networks and present the popular modification used in this work: long short-term memory. Following it, we present Neural Machine Translation (NMT) with focus on describing the baseline model used in this project.

2.2.1 Recurrent Neural Networks

Approaching Natural Language Processing (NLP) tasks, the need of context to interpret information is intuitive. Not only words are bound by context, but they can have dependencies that are far away, such as the subject affecting a verb after its appearance.

Unfortunately, basic neural networks (such as the multi-layer perceptron) are not enough to model this knowledge, and we often use Recurrent Neural Networks (RNN) to approach these tasks. Their main difference lays on their hidden layers, which are defined recursively.

Mathematically, a RNN hidden layer can be defined as in the equation below, where h represents a hidden layer, W represents a weight vector, x represents the input and b represents a bias vector[1]. Using this recursive definition, the network has the ability of passing information across an arbitrary number of consecutive time steps.

$$h_t = \begin{cases} \tanh(W_{xh}x_t + W_{hh}h_{t-1} + b_h) & t \geq 1 \\ 0 & otherwise. \end{cases}$$

The hidden layer is first initialized, and at each time step, it is defined by applying a nonlinear transformation on the sum of three elements. They are the product between the weights and the input at time step t (what we are seeing now), the product between the weights and the last calculated hidden layer (what we saw before) and the layer bias. In this example hyperbolic tangent (tanh) was used, but it is not the only option.

However, as explored by Bengio et al. (1994) [10], this network still have problems dealing with really long chains due to vanish/exploding gradient problems. The vanish gradient problem (similar to the exploding gradient case) affects this architecture because of the tendency of small loss values becoming exponentially small when back-propagated to the initial layers. This results in no impact in the parameters update for the initial layers and consequently, it leads to under-training.

To fix these problems, networks designed to keep long-term dependencies became popular, and in the next section we present the Long Short-Term Memory (LSTM) RNN. More detailed information about the networks presented here can be found in the survey done in [28].

Long Short-Term Memory (LSTM)

Long Short-Term Memory Recurrent Neural Networks, first introduced in [22], are networks especially designed to keep long-term dependencies. As they became popular, several variations emerged [29]. In this chapter we present the popular variation that includes a forget gate [37].

[1] This representation holds for all the other equations in this section.

The difference between an LSTM RNN and the regular RNN lays on the number of units at each time step. While a regular RNN has a single unit (such as the hyperbolic tangent defined in the last section), LSTMs are made of four additional units: cell state and input, forget and output gates.

The most important aspect of the LSTM is its cell state (or memory cell), which passes information across the time steps, being controlled by the gates. This cell state has its derivative at each time step exactly equal to one. Because of that, information kept in the cell state does not suffer from the effects of gradient vanishing, and therefore the LSTM is able to better capture long-term dependencies.

The input, forget and output gates are sigmoid neural net units, returning values between zero and one, whose extremes are interpreted as "let nothing through" and "let everything through" respectively. Using these three gates, we are able to protect and control the information inside the cell state.

The first of these gates is the forget gate, defined in equation 2.2, where σ represents the sigmoid function. The intuition behind it is straightforward: sometimes information needs to be kept only until its dependency is found, and then it stops being relevant. This gate decides if the information at each line of the cell state at time step $t-1$ should be kept. An example of this gate usefulness: if at time step t we change the subject, the forget gate can decide to forget (outputting zero) all the information kept on the cell state until the time step $t-1$.

$$f_t = \sigma(W_{xf}x_t + W_{fh}h_{t-1} + b_f) \tag{2.2}$$

The next necessary task is to decide what to put in the cell state at time step t. This is done by two units: the input gate and the nonlinear function (again, here we illustrate using hyperbolic tangent). The input gate, defined in equation 2.3, decides which values will be updated, while the nonlinear layer, defined in equation 2.4, generates a vector of candidates for updating the cell state at time step t. These two outputs are combined with the forget gate, creating the new values for the cell state, defined in equation 2.5.

$$i_t = \sigma(W_{xi}x_t + W_{ih}h_{t-1} + b_i) \tag{2.3}$$

$$u_t = \tanh(W_{xu}x_t + W_{hu}h_{t-1} + b_u) \tag{2.4}$$

$$c_t = i_t \odot u_t + f_t \odot c_{t-1} \tag{2.5}$$

Lastly, the output gate, defined in equation 2.6, will decide if what was generated at this time step should be propagated to the next hidden state. More specifically, the final hidden layer for the time step, defined in equation 2.7, will be a filtered version of what was calculated by the gates, and this filter will highlight what the network believes to be relevant for the next time step $t+1$.

$$o_t = \sigma(W_{xo}x_t + W_{oh}h_{t-1} + b_o) \tag{2.6}$$

$$h_t = o_t \odot \tanh(c_t) \tag{2.7}$$

Finally, in order to reduce the amount of parameters needed for training this RNN modification, Cho et al. (2014) [14] introduced Gated Recurrent Units (GRU) RNNs. These networks

merge the input and forget gate in the form of an update gate. They also merge the hidden state with the cell state, creating a simpler model, with less parameters to train, and one less affine transformation. We intend to explore GRU networks in future work.

2.2.2 Attentional Encoder-Decoder Architectures

Encoder-decoder architectures were initially introduced as Neural Machine Translation (NMT) techniques. Their goal is to maximize the conditional probability of a translation y given the source sentence x. Formally, this can be expressed by $argmax_y p(y|x)$.

For accomplishing that, encoder-decoder architectures use two neural networks. One for encoding the knowledge in the source language, and the second for decoding this information and for generating the most suitable translation in the target language.

Both encoder and decoder are trained jointly[2], and they are connected by a structure that passes the encoder's output to the decoder network. In the following sections, we describe the baseline encoder-decoder NMT architecture used in this project.

Architecture Schema

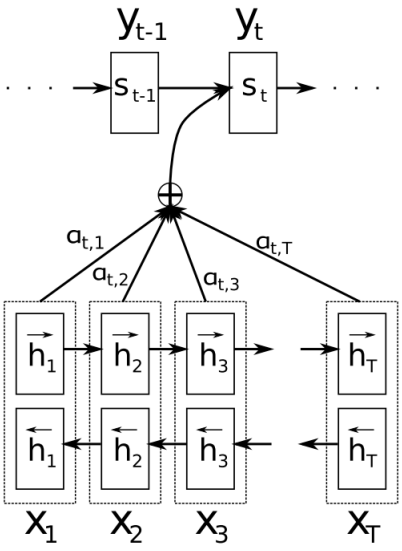


Figure 2.1 – The graphical illustration of the model generating the t-th target word y_t given the source sentence (x_1,...,x_T). Extracted from [7].

The architecture used in this project, an encoder-decoder NMT system using global attention, was introduced in Bahdanau et al. (2014) [7]. Figure 2.1, extracted from their work, illustrates the process to generate a translation y_t from the source sentence x at time step t.

Starting from the bottom of the image, the source sentence is separated in word-like units, and each one of these units has its annotation generated by concatenating the forward and backward hidden states. All these operations happen inside the encoder network.

[2]The number of encoders and decoders is not limited, and there are approaches using ensemble decoders, or even several pairs of encoder-decoder networks to generate the final translation.

In the decoder network, the generated annotations are weighted by an alignment model, generating probabilities ($\alpha_{t,1}$,...,$\alpha_{t,T}$) that represent how much each unit helps to translate x_t. The weighted sum of these annotations creates a context vector for the step t. This vector, together with the last hidden state s_{t-1} and the last generated translation y_{t-1}, are used to choose the best translation y_t.

Finally, the annotations created in the encoder network and used by the decoder network are the same for all the translations belonging to the same sentence. What changes is their weight, that is to say, their influence in the translation of the other sentence's words.

The following sections detail the attention mechanism, alignment model, encoder, and decoder networks.

Attention Mechanism

Attention was introduced to encoder-decoder NMT systems by Bahdanau et al. (2014) [7]. This feature increased these systems translation performance and created a more flexible model that generates soft-alignment information together with the translation.

Encoder-decoder architectures before attention used a fixed-size linking structure between the encoder and decoder networks. This structure contained all the source sentence information summarized by the encoder network, which was used by the decoder for generating the translation.

Working with a fixed-size structure is a waste of memory and computation time for small sentences, and at the same time it may be not enough for representing information in long sentences. Because of that, encoder-decoder architectures suffered with performance degradation when increasing the sentence length.

This problem was attenuated by allowing the decoder to have an attention mechanism. The encoder network generates annotations for all the words inside the source sentence, but this information is not summarized into a fixed-size vector anymore. Instead, at each time step t, the decoder network creates a context vector that decides which annotations can help the system to generate the translation.

These context vectors, which are different at each time step, are a weighted sum of the source sentence's annotations. The weights — which give us the importance of an annotation in the translation of the word in focus — are learned by the alignment model, part of the attention-based decoder.

This model is usually modeled as a feed-forward neural network[3] that scores how well parts of the source sentence match the translation being generated at each time step. The generated weights can be seen as alignment probabilities between parts of the source sentence and the word in focus.

Putting the context vectors' learned information together, Bahdanau et al. (2014) [7] discovered that the system generated soft-alignment probability matrices as a consequence of the implemented attentional mechanism. The authors illustrated it with four examples for their model, available in Figure 2.2.

Through these images, we can see the system was able to learn a good alignment representation between source and target languages, being even able to successfully detect inversions, such as the one that happens between "European Economic Area"' and "zone eć onomique europeénne" (Figure 2.2a). Even in the cases where the alignment is not very clear, such as in 2.2d, we believe the present information could still be useful with some post-treatment.

[3]Other popular options for this modelization are dot product and bi-linear functions [28].

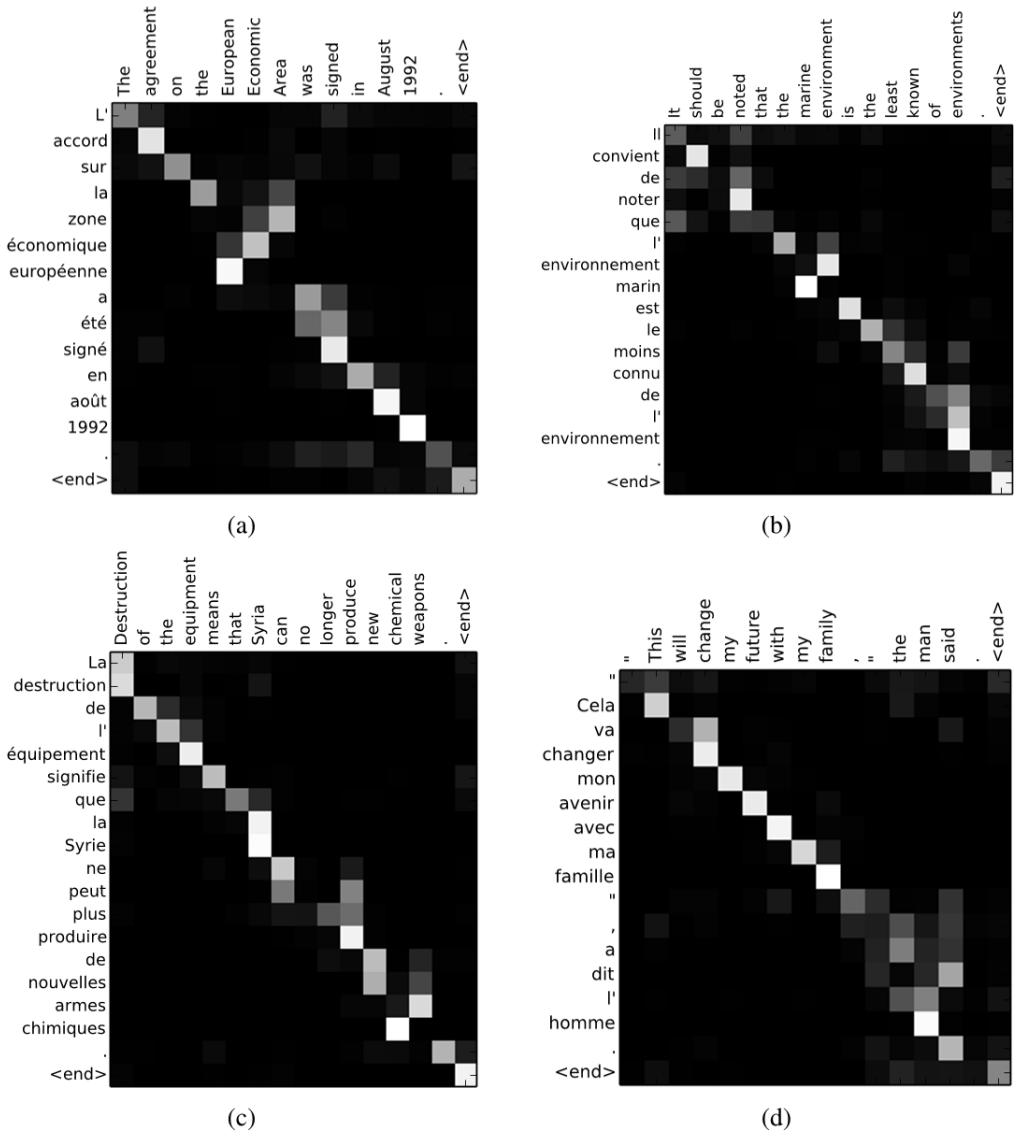


Figure 2.2 – Four soft-alignment probability matrices from the attentional model. The darker the square color, the lower the probability value. Source language is at the top of the matrices and target language is on the left. Extracted from [7].

Encoder

The Encoder is the first part of the NMT system, and it is responsible for reading the source sentence and capturing relevant information for the translation. At the end of this process, this network returns annotations for each word-unit inside the source sentence x.

When creating an annotation for a word, a unidirectional RNN can summarize information about what happens before that word, since it reads the sentence in order, from the beginning x_1 to the end x_{T_x}, T_x being the total length of the sentence x. However, sometimes it is equally relevant to have information about what happens after the word we are trying to translate.

Because of that, Bahdanau et al. (2014) [7] introduced a bidirectional RNN as encoder, which consists in a forward and a backward RNN. The forward RNN reads the sentence from the beginning to the end, calculating the forward hidden states ($\overrightarrow{h}_1$,..., $\overrightarrow{h}_{T_x}$). The back-

ward RNN reads the same sentence in the reverse order, creating the backward hidden states $(\overleftarrow{h}_1,...,\overleftarrow{h}_{T_x})$.

To obtain an annotation for each word-unit x_j, they concatenate the forward and backward hidden states, achieving a representation (Equation 2.8) that contains information about what happens before and after each word-unit. Since RNNs have the tendency to better represent recent inputs, the annotation h_j will be focused on the words around x_j.

$$h_j = [\overrightarrow{h}_j^T; \overleftarrow{h}_j]^T \tag{2.8}$$

Attention-based Decoder

For generating a translation y_i, Bahdanau et al. (2014) [7] define each conditional probability as in Equation 2.9. There, the probability of generating the translation word y_i for the source sentence x depends on the last generated translation y_{i-1}, the hidden state s at time step i and the context vector c_i.

$$p(y_i|y_1,...,y_{i-1},x) = g(y_{i-1},s_i,c_i) \tag{2.9}$$

The RNN hidden state s_i is calculated by the Equation 2.10, and both equations (2.9 and 2.10) use the same context vector c_i. As the name suggests, the context vector provides a different context for translating each word in the source sentence.

This vector is a weighted sum of the encoder annotations, and it changes for each position j in the source input. The consequence of this is a different associated context vector at each time step.

$$s_i = f(s_{i-1},y_{i-1},c_i) \tag{2.10}$$

The Equation 2.11 defines the context vector at time step i, which will be used to generate the translation y_i for the source input x_j using its annotation vector h_j. In this equation T_x is the length of the source sentence x, and α_{ij} is the weight (or impact) of the annotation h_j in the translation generated at time step i.

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j \tag{2.11}$$

These weights α_{ij} are calculated by the alignment model, which is a feed-forward neural network that scores how well the inputs around position j and the output position i match. This score is based on the decoder RNN hidden state s_{i-1} (just before emitting y_i) and the j-th annotation h_j of the input sentence.

This feed-forward neural network is trained together with the rest of the system. This is possible because it directly computes a soft-alignment (instead of representing the alignment as a latent variable), which allows the gradient of the cost function to be back-propagated through. This way, this gradient can be used for both training the alignment model and the whole translation model.

Finally, they interpret this weighted sum of all the annotations for the context vector c_i as an expected annotation over possible alignments. Let α_{ij} be the probability that the target word y_i is aligned to (or translated from) a source word x_j. Then the i-th context vector c_i is the expected annotation over all the annotations with probabilities α_{ij}.

This probability α_{ij} reflects the importance of the annotation h_j with respect to the previous hidden state s_{i-1} in deciding the next state and generating y_i. This way, we create a decoder which can decides which part of the source sentence to look at in order to generate the translation at each time step i.

— 3 —
Methodology

This project main objective is to investigate the usefulness of encoder-decoder NMT systems for language documentation. These systems train an alignment model that provides soft-alignment information that we use for performing unsupervised word discovery: the discovery of words from unsegmented source. Moreover, this information could also be helpful for the task of unsupervised lexicon discovery, the discovery of words and translations.

However, neural approaches are known for needing huge amounts of data in order to train their innumerable parameters. Because of that, we focus on investigating if these approaches are robust to low-resource scenarios, where the data is in the order of several thousand of sentences only.

In this chapter we describe the parallel corpus used in this project and our methodology. We present information about the neural translation system used and we discuss network parameters and expected results.

3.1 Mboshi-French Parallel Corpus

This corpus is a 5,157 sentences long parallel corpus in an unwritten[1] African language called Mboshi (Bantu C 25), aligned to French translations on sentence-level. Mboshi is a language spoken in the north of the Republic of the Congo, and it was one of the languages documented by the BULB (Breaking the Unwritten Language Barrier) project [4], using the LIG-AIKUMA tool [12]. Preliminary experiments for a small portion of it were reported by [19].

Three speakers recorded different sets built from 3,531 sentences from the Bouquiaux's dictionary [13] and 1,174 sentences from general use. This resulted in 5,157 sentences after preprocessing. Information about the corpus division is available at Table 3.1. For each sentence we have a grapheme transcription (the gold standard for segmentation), a translation in French, a pseudo-phone machine transcription[2] and an audio file.

To explore the impact of reducing the amount of information in the target language side, we generated a lemmatized version of the French translations. These lemmas, as well as the tokenization for the French side of the corpus, were created by using the TreeTagger tool [33]. For

[1] Even if it is unwritten, we have a non-standard grapheme form, considered to be close to the language phonology.

[2] We used the language independent pseudo-phone transcriptor presented in [18] to generate our unsegmented phone transcription.

Speaker	Number of Sentences Recorded	Percentage	Average Audios Duration (in seconds)	Total Audios Duration (in hours)
Abiayi	3,704	71.82%	3.08	3.18
Kourata	1,234	23.93%	3.29	1.13
Martial	219	4.25%	3.32	0.20
All	**5,157**	**100%**	**3.14**	**4.51**

Table 3.1 – Sentences amount, average and total duration per speaker.

the Mboshi side, we created a tokenization script and inserted space between all the characters, creating the unsegmented version that is used in our experiments.

Figure 3.1 uses a sentence to exemplify the parallel information available in our corpus. The gold standard transcription for Mboshi is used for evaluation, and three of the four representations are used in the experiments presented in the next chapter[3].

Gold Standard Segmentation:
omisíá osíísá isála lá nɯ

Unsegmented Grapheme:
o m i s í á o s í í s á i s á l a l á n ɯ

Machine Pseudo-phone Transcription:
phn_16 phn_6 phn_6 phn_11 phn_15 phn_10 phn_15 phn_15 phn_10
phn_9 phn_7 phn_9 phn_2 phn_10 phn_1 phn_2

French Translation:
as tu achevé ton travail

French Translation (lemmas):
avoir tu achever ton travail

Figure 3.1 – An example of a sentence and its available parallel information in Mboshi (bold) and French (italic). The sentences also have aligned audio files, the source of the machine pseudo-phone transcription.

Finally, Table 3.2 presents token and type information for both sides of the parallel corpus, including information for the lemmatized French version. Looking at the numbers, both languages do not seem to differ greatly in terms of expressivity and vocabulary size, with French looking a little more verbose than Mboshi, but having a smaller vocabulary. This information affects the type of soft-alignment probability matrices we can expect our alignment model to produce.

In the subsection that follows, we explain how we separate the corpus between training and development sets.

[3]The machine pseudo-phone transcriptions are not used in this work, as they lack gold segmentation for evaluation. We intend to approach them in future work.

	Mboshi	French	French (lemmas)
Avg # of Tokens / Sentence	6.95	8.52	8.52
# of Types	6,677	5,203	3,152
# of Tokens	30,712	42,547	42,547

Table 3.2 – Type and Token information for our Mboshi-French parallel corpus.

3.1.1 Development and Training Sets

Neural Machine Translation systems need two data sets for the training procedure. The training set, which is the largest of the two, contains the examples that will help the network to learn. In contrast, the development set is a small set whose objective is to tune the model's parameters and to ensure that the network is learning meaningful information through the data inside the training set. Since our data set size is already limited, we decided to use only 10% for development, leaving the remaining for training.

We do not use a test set as we have a small corpus compared[4] to what is usually necessary by a neural approach in order to create an acceptable translation system. Therefore we focus on the alignment information that is generated during training, not worrying about the translation system quality.

It is important that the sets generated for the training and development to be similar, because huge discrepancies could affect the generated translation model. In order to verify if the two generated sets for each language meet this constraint, we analyzed token and type information.

The results in Table 3.3 show that they are considerably similar with respect to the metrics evaluated here, and therefore they should be appropriated to the task in hand.

	Mboshi Dev	Mboshi Train	French Dev	French Train
Avg # of Tokens / Sentence	6.0	5.9	8.2	8.4
# of Types	1,324	6,245	1,343	4,903
# of Tokens	3,133	27,579	4,321	38,226
# of Sentences	514	4,643	514	4,643

Table 3.3 – Comparison between the sets for both languages.

3.2 Method

Our approach is illustrated in Figure 3.2. We start by training an NMT system using our Mboshi-French corpus. Firstly, we have to discover the best parameters for creating our translation model, considering we are limited by the amount of data available. This is shown as 1 in the figure, and the parameters we optimize are discussed in the Section 3.2.1.

Secondly, we insert the same set used for training into the generated translation model in order to extract alignment information. We use the same set because the system should be able to achieve a good performance aligning the information it already saw. This process is labeled

[4]Popular approaches vary from several hundreds of thousands of sentences to data sets with several millions examples.

as 2 in the figure, and while it generates a translation, here we focus on retrieving the soft-alignment probability matrices from the alignment model, jointly trained with the translation architecture.

Finally, having this soft-alignment information, we study the best way to process it in order to highlight the useful information. This post-processing results in our final model for segmentation.

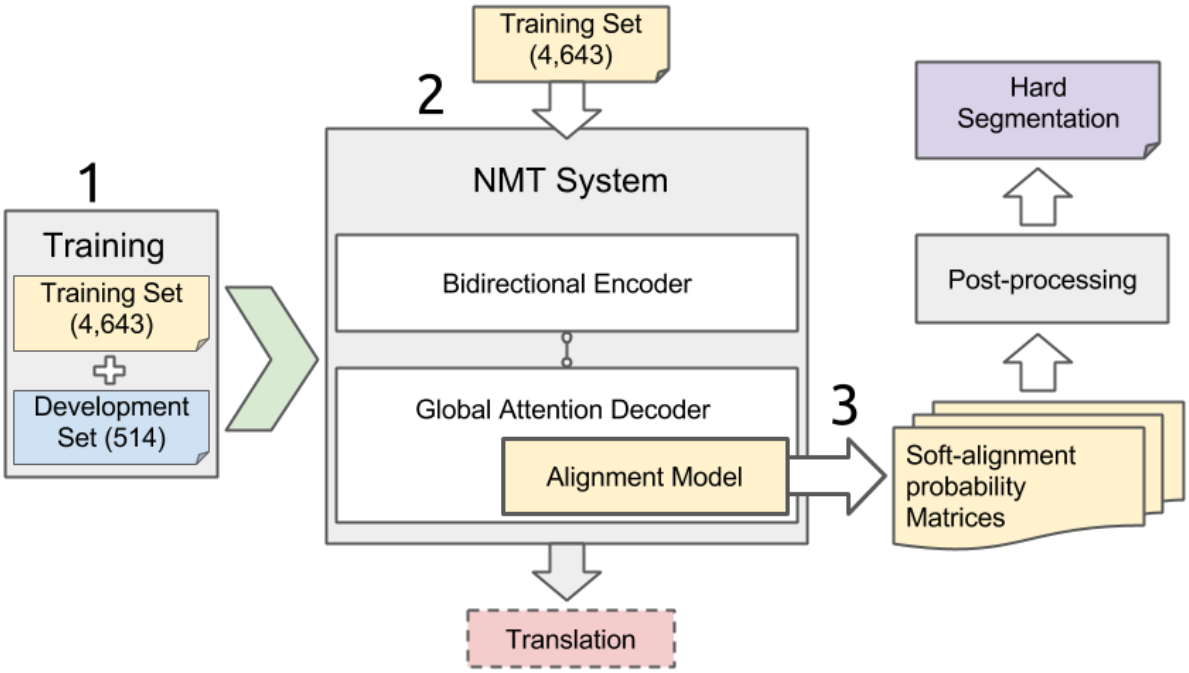


Figure 3.2 – Overview of the steps involved in this project.

In the following section we detail the encoder-decoder NMT system setup. After, we present our post-processing technique, the evaluation metrics and the expected results.

3.2.1 Neural Network Setup

In this project, we train global attention encoder-decoder neural machine translation systems following the settings described in [11]: we use a bidirectional RNN (LSTM, 256 units) encoder and an attention based decoder RNN (LSTM, also 256 units). As optimizer for the learned parameters, we use Adam [25] with initial learning rate of 0.001 and 32 as mini-batch size.

In our case, this mini-batch size represents the number of sentences observed before each parameters update. This value is half of the value used in [11] to train their data set, which was approximately four times larger than ours.

The dropout technique [37], which makes the system more robust to overfitting, with a rate of 50% was applied between encoder and decoder LSTM layers. Dropout creates an ensemble of models by "turning off" some cells and training only variables sub-sets at each step. The models are averaged to reach the network prediction.

Between the different models trained, we varied the size of embeddings and the number of layers. In the following sections we present some considerations about these parameters.

Embeddings Dimension Size

An embedding layer in a neural network is an initial layer whose goal is to learn a task-specific reduced representation for the input vocabulary. In simple terms, it projects the input into a smaller dimension, outputting vectors in a way that similar words (in terms of context) will have a similar representation (their vectors will be similar).

Reducing the dimensionality of the data simplifies training, since the following layers in the neural network will only have to learn the weights for these embeddings, which are in a lower feature space and already offer information of similar contexts. Finally, these embeddings are trained with everything else.

Intuitively, the alignment and translation tasks will be strongly influenced by how we choose to represent our input symbols, and therefore, by the size of the embeddings. On the one hand, huge embedding size values will result in one-hot vector representations, that not only are wasteful in terms of computation, but fail to capture possible relationships that emerge when we force the clustering of the vocabulary. On the other hand, reducing too much the embedding vector will result in a serious loss of information, since all the word vectors start to look alike [28].

For our experiments we use small embedding sizes, what is justified by our small vocabulary for both source (segmented and unsegmented) and target languages.

Hidden Layers Number

When layers are added to a network, they help the system to create a more abstract representation of the input sequence. In cases where we deal with speech or phonetic input sequence, adding layers can be essential for creating a system capable of clustering and interpreting the information present in this input sequence.

However, adding layers also means increasing the number of parameters learned. For instance, in one of our models, by changing the number of hidden layers from one to two for both encoder and decoder, we had additional 1,874,688 parameters to learn. This is problematic for low-resource languages, such as Mboshi, where we work with small data sets.

For our experiments we investigate a middle ground between abstraction gain and data need. We test the addition of hidden layers for both encoder and decoder networks.

3.2.2 Post-processing

To evaluate the generated alignments and compare our models, we implemented a simple algorithm to transform the generated soft-alignment probability matrices in hard segmentation. The implemented technique is a maximum *a posteriori* approach.

We explain the technique supported by the example in Figure 3.3. The algorithm starts by identifying the maximum probability (the squares in bold in the example) for all the characters. Then, when two consecutive characters have their maximum probability in the same line, such as the last four characters in the example, we consider they are a part of the same word.

When we identify that for two consecutive characters, the maximum probability of the second moves for a different line (which means the model aligned it to a different target word),

we insert a space, segmenting these characters. Applying this process in the example below, we arrive in the segmentation “é d í ngonda yá kala”.

	é	d	í	n	g	o	n	d	a	y	á	k	a	l	a
c'	**0.381**	0.151	0.045	0.026	0.072	0.060	0.038	0.026	0.035	**0.030**	**0.035**	0.021	0.016	0.013	0.015
est	0.114	0.103	0.038	0.057	0.097	0.032	0.021	0.019	0.021	0.018	0.015	0.020	0.015	0.016	0.015
une	0.178	0.157	**0.050**	0.067	0.208	0.046	0.030	0.025	0.034	0.028	0.025	0.035	0.026	0.026	0.023
ancienne	0.229	0.201	0.046	0.052	0.082	0.065	0.039	0.027	0.031	0.028	0.032	**0.050**	**0.031**	**0.031**	**0.025**
plantation	0.181	**0.204**	0.049	**0.074**	**0.469**	**0.067**	**0.040**	**0.042**	**0.037**	0.027	0.023	0.045	0.028	0.028	0.023

Figure 3.3 – An example of a soft-alignment probability matrix and the hard-segmentation (marked by the squares in bold) created using its probabilities. At the top we have the Mboshi characters, whose gold standard segmentation is **“édí ngonda yá kala”**, and in the left we have the French translation.

3.2.3 Evaluation

In this section we discuss the evaluation methods used in the project. First we discuss BLEU score, used for evaluating the generated translation model’s general quality. We follow this discussion by presenting the metrics we chose for evaluating word discovery.

BLEU Score

BLEU score [31] is a metric for evaluating translation systems without the need of human intervention. It is a language-independent metric that measures how many words in common the system’s output have with the correct translation.

In our project, since we are not interested in translation, we only use this metric as an indicator of the model’s general quality, evaluating it during training procedure for both sets (development and training). Our definitive evaluation is done after looking at that model’s segmentation results.

Metrics for Word Discovery Evaluation

For evaluating the hard segmentation generated in our experiments, we focus on the total number of tokens and types discovered by the models. The total number of tokens give us information about how many words the model was able to segment correctly, while the total number of types tell us how many different words the model was able to learn to segment.

For accomplishing so, we use classic evaluation metrics: recall, precision and F-score. These three metrics are described in the equations below.

$$recall = \frac{CorrectlySegmented}{TotalGoldStandard} \tag{3.1}$$

$$precision = \frac{CorrectlySegmented}{TotalSegmented} \tag{3.2}$$

$$F_{score} = \frac{2 \times precision \times recall}{precision + recall} \tag{3.3}$$

The first one, recall, is the more interesting for us. It give us the ratio between the number of correctly segmented words (*CorrectlySegment* in the Equation 3.1) and the total number of words we had to segment with respect to the gold standard (*TotalGoldStandard* in the same equation).

The second metric, precision, give us the ratio between the correctly segmented words and the total number of segmentations proposed by the model (*TotalSegmented* in Equation 3.2). The difference between recall and precision is that recall give us a value that represents how much information we were able to discover, while precision focus on how many candidates we generated in order to get that amount correct.

Finally, to balance the information present in recall and precision, we use F-score (also called F_1 score or F-measure), presented in the Equation 3.3. This measure considers both precision and recall, being a weighted average of these two other measures.

3.2.4 Expected Results

As commented before, only one other work that we are aware of approached language documentation using the soft-alignment information present in encoder-decoder NMT systems. In comparison to the corpus used in their work [16], an 18,600 Spanish-English parallel corpus that emulates the behavior of an endangered language, ours is more than three times smaller.

Their model, which had several modifications in the encoder, was able to achieve an F-score of 53.6 for the task of word discovery. Knowing that neural systems are sensitible to the amount of data available, we do not expect better results than theirs. However, it rests to discover how affectable by the data reduction the architecture is.

Approaching word discovery from a different angle, Godard et al. (2016) [19] were able to achieve better results by using only a small portion of the corpus used in this project (1,200 sentences). They used a nonparametric bayesian model[5] for the task of word discovery, achieving an F-score of 67.5.

We believe both works are upper-bounds for the performance[6] we will achieve in this project. Our main goal is to get as close to these upper bounds as possible. However, achieving this while working with only 5,157 sentences for training a NMT system is a considerably hard challenge.

[5] More specifically dpseg, described in [20].

[6] When we talk about performance in this project, we are referring the performance results for the task of word discovery.

— 4 —
Experiments and Results

In this project we investigate the usefulness of encoder-decoder NMT architectures as the starting point in our unsupervised word discovery task. We recover the soft-alignment probability matrices generated between the training source and target sentences, and we use these soft-alignments to segment unsegmented input symbols.

Moreover, we apply this transductive technique for a real unwritten language, and we aim to retrieve words from this language's unsegmented grapheme transcriptions — known for being close to a phoneme transcription. We explore different optimizations for our post-processing method, and we also experiment in a semi-supervised scenario, where initial expert knowledge is available.

This chapter's content is organized as follows. Firstly we investigated if our small data set was enough to retrieve alignment information in the ideal scenario where we already have the golden word segmentations. These results are reported in the Section 4.1.

We followed this validation by experiments covering the real scenario of interest: we use unsegmented graphemes as source input and study the system's performance for the task of word segmentation. These experiments are presented in Section 4.2.

Finally, we present an extension of our approach for the semi-supervised scenario, using the most frequent words from the language we wish to document as help to discover the rest of the vocabulary. These experiments are detailed in Section 4.3.

4.1 Architecture Validation

NMT systems require considerable amounts of data for generating their translation. However, in our scenario, we are not interested in creating a generalizable translation system, instead we only want to retrieve the alignment information learned during the training procedure. Therefore, we examine if our small data set, described at Chapter 3, is enough for such a task.

To confirm this, we investigated if we were able to align the gold standard segmentation for Mboshi sentences with their French translations. This scenario is the ideal case were we already have all the word boundaries available.

If we are not able to align source and target language in this case, we know it would be impossible in the real scenario, where we do not have boundaries between the input symbols. In the following sections we describe the evaluation method, setup and results for the architecture validation experiment.

4.1.1 Qualitative Evaluation

For our architecture validation, we had to evaluate our results without the help of word-level alignment between the Mboshi sentences and their French translations. For accomplishing this task, we used a small Mboshi-French dictionary [9].

We created a very small test set from the training set, made of 20 randomly selected sentences — of which at least one of the words was present in the dictionary we had access — which resulted in a set with 68 known Mboshi words (28 different entries). We had to manually read each sentence and search for context and dictionary entries to check if these known Mboshi words were being correctly translated[1]. Because this manual verification costs considerable time, we did not expand the test set any further.

For evaluating this set, we generated the heat maps from soft-alignment probability matrices and manually counted the number of words correctly aligned. These results were used, together with the BLEU score, to rank our models with respect to their overall quality.

A maximum *a posteriori* strategy was adopted for evaluation, in which a word was considered well aligned if the probability of it being translated to the word that we had in our dictionary was the highest probability over all the other options. In other words, the source word was more likely to be translated for this target word, and this target word was more likely a translation of that source word.

When more than one pair for a given source word had the same probability, the alignment was considered incorrect. Multiword expressions were considered correct only if all their components had the same probability (and this probability was the highest).

We illustrate the process on Figure 4.1, where we have two heat maps generated using the soft alignment model. For the sentences in this example, we are considering the Mboshi word "poo", which means "village" in French.

In Figure 4.1b the word was correctly aligned, since we can see that the square representing the pair (poo, village) is the stronger for both words. Meanwhile, on Figure 4.1a, this word was instead aligned to "dans" (which should had been aligned to "sá").

It is worth mentioning that we are not penalizing the model when the alignment includes the article (for instance, aligning "poo" to "le village"). That is because the Mboshi language have an agglutinative morphology [19], usually having the article information as the word's affix.

[1]For instance, in some sentences the Mboshi word "nyama", whose dictionary entry is "animal", was aligned with the word "gibier" ("prey" in English). However, when that happened, the context was the same from the cases where it was aligned to ''animal", and this context was hunting. Therefore, we considered both alignments correct.

(a) Bad alignment example for the word “poo”.

(b) Good alignment example for the word “poo”.

Figure 4.1 – Soft-alignment probability heat maps for two sentences. On top we have the source words in Mboshi, and on the left their French translations. The squares represent the alignment probability, and the higher the probability, the darker its square is.

4.1.2 Setup

As explained in Chapter 3, for our experiments we changed the size of the embeddings and the number of hidden layers in the encoder and decoder networks. For the embedding size we started by using 256 as an upper bound, since in [11] they use this value for a larger corpus, and we explored smaller representations, more adapted to our vocabulary size. We tested the values 16, 32, 64 and 128. For the hidden layers we tested models using one and two layers for both encoder and decoder RNNs.

Finally, after finding our best model by varying these settings, we ran an overfitting experiment, where we used the same set for training and development. The intuition behind this experiment lays on our transductive learning scenario: we are not interested in the generality of our system, or worried about overfitting[2]. Instead, we only wish to achieve the best possible soft alignments on the training data.

At total, 11 different setups[3] were tested, listed in Table 4.1. The final generated translation model, from which we extract the soft-alignments, is not necessarily the model generated at the last step, but instead the model with the best BLEU score over the development set (excluding the overfitting version, where we use the training set for both tasks).

	T1	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**	**T10**	**Overfitting**
Embeddings	256	256	128	128	64	64	32	32	16	16	32
Layers	1	2	1	2	1	2	1	2	1	2	1

Table 4.1 – List of tested setups and their parameters.

[2]For this version, we turned off dropout.

[3]For executing the setups, we used the preprocessing scripts and NMT models implemented in the baseline branch of this repository: https://github.com/eske/seq2seq. The neural models were implemented using Python3, and they use the TensorFlow library [1] with GPU support.

Since all of our setups have different numbers of parameters, their training time is also different. In order to ensure that our models were not under-trained, we monitored the loss function behavior, as well as the BLEU score for the development set, and ended the training only after verifying that both had reached their function's plateau. In Figure 4.2 we have an example (from model T3) for the loss function and BLEU score behavior through training.

Loss function evolution for development and training sets.

BLEU score behavior.

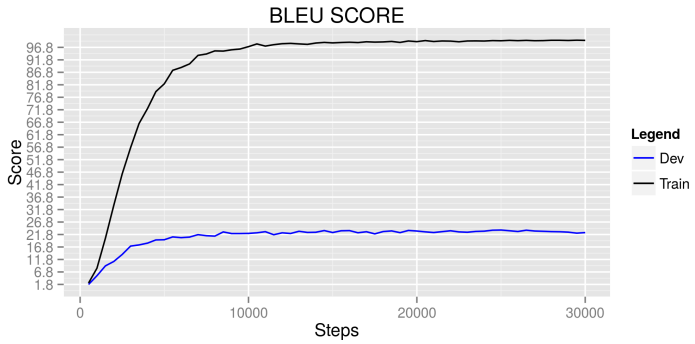


Figure 4.2 – Loss function and BLEU score behavior through training.

4.1.3 Results

Table 4.2 shows our results for word alignment for the different models (excluding overfitting, which will be discussed in the end of this section), evaluated using our small test set. Considering these numbers, our best model was the one with embedding size of 32 and one layer, which corresponds to the setup T7. This model correctly aligned 45 from the 68 Mboshi words in the test set, and the overall quality of the alignments was good, with strong probabilities in the diagonal[4].

As expected, looking at the results we conclude that smaller embedding sizes fit our task better. That is because they are more efficient dealing with our vocabulary, avoiding one-hot vector representation, that happen when the embeddings dimensionality is equal or larger than

[4] As discussed in the Section 3.1, the languages do not vary much with respect to number of tokens per sentence and therefore, we can expect the majority of the alignments to be one-to-one alignments. Moreover, as mentioned in [19] and can be verified in [5], Mboshi do not differs a lot from the French structure, so we can expect most of the alignments to fall near the matrix diagonal.

Embeddings	**256**	**128**	**64**	**32**	**16**
one layer	29.41	8.82	61.76	**66.17**	57.35
two layers	16.17	20.58	38.23	20.53	44.11

Table 4.2 – Percentage of words correctly aligned (from our mini-set of 68 tokens)

the vocabulary. Also, we have overall better results by using less layers, which we believe is due to the smaller number of parameters to train.

In Figure 4.3 we have the BLEU score values for all the tested models. It is interesting to notice that our best model (T7) do not have the best BLEU score neither for training or for development. Other very good model, T9 has terrible results using this translation metric.

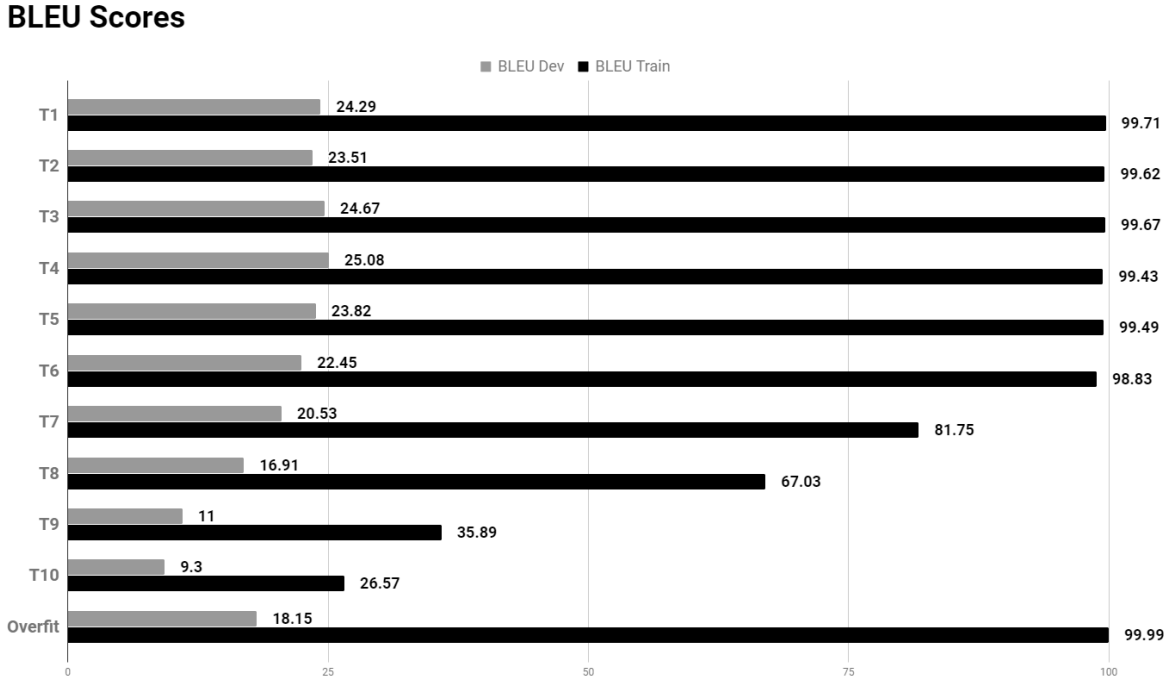


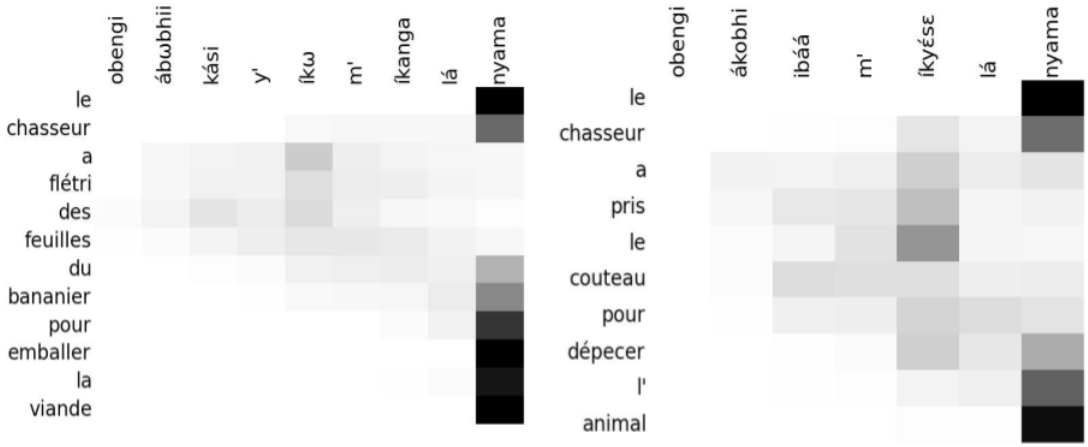
Figure 4.3 – The resulting BLEU scores for all the tested setups.

At this point, and using a very limited test set, it is not possible to conclude if the BLEU score is related to the success in our task or not. However, we point out to the importance of considering more than BLEU score, and having the right combination of embeddings dimensionality and hidden layers, since they affect greatly our generated soft-alignment matrices.

In Figure 4.4 we have two heat map examples for T2 and T7. Supported by the BLEU score results, we could expect T2 to have better results aligning the sentences, since it has one of the better BLEU scores for the training set. However, we see that although this model is better translating the training set, it is not good at aligning it.

Finally, after concluding that the setup T7 was the best between our models, we ran the overfitting experiment. The BLEU scores values for this setup are available at Figure 4.3. This model was able to correctly align only 5 words (7.35%) in the test set, and the overall quality of the alignments was not good. This is somewhat expected, since we had already verified that a high BLEU score for the training set does not necessarily mean a good alignment quality.

T2 heat maps. The examples illustrate the poor quality of the learned alignment model.



T7 heat maps. The examples illustrate the good quality of the learned alignment model.

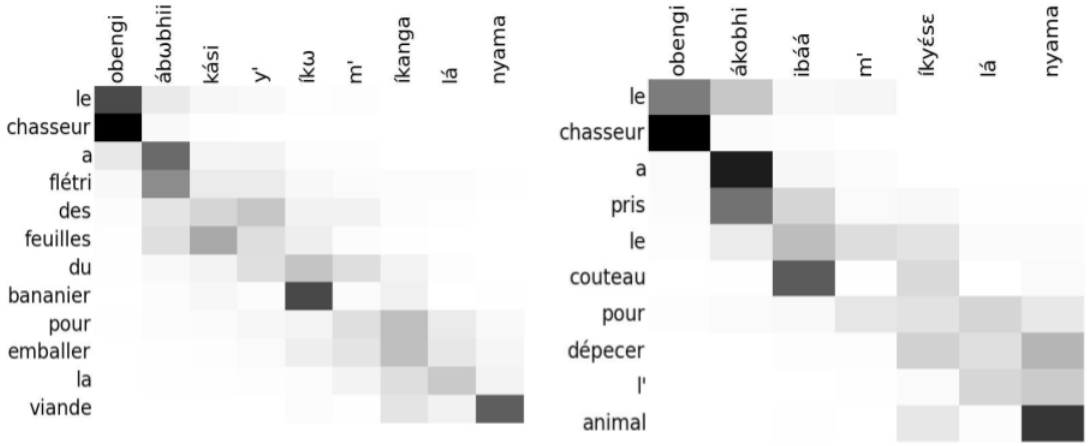


Figure 4.4 – Example of two sentences' soft-alignment probability heat maps for T2 and T7 setups.

Moreover, we believe the models need the development set in order to verify that the alignments they are learning are coherent.

To conclude, we were able to verify that our data set is enough to generate meaningful alignment information between the gold segmented Mboshi source and its French translation. This can be proved by the overall good quality achieved by some of our models, notably the

T7 setup. We were able to discover that we cannot depend on the BLEU score alone in order to evaluate our model's quality, and we verified that we need a development set in order to tune the alignment model.

4.2 Unsupervised Word Discovery Experiments

Following our architecture validation, we approach the real scenario: the extraction of alignment information from unsegmented source sentences aligned with their translations. We have as source an unsegmented version of the Mboshi sentences used in the last section, which contain 44 different symbols. For target, we keep the same French sentences used in the last experiment.

It is important to remember that this grapheme form used here follows the Mboshi phonology, being close to a phonetic transcription. In the next sections we describe our setup, segmentation algorithm, results and optimizations.

4.2.1 Setup

For these experiments, we worked with character symbols, and our intuition was that more hidden layers could help the system to extract more information from the source sentence, creating better representations in the encoder side. However, we still have a limited amount of data.

To investigate if keeping only one hidden layer is still the best for our system's performance, we tested models with one, two and three layers in the encoder network[5]. These three different setups were tested with the embedding size of 12.

For embeddings, we worked with smaller values compared to the ones used when we approached the gold segmentation scenario. This choice is because we move from a vocabulary size of 6,245 words, to a vocabulary of 44 grapheme units.

We tested the embeddings values of 6, 12, 32 and 64, and first we ran the hidden layers experiments, and then we used our best number of layers in all our setups for investigating the best embedding size.

4.2.2 Preliminary Results (Mboshi-French)

For evaluating our models, we first generated the soft-alignment probability matrices and transformed them in hard segmentation, as described in Section 3.2.2. Then we compared this segmentation with the Mboshi gold standard by extracting recall, precision and F-score metrics (presented in Chapter 3) for tokens and types.

While all the metrics are interesting to analyze[6], the one that weighted more in our decision for choosing the parameters was the recall. That is because it evaluates how many words (tokens) and different words (types) we were able to retrieve using this unsupervised word segmentation technique.

In Table 4.3, we present the results for our hidden layers experiment. We can see that adding one hidden layer increased the model capability of discovering different words, what

[5]For the experiments in the Section 4.1 we used layers for both encoder and decoder networks.

[6]More detailed interpretation of these metrics was presented in Section 3.2.3.

	TOKENS			TYPES		
	Recall	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
One layer	**7.49**	3.23	4.52	8.91	**6.75**	7.68
Two layers	7.16	**4.50**	**5.53**	**12.85**	6.41	**8.55**
Three layers	5.47	3.35	4.16	9.63	4.87	6.47

Table 4.3 – Results for the models with one, two and three layers and embedding size of 12. In bold are the best values for each metric.

can be observed by the increase in the types recall for the model with two layers. However, going further and adding a third layer made the system worse in all the metrics evaluated.

Comparing the models with one and two layers, the one with two was slightly worse in the number of words retrieved (tokens recall). However, this model was also more precise, what made it achieve a better F-score for tokens. Since it was also better retrieving types (types recall), a harder task than retrieving tokens, we classified this model as our best, and for the rest of the experiments in this section, we used two layers in all our setups.

Following these experiments, we investigated the values for the embeddings, using models with two hidden layers. We tested four different setups (6, 12, 32 and 64) and the results are presented in Table 4.4.

Training these models, we observed worse BLEU scores for training and development sets as we reduced the embedding size. Our best model (12) had a BLEU score of 3.66% for development and 3.39% for training. In comparison to that, the model with largest embedding size (64) achieved BLEU score of 18.99% for development and 95.29% for training. Based in these results, we concluded that we cannot use the BLEU metric to evaluate the quality of our alignment models.

Looking at the results, we observe that smaller values are better discovering types, which we believe is due to the lower dimensionality of the generated embeddings. As discussed in Section 3.2.1, these smaller vectors potentially have more information, obtained by clustering the input.

	TOKENS			TYPES		
Embeddings	**Recall**	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
6	6.22	4.47	5.2	12.81	6.04	8.24
12	7.16	**4.5**	**5.53**	**12.85**	6.41	**8.55**
32	**8.37**	3.25	4.69	7.18	**6.6**	6.88
64	6.53	3.17	4.27	8.26	5.55	6.64

Table 4.4 – Results for models with different values for embedding. In bold are the best values for each metric.

In general, the word discovery results are far from what we were aiming, as the upper-bounds presented in Section 3.2.4 are several times higher in terms of F-score. However, these results are almost "out-of-the-box" results for the neural system, since we did not apply any sophisticated post-processing to the generated soft-alignment probability matrices. In the next sections we describe implemented optimizations and their impact in our results for the task of unsupervised word discovery. We summarize our best model in Section 4.2.6.

4.2.3 Alignment Smoothing

In this project, where we use an NMT system to retrieve alignment information between unsegmented source and target words, we know our source input will always be larger than the target input. To encourage alignments of the type many-to-one, Duong et al. (2016) [16] proposed two alignment smoothing techniques.

The first one is a post-processing filter, that replaces each probability α_{ij} in the soft-alignment probability matrices by $\frac{1}{3}(\alpha_{i,j-1} + \alpha_{i,j} + \alpha_{i,j+1})$, i and j being respectively the target words and source symbols indexes. Since this modification is done after training, this technique does not affect the NMT system.

The second modification is done during training: Duong et al. (2016) flatted the softmax function[7] by adding a temperature factor. The softmax with temperature is shown in Equation 4.1 (defined for T $\geq$ 1).

$$\alpha_{ij} = \frac{exp(e_{ij}/T)}{\sum_k exp(e_{ik}/T)} \tag{4.1}$$

When this temperature factor is equal to one, we have the standard softmax function. Their model with the alignment smoothing[8] increased their segmentations results by 4.6% (F-score for tokens).

In our experiment, we compared the best model from the last section, which we refer to as "base model", with its version with post-processing filter. We also trained a model using the same setup but adding the temperature factor, and then applied the post-processing filter. Table 4.5 summarizes our results.

	TOKENS			**TYPES**		
	Recall	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
Base Model	**7.16**	4.50	5.53	12.85	6.41	8.55
+ post-processing	6.75	5.53	6.08	14.74	6.45	8.98
+model temperature	7.00	4.59	5.55	12.69	6.58	8.66
all modifications	6.82	**5.85**	**6.30**	**15.00**	**6.76**	**9.32**

Table 4.5 – Comparison between the base model and the proposed modifications. Best values for each metric are presented in bold.

We observe that this smoothing procedure resulted in a small deterioration in the absolute number of tokens found[9], however it improved all the other metrics, in special types recall. We therefore concluded that this smoothing technique is helpful to us, and we use it for the remaining of our experiments.

4.2.4 Reverse Architecture (French-Mboshi)

Evaluating the matrices generated by the best model present in the Section 4.2.2, we observed that the system was consistently ignoring part of the source symbols when generating the trans-

[7]Softmax is responsible for normalizing the scores generated in the hidden layers, and it does so by transforming them in probability values between zero and one.

[8]For their experiments they used temperature factor equal to 10.

[9]The model with smoothing had a reduction of 96 correct tokens and an increase of 134 correct types.

lation. This resulted in a lot of characters having only extremely low probabilities, harming the segmentation quality.

In NMT systems, the soft-alignments are created by forcing the probabilities for each target word j to sum to one (that is $\sum_i \alpha_{ij} = 1$), what ensures all the target words are used. However, there is no similar constraint for the source symbols, as discussed in [16].

Considering that we are interested in segmentation, our system must use all these source units from the unsegmented source input when processing a sentence. Duong et al. (2016) [16] solved this by adding a regularization penalty that encouraged the columns' probabilities (in other words the characters' probabilities) to sum to one. For our work, since we are not interested in translation, we simply reversed the system input, creating a French-Mboshi words-to-characters translation system.

For the embeddings we used the same value used in the architecture validation experiments (Section 4.1), since the vocabulary size is almost the same[10]. We also added two layers in the decoder side, since in Section 4.2.2, we used two layers for the encoder. Results are presented in Table 4.6.

	TOKENS			**TYPES**		
	Recall	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
Base Model (Mb-Fr) with alignment smoothing	6.82	5.85	6.30	15.00	6.76	9.32
Reverse Model (Fr-Mb) without alignment smoothing	20.04	10.02	13.36	18.62	14.80	16.49
Reverse Model (Fr-Mb) with alignment smoothing	**22.29**	**17.15**	**19.39**	**27.09**	**15.01**	**19.32**

Table 4.6 – Comparison between our base model and the proposed reverse model with and without alignment smoothing. Best values for each metric are presented in bold.

By analyzing our results, we concluded that by reversing the source and target languages, we created a better model, which is justified by the considerable performance increase we had for tokens and types. Adding the alignment smoothing described in Section 4.2.3 further improved the system's performance for the task of unsupervised word discovery.

4.2.5 Unsuccessful Attempts

In the following sections we present two attempted optimizations that did not result in performance increase for our word discovery models. The first one, alignment fusion, was inspired by the symmetrization done by statistical machine translation systems. The second one, French information reduction, was performed in [19], and our goal was to investigate if the performance degradation would hold using a neural approach.

Alignment Fusion

Statistical machine translation systems, such as the baseline Moses [26], extract alignments in both directions (source-to-target and target-to-source) and then fuse them, creating the final translation model. This alignment fusion is called symmetrization.

[10] As shown in Section 3.1, Mboshi vocabulary has 6,677 words, while the French one has 5,203.

In order to investigate if a Moses-like symmetrization would help our results, we merged the soft-alignments matrices generated by our best model (Mboshi-French) in the Section 4.2.2 with the matrices from our reverse model (French-Mboshi) described in Section 4.2.4. To merge the probabilities, we replaced each probability $\alpha_{i,j}$ by $\frac{1}{2}(\alpha_{i,j} + \beta_{j,i})$, where $\beta_{j,i}$ is the probability for the same alignment i-j in the reverse architecture.

This symetrization did not lead to any improvement, and the results are presented in the appendix (Table 6.1). We tested three possible setups for alignment fusion using the matrices generated by the base and reverse models. We tested the fusion of the raw soft-alignment probability matrices (without post-processing), the fusion of already smoothed matrices, and this last fusion followed by re-smoothing process.

We believe the results were worse because, the base model do not have the probability constraint discussed in Section 4.2.4. Because of that, by merging the base model with the reverse one, we can accidentally destroy good alignments that were not found in the base model. This is because these alignments can have very low probabilities in the base model (for instance, in the case where the model aligns a part of the characters and ignores the rest), which will weight down the mean.

French Information Reduction

The final experiment performed for the task of unsupervised word discovery, was French (or target) information reduction. This information reduction is the replacement of the target sentences' words by their lemmas.

The intuition is that, by simplifying the target information, the model could more easily learn relationships between the two languages. For the task of unsupervised word discovery, this technique was recently investigated by Godard et al. (2016) [19].

In their work, the authors performed unsupervised word discovery experiments using two corpora: a large English-French parallel corpus and a small portion of the Mboshi-French corpus[11]. Their results showed that lemmatizing the target sentences helped the larger corpus, but worsened the achieved results for the Mboshi-French corpus.

Here, we investigated if this performance drop holds in the case where we have a larger Mboshi-French corpus and a neural approach. Our results, which demonstrated that this lemmatization results in performance drop of approximately 2% for tokens and types, are presented in the appendix (Table 6.2).

4.2.6 Final Results for Unsupervised Word Discovery

In the last sections, we presented two optimizations that successfully increased our model's performance (alignment smoothing and reverse architecture) and two optimizations that did not help (alignment fusion and target information reduction). Our best model for the task of unsupervised word segmentation was the last one presented in Section 4.2.4, which combined the reverse architecture with the alignment smoothing presented in Section 4.2.3.

In Section 3.2.4 we discussed two upper-bounds for our work. The first was the work presented in Godard et al. (2016) [19], which used nonparametric bayesian networks with a small portion of our corpus. The second was the work of Duong et al. (2016) [16], which used the same neural approach than us.

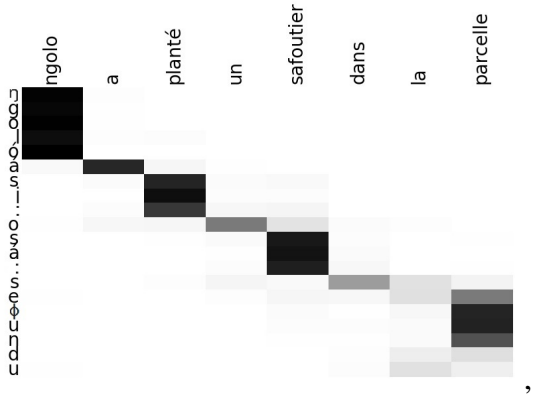
[11] They worked with 1,200 sentences from the 5,157 sentences corpus used here.

The nonparametric bayesian approach, known for being robust to scenarios with small amounts of data, achieved a recall of 44.8% for tokens and 51.57% for types, while Duong et al. (2016) [16], using a neural approach and an 18,600 sentences corpus, achieved a F-score of 53.6% for tokens[12].

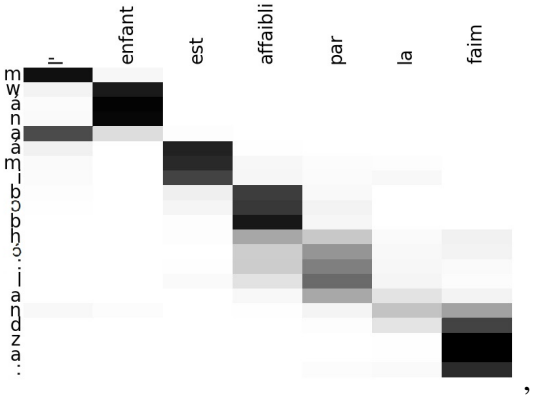
Compared to nonparametric bayesian approaches, our lack of performance is somewhat expected, since neural systems are highly affected by the data set size, lacking the same robustness that nonparametric bayesian approaches have. We believe this problem holds when comparing our results against the ones achieved by Duong et al. (2016) [16].

However, it is important to notice that our proposed model does not only segment the unsegmented input (as nonparametric bayesian approaches do) but it also aligns it to a foreign word (input is segmented, and aligned to a potential translation as a by-product). We also believe that attentional neural models are very well suited for processing signals (which, again, is not the case of nonparametric bayesian models).

In Figure 4.5 we present two examples of heat maps generated from the soft-alignment probability matrices produced by our best model. We believe these two examples are a good representation of the overall quality of our soft-alignments.



(a) In this example, we observe some degree of linguistic incorporation, as the model learned to assign low weights to the article "la", usually absent in Mboshi.



(b) In this example, we observe a good alignment that resulted in a bad segmentation. The Mboshi word "mwána", which is aligned to "l'enfant", was not correctly segmented but, it was correctly aligned to its translation.

Figure 4.5 – Two examples of soft-alignments produced by our best model. The darker the square color is, the higher is the alignment probability for that pair.

Investigating the heat maps, we observed that the model was able to incorporate some linguistic characteristics, such as the absence of articles before the nouns in Mboshi. This is illustrated by the low probabilities (brighter colors) in the alignments for the French article "la" in Figure 4.5a, which means the system is not giving importance to that information when aligning the input.

For other cases, we noticed an incorrect segmentation, but a correct translation alignment. This is the case of Figure 4.5b, where the word "mwána" was incorrectly segmented as "m wán a", but it was correctly aligned to "l'enfant". This illustrates that our model is retrieving translation information for the discovered words, and this information could be exploited to help the segmentation process.

[12]Types results, as well as precision and recall, were not reported in their work.

Finally, the incorporated information in the soft-alignment probabilities matrices is even more interesting when we consider that we only worked with 5,157 sentences. We believe that by exploiting this information, we could improve our unsupervised word discovery task in the future.

4.3 Semi-supervised Word Discovery Experiments

In the last experiments, we investigated a totally unsupervised scenario for word segmentation. This is justified by our project aiming at endangered language documentation.

However, a language documentation task is hardly totally unsupervised, since linguists usually immerse themselves in the community when documenting its language. Because of that, in this section we explore a semi-supervised approach for word segmentation, using the same architecture from the last experiments.

For these semi-supervised experiments, we select the most 100 frequent words inside the gold standard for Mboshi segmentation. We consider this amount reasonable for representing the knowledge a linguist could acquire after spending some days in a community.

Our intuition is that by providing the segmentation for these words, we can help to improve the system's performance for the rest of the vocabulary. We experiment using our best unsupervised word segmentation model, and Table 4.7 presents our results.

# of known types	Recall	Precision	F-score	# correct types	# generated types
0	27.09	15.01	19.32	1,692	11,266
100	29.49	24.64	26.85	1,842	7,473

Table 4.7 – Results for the semi-supervised word discovery task.

To consider existing expert knowledge (100 words in our case), we simply add known tokens on the Mboshi side and keep the remaining symbols unsegmented. This created a mixed representation, where the Mboshi input has at the same time unsegmented graphemes and segmented words. Finally, since languages follow exponential distributions [32] and we are giving to the model the most frequent words in the corpus, we do not focus on the tokens results (presented in the appendix, Table 6.3), since they could bias us to evaluate the model as better than it really is.

Our types results in Table 4.7 indicate that the known words helped the model to discover a few more types [13]. However, the impact was not as big as expected, and we can observe that the types recall varies only 2.4% between the models. Investigating further, we discovered that our unsupervised model was able to discover 97 from these 100 words, which could justify the small performance difference between the models.

Finally, it is interesting to notice that, while the performance increase was not huge, the semi-supervised system reduced considerably the number of types generated, from 11,266 to 7,473. This suggests that the information helped the model to create a better vocabulary representation and consequently, a better precision score. To visualize this reduction in the vocabulary size, in Figure 4.6 we have the number of generated types by their lengths.

[13] Excluding the 100 types already known, the semi-supervised model found 50 types that the unsupervised system was unable to.

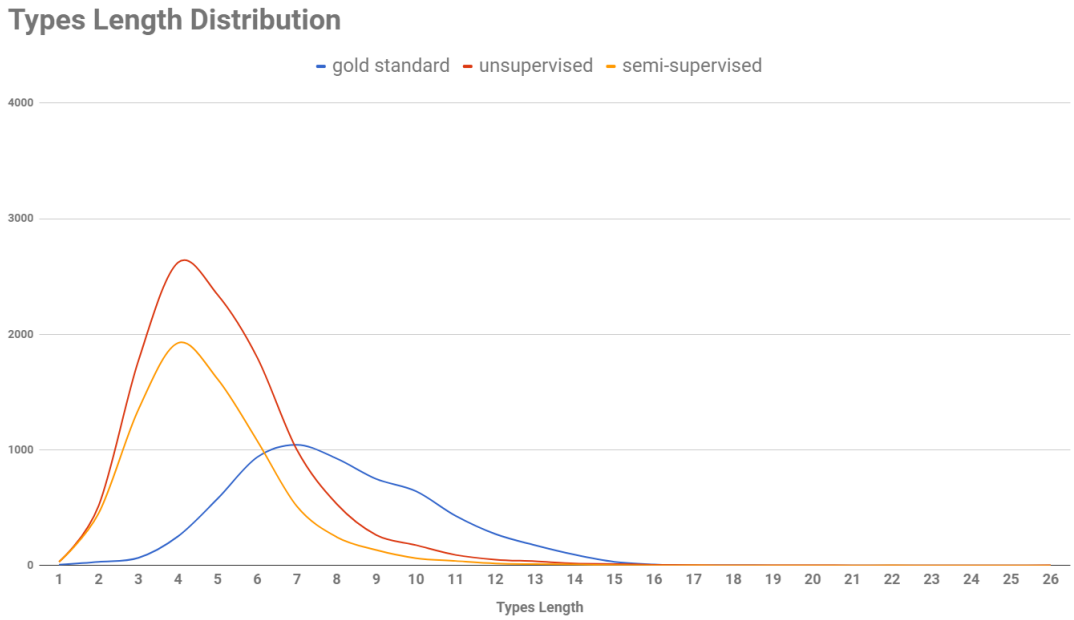


Figure 4.6 – Types length distribution for our two setups and the gold standard.

By analyzing the graphic, we can see that our neural approaches are over-segmenting the input, creating vocabularies with smaller words than the ones in the gold standard. We also can observe that while the semi-supervised approach created a smaller vocabulary, it still over-segmented the input, following the same behavior than the unsupervised approach. This helps us to understand why we did not achieved a better performance retrieving types with this technique.

Neural approaches, where all the input symbols start separated from each other, have the tendency to over-segment the output. Moreover, for accomplishing segmentation, practically all the alignments between Mboshi and French should be many-to-one alignments. We tried to encourage the system to create these alignments by adding the temperature factor discussed in 4.2.3, but we believe that further modifications are needed in order to attenuate this over-segmentation effect, that prevent us from discovering longer types.

— 5 —
Conclusion

In this work, we investigated a transductive approach that used an encoder-decoder NMT system as the starting point of an unsupervised word discovery technique. We used a real endangered African language called Mboshi in order to verify if the learned attentional models are robust to low-resource scenarios (limited amount of data available).

We discovered that for the ideal scenario where we have the Mboshi word segmentations, we are able to generate good alignments between source and target languages. This scenario could be potentially useful to learn a bilingual lexicon in an unknown language, for language documentation.

However, the main task we were interested in was unsupervised word discovery using an unsegmented Mboshi transcription (grapheme symbols). We verified that the way neural architectures are defined, not constraining the use of all the source units for generating the soft-alignment probabilities, makes the segmentation task difficult to accomplish.

However, by reversing the system, creating a French-Mboshi words-to-characters translation system, we were able to double our tokens F-score, and to increase the types F-score by 7.17%. Applying the alignment smoothing techniques described in [16], we were able to further enhance our model's performance, retrieving approximately 27% of the Mboshi total vocabulary.

We consider these results encouraging, since we worked with a model known for needing considerable amounts of data, and we reached these results by using only 5,157 examples. Our approach is less performing than a nonparametric bayesian model baseline, but our model also learns translation information, and it is easily extendable to work from speech input.

We considered the tuning of these models to be a great challenge in our project. This happens mostly because, as presented through our experiments, we discovered that BLEU score was a poorly reliable proxy to evaluate our trained systems capability of segmenting the input.

Moreover, we consider that the neural systems optimization process is tough and time consuming. While we only focused on two parameters (embeddings and hidden layers), there were several other options that could have been investigated (for instance, LSTMs could be replaced by GRU cells).

Finally, we also discovered that for our data set, the information reduction in the target language (French) does not help our task. The same result was reported by [19] with a small portion of the corpus, and they discussed that this may be due to the nature of the manually generated translations for Mboshi. That is to say, it is a characteristic of the corpus, and not necessarily a generalizable result. Performing Moses-like symmetrization (alignment fusion) between the Mboshi-French and French-Mboshi systems also lead to negative results.

Investigating a scenario where we have some prior expert knowledge, we selected the 100 most frequent words in the Mboshi corpus and gave their correct segmentation to the system. The results for this semi-supervised approach were slightly better, retrieving almost 30% of the Mboshi vocabulary. However, although the results were not impressive, we verified that the learned vocabulary was more concise and closer to the real one, compared to the one learned unsupervisedly. This indicates that this information is helping the system to learn a better vocabulary representation. Nevertheless, we believe the model still needs modification to fit the segmentation task, where the alignments must be one-to-many alignments between target and source languages in order to retrieve longer tokens.

Lastly, while our results are considerably lower than our defined upper-bounds, we still verified that it is possible to retrieve information from attentional encoder-decoder architectures in these restricted scenarios, although these results are not optimal. Also, the retrieved soft-alignment probability matrices were able to incorporate some of the Mboshi morphological information, which is impressive considering we only used 5,157 sentences for training the models.

Finally, we consider our model more appropriate for working from signal, and we believe this approach has potential for documenting unwritten languages, since it also incorporates translation information. In the following section we discuss perspectives for future work.

5.1 Future Work

In this work we presented a first attempt of using encoder-decoder attentional models to segment and align a real scenario of endangered language. We plan to continue to investigate attentional architectures, to fully understand their contribution to language documentation scenarios.

As mentioned in this work, we identified a series of optimizations that could help the architecture to better perform the word discovery task. We are interested in investigating a better metric to evaluate the models segmentation and alignment ability, since we identified a problem using the BLEU score during training. We also would like to enhance our model, adding the structural bias components described in Duong et al. (2016) [16] to our global attention mechanism. We believe these modifications could enhance our results and produce a model more robust against over-segmentation.

Our goal for future work is to apply these neural architectures, robust to audio input and able to incorporate parallel information into their models, for clustering audios fragments and to align them to translations. For approaching signal, we plan to use the convolutional attention presented in [15] and applied[1] in the end-to-end speech-to-text NMT system presented in [11].

We also intend to study the machine pseudo-phone transcription presented in Chapter 3, but not used in this work. By approaching grapheme, pseudo-phone transcription and speech, we will investigate how the different representation levels impact the quality of the retrieved information. By doing so, we will be able to understand how these different representation levels can be learned from small data sets.

From a different perspective, we would also want to investigate the data impact in the soft-alignment probability matrices. Our intuition is that, by feeding these neural systems more

[1] Implementation available at: https://github.com/eske/seq2seq, branch speech.

data, this soft-alignment information becomes closer to a hard segmentation. In this scenario, we would like to search for the trade-off between data quantity and matrices quality.

Finally, from a more technological perspective, this work opens perspectives for direct alignment between speech (or phoneme symbols) in a source language and text in a target language. This question is very important if we want to be able to directly translate speech without going through the source language transcription, as discussed in [11, 35].

— 6 —
Appendix

6.1 Table Results

	TOKENS			TYPES		
	Recall	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
Base Model	6.82	5.85	6.30	15.00	6.76	9.32
Reverse Model	**22.29**	**17.15**	**19.39**	**27.09**	**15.01**	**19.32**
Alignment Fusion	19.42	10.51	13.64	18.33	14.91	16.45
Alignment Fusion (SB)	19.75	16.01	17.68	25.41	14.75	18.67
Alignment Fusion (SBA)	17.14	16.56	16.84	26.45	13.76	18.11

Table 6.1 – Comparison between our base and reverse model results and the word discovery results obtained by merging their soft-alignment probability matrices. SB and SBA are acronyms to "smoothed before" and "smoothed before and after", respectively.

	TOKENS			TYPES		
	Recall	**Precision**	**F-score**	**Recall**	**Precision**	**F-score**
Reverse Model	22.29	17.15	19.39	27.09	15.01	19.32
Lemmatized Version	19.25	14.31	16.42	25.28	13.88	17.92

Table 6.2 – Results for the task of unsupervised word discovery for the reverse model and a version using lemmas for the target input.

# of words	**Precision**	**Recall**	**F-score**	**# correct tokens**	**# generated tokens**
0	22.29	17.15	19.39	6,150	35,844
100	60.79	39.16	47.64	16,766	42,804

Table 6.3 – Tokens results for the task of semi-supervised word discovery.

6.2 Paper Accepted to the WiNLP Workshop (ACL 2017 in Vancouver, Canada)

Unsupervised Word Discovery Using Attentional Encoder-Decoder Models

Marcely Zanon Boito
LIG, Grenoble Alpes University
Institute of Informatics, UFRGS
marcely.zanon-boito@grenoble-inp.org

Laurent Besacier
LIG, Grenoble Alpes University
laurent.besacier@imag.fr

Aline Villavicencio
Institute of Informatics, UFRGS
avillavicencio@inf.ufrgs.br

Abstract

Attention-based sequence-to-sequence neural machine translation systems have been shown to jointly align and translate source sentences into target sentences. In this project we use unsegmented symbol sequences (characters and phonemes) as source, aiming to explore the soft-alignment probability matrices generated during training and to evaluate if these soft-alignments allow us to discover latent lexicon representations.

If successful, such approach could be useful for documenting unwritten and/or endangered languages. However, for this to be feasible, attention models should be robust to low-resource scenarios, of several thousand of sentences only. We use a parallel corpus between the endangered language Mboshi and French, as well as a larger and more controlled English-French parallel corpus. Our goal is to explore different representation levels and study their impact, together with the impact of different data set sizes, in the quality of the generated soft-alignment probability matrices.

1 Introduction

The general consensus between specialists is that there are around 7000 languages currently spoken in the world, and between 50 and 90% of them will become extinct by the year 2100 (Austin and Sallabank, 2011). Even with a joint effort from the linguistics community, manually documenting all these languages before their extinction is not feasible. Recently, initiatives for helping with this issue include organizing tasks (Versteegh et al., 2016; Jansen et al., 2013) and offering tools and methodologies to help to automate (part of) this documentation process (Besacier et al., 2006; Bartels et al., 2016; Bansal et al., 2016; Lignos and Yang, 2010; Anastasopoulos and Chiang, 2017).

Endangered languages are characterized by the small number of speakers and often by the lack of a well-defined written form, which makes their documentation an even more challenging task. To deal with the absence of standard written form, collected speech can be aligned to its translation in a well-documented language. The resulting parallel corpora, though, usually lack size.

Nonparametric bayesian models (Goldwater et al., 2009; Lee et al., 2015; Elsner et al., 2013; Adams et al., 2015, 2016) and Neural Network systems (Duong et al., 2016; Bérard et al., 2016; Franke et al., 2016) emerged as popular approaches for phonetic unit discovery, unsupervised segmentation and lexicon discovery, common subtasks to the documentation process. Our ongoing project covers unsupervised segmentation and lexicon discovery, and we are interested in examining the performance impact of executing these tasks from different representation levels. We approach grapheme and phonetic representation, and later we wish to extend our methodology to raw speech.

In this work, we present preliminary results using the attention models soft-alignment probability matrices from a global attention-based sequence-to-sequence Neural Machine Translation (NMT) system as the starting point in our unsupervised segmentation process. We investigate if this approach is realistic using a small corpus from an endangered language, and we compare our results against a nonparametric bayesian model (Goldwater et al., 2009).

We define our architecture in a way that allows us to easily extend it for working directly from raw signal (Bérard et al., 2016; Weiss et al., 2017) in the future, which would be ideal for endangered languages that lack written form. We are also interested in discovering how much data is necessary to achieve good segmentation and lexicon discov-

ery results, and consequently, how applicable this approach is to the endangered languages case.

2 Related Work

Encoder-decoder NMT architectures using attention were first presented in Bahdanau et al. (2014), and we use the implementation of Bérard et al. (2016), an end-to-end translation architecture that can work directly from raw speech. Attention-based NMT systems are known for producing not only good translations, but also attentional information in the form of soft-alignment probability matrices. They demonstrate how these architectures jointly learn to align and translate. We believe this information can be useful for both segmentation and lexicon discovery.

The work by Duong et al. (2016) is the most similar to ours. They also used attention models for their unsupervised segmentation task, achieving very good results compared to three baselines. The Spanish-English parallel corpus used in their work was approximately 18,300 sentences long.

In comparison to that, in this project we use a small parallel corpus from a real unwritten language, for which we study the applicability of the proposed approach for language documentation considering the limitations in data size.

3 Methodology and Preliminary Results

We use a 5,157 sentences parallel corpus in an unwritten[1] African language called Mboshi (Bantu C 25), aligned to French translations on sentence level. Mboshi is a language spoken in the north of the Republic of the Congo, and it counts with 32 different phonemes (25 consonants and 7 vowels) and two tones (high and low). The corpus was recorded using the LIG-AIKUMA tool (Blachon et al., 2016) in the scope of the BULB project (Adda et al., 2016), and preliminary experiments were reported by Godard et al. (2016).

Our approach consists of using the entire corpus for training[2] a global attention sequence-to-sequence NMT system, leaving nothing for testing, since we are not interested in the translations. Then we extract the soft-alignments probability matrices for all the sentences used for training, and we use these matrices to transform the soft-alignment information in hard alignment. We do so by selecting the target word that maximizes the probability of the input symbol given all the target possibilities.

	Recall	**Precision**	**F-Score**
base_model	6.53	3.17	4.27
base_s	8.39	5.38	6.56
reverse	20.04	10.02	13.36
reverse_s	22.29	17.15	19.39
dpseg*	19.73	36.20	25.54

Table 1: Results for the unsupervised segmentation task of tokens using 4,643 parallel sentences. The "s" identifies the models' smoothed versions.

In order to validate our architecture we executed a version using the gold standard segmentation for Mboshi as source. That allowed us to discover if our data set was enough to generate good soft-alignments in the ideal scenario where we already have the segmentation. For this analysis, the evaluation was qualitative, and we observed very good alignments between known Mboshi words and their translations[3].

For a more realistic setup we replaced the source by its unsegmented version. The results had noisy and unhelpful soft-alignment probability matrices, what can be verified by precision and recall being both low. We also trained a model using the alignment smoothing described in Duong et al. (2016), what helped the model's performance. The results are respectively base_model and base_s at Table 1.

In more details, this alignment smoothing is applied by training the model with a temperature factor in the softmax function. The resulting probability matrices are further smoothed by replacing each probability α_{ij} by $\frac{1}{3}(\alpha_{i,j-1} + \alpha_{i,j} + \alpha_{i,j+1})$, i and j being respectively the target words and source symbols indexes.

Evaluating the matrices generated by this first model, we observed that the system was consistently ignoring part of the source symbols when generating the translation. In NMT systems, the soft-alignments are created forcing the probabilities for each target word j to sum to one, what ensures all the target words are used. However, there is no similar constraint for the source symbols, as discussed in Duong et al. (2016).

[1] Even if it is unwritten, we have a non-standard grapheme form, considered to be close to the language phonology.

[2] 10% for development set, which corresponds to 514 sentences, and the remaining 4,643 sentences for training.

[3] We had access to a small Mboshi-French dictionary (Beapami et al., 2000).

Considering that we are interested in segmentation, our system must use all these source units from the unsegmented input when processing a sentence. To solve this, we reversed the system input, creating a French-Mboshi words-to-characters system. As we can see in Table 1, this constraint impacted greatly in the segmentation performance. The addition of alignment smoothing further improved the system performance.

Finally, for comparison, we executed the nonparametric bayesian model implemented in dpseg[4] (Goldwater et al., 2009), using it as an out-of-the-box tool. We used default configurations for the bigram model and 20,000 iterations. We considered the achieved result to be a lower bound result for this technique in this scenario.

The out-of-the-box trained nonparametric bayesian model presented better overall results than our reverse neural model. This is consistent and expected, since bayesian models are known for being able to achieve good segmentation with small amounts of data. In the other hand, neural approaches are known for needing large data sets to train their parameters.

Moreover, even if we still can apply some optimizations to our model, we do believe there is a limit of how much is achievable with this amount of data. Unpublished results to which we had access in our laboratory investigated the dependency between data set size and the soft-alignment probability matrices quality.

It seems that, even when the model performs well in translation, sometimes that amount of data is not enough to create consistent soft-alignment matrices, and the network learns a global sentence representation which is not meaningful for us. In that case, adding more data to the model seems to make the soft-alignment matrices "converge" to the desirable representation.

4 Conclusion and Future Experiments

In this work we presented our preliminary results approaching the task of unsupervised segmentation. We used a neural machine translation system to retrieve soft-alignment information using a data set from a real endangered language.

By reversing source and target languages, we were able to achieve interesting results considering the amount of data available. However, these results are still inferior to what we can achieve with bayesian systems such as dpseg.

We are following our experiments by using a large English-French corpus to study the impact that more data can have in the soft-alignment matrices quality. Doing so, we want to narrow down the amount of data needed in order to retrieve enough information from these alignments. This will answer how applicable this approach is for language documentation scenarios.

We are also investigating a semi-supervised approach. We believe that by offering some already segmented units (such as function words), we could improve the system's performance segmenting the rest of the vocabulary.

Finally, we want to investigate how different representation levels for the source impact the amount of data needed, and if it is possible to achieve good results working directly from speech. In this scenario, we would like to explore how reducing information in the target side, by replacing the translations by their lemmas or part of speech, could help to decrease the amount of data needed for this task.

[4] Available at http://homepages.inf.ed.ac.uk/sgwater/.

References

Oliver Adams, Graham Neubig, Trevor Cohn, and Steven Bird. 2015. Inducing bilingual lexicons from small quantities of sentence-aligned phonemic transcriptions. In *12th International Workshop on Spoken Language Translation (IWSLT)*.

Oliver Adams, Graham Neubig, Trevor Cohn, and Steven Bird. 2016. Learning a translation model from word lattices. In *17th Annual Conference of the International Speech Communication Association (INTERSPEECH 2016), San Francisco, California, USA*.

Gilles Adda, Sebastian Stüker, Martine Adda-Decker, Odette Ambouroue, Laurent Besacier, David Blachon, Hélène Bonneau-Maynard, Pierre Godard, Fatima Hamlaoui, Dmitry Idiatov, et al. 2016. Breaking the unwritten language barrier: The bulb project. *Procedia Computer Science* 81:8–14.

Antonios Anastasopoulos and David Chiang. 2017. A case study on using speech-to-translation alignments for language documentation. *arXiv preprint arXiv:1702.04372* .

Peter K Austin and Julia Sallabank. 2011. *The Cambridge handbook of endangered languages*. Cambridge University Press.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly

learning to align and translate. *arXiv preprint arXiv:1409.0473* .

Sameer Bansal, Herman Kamper, Sharon Goldwater, and Adam Lopez. 2016. Weakly supervised spoken term discovery using cross-lingual side information. *arXiv preprint arXiv:1609.06530* .

Chris Bartels, Wen Wang, Vikramjit Mitra, Colleen Richey, Andreas Kathol, Dimitra Vergyri, Harry Bratt, and Chiachi Hung. 2016. Toward human-assisted lexical unit discovery without text resources. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, pages 64–70.

Roch Paulin Beapami, Ruth Chatfield, Guy-Noël Kouarata, and Andrea Embengue-Waldschmidt. 2000. *Dictionnaire Mbochi-Franais*. SIL-Congo Publishers.

Alexandre Bérard, Olivier Pietquin, Christophe Servan, and Laurent Besacier. 2016. Listen and translate: A proof of concept for end-to-end speech-to-text translation. *arXiv preprint arXiv:1612.01744* .

Laurent Besacier, Bowen Zhou, and Yuqing Gao. 2006. Towards speech translation of non written languages. In *Spoken Language Technology Workshop, 2006. IEEE*. IEEE, pages 222–225.

David Blachon, Elodie Gauthier, Laurent Besacier, Guy-Noël Kouarata, Martine Adda-Decker, and Annie Rialland. 2016. Parallel speech collection for under-resourced language studies using the lig-aikuma mobile device app. *Procedia Computer Science* 81:61–66.

Long Duong, Antonios Anastasopoulos, David Chiang, Steven Bird, and Trevor Cohn. 2016. An attentional model for speech translation without transcription. In *Proceedings of NAACL-HLT*. pages 949–959.

Micha Elsner, Sharon Goldwater, Naomi Feldman, and Frank Wood. 2013. A joint learning model of word segmentation, lexical acquisition, and phonetic variability. In *Proc. EMNLP*.

Joerg Franke, Markus Mueller, Fatima Hamlaoui, Sebastian Stueker, and Alex Waibel. 2016. Phoneme boundary detection using deep bidirectional lstms. In *Speech Communication; 12. ITG Symposium; Proceedings of* . VDE, pages 1–5.

Pierre Godard, Gilles Adda, Martine Adda-Decker, Alexandre Allauzen, Laurent Besacier, Helene Bonneau-Maynard, Guy-Noël Kouarata, Kevin Löser, Annie Rialland, and François Yvon. 2016. Preliminary experiments on unsupervised word discovery in mboshi. In *Interspeech 2016*.

Sharon Goldwater, Thomas L Griffiths, and Mark Johnson. 2009. A bayesian framework for word segmentation: Exploring the effects of context. *Cognition* 112(1):21–54.

Aren Jansen, Emmanuel Dupoux, Sharon Goldwater, Mark Johnson, Sanjeev Khudanpur, Kenneth Church, Naomi Feldman, Hynek Hermansky, Florian Metze, Richard Rose, et al. 2013. A summary of the 2012 jhu clsp workshop on zero resource speech technologies and models of early language acquisition .

Chia-ying Lee, Timothy J O'Donnell, and James Glass. 2015. Unsupervised lexicon discovery from acoustic input. *Transactions of the Association for Computational Linguistics* 3:389–403.

Constantine Lignos and Charles Yang. 2010. Recession segmentation: simpler online word segmentation using limited resources. In *Proceedings of the fourteenth conference on computational natural language learning*. Association for Computational Linguistics, pages 88–97.

Maarten Versteegh, Xavier Anguera, Aren Jansen, and Emmanuel Dupoux. 2016. The zero resource speech challenge 2015: Proposed approaches and results. *Procedia Computer Science* 81:67–72.

Ron J Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. 2017. Sequence-to-sequence models can directly transcribe foreign speech. *arXiv preprint arXiv:1703.08581* .

Bibliography

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] Oliver Adams, Graham Neubig, Trevor Cohn, and Steven Bird. Inducing bilingual lexicons from small quantities of sentence-aligned phonemic transcriptions. In *12th International Workshop on Spoken Language Translation (IWSLT)*, 2015.

[3] Oliver Adams, Graham Neubig, Trevor Cohn, Steven Bird, Quoc Truong Do, and Satoshi Nakamura. Learning a lexicon and translation model from phoneme lattices.

[4] Gilles Adda, Sebastian Stüker, Martine Adda-Decker, Odette Ambouroue, Laurent Besacier, David Blachon, Hélène Bonneau-Maynard, Pierre Godard, Fatima Hamlaoui, Dmitry Idiatov, et al. Breaking the unwritten language barrier: The bulb project. *Procedia Computer Science*, 81:8–14, 2016.

[5] Celestin Amboulou. Le mbochi: langue bantoue du congo brazzaville (zone c, groupe c20). *PhD diss, INALCO, Paris*, 1998.

[6] Peter K Austin and Julia Sallabank. *The Cambridge handbook of endangered languages*. Cambridge University Press, 2011.

[7] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[8] Chris Bartels, Wen Wang, Vikramjit Mitra, Colleen Richey, Andreas Kathol, Dimitra Vergyri, Harry Bratt, and Chiachi Hung. Toward human-assisted lexical unit discovery without text resources. In *Spoken Language Technology Workshop (SLT), 2016 IEEE*, pages 64–70. IEEE, 2016.

[9] Roch Paulin Beapami, Ruth Chatfield, Guy-Noël Kouarata, and Andrea Embengue-Waldschmidt. *Dictionnaire Mbochi-Français*. SIL-Congo Publishers, 2000.

[10] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[11] Alexandre Bérard, Olivier Pietquin, Christophe Servan, and Laurent Besacier. Listen and translate: A proof of concept for end-to-end speech-to-text translation. *arXiv preprint arXiv:1612.01744*, 2016.

[12] David Blachon, Elodie Gauthier, Laurent Besacier, Guy-Noël Kouarata, Martine Adda-Decker, and Annie Rialland. Parallel speech collection for under-resourced language studies using the lig-aikuma mobile device app. *Procedia Computer Science*, 81:61–66, 2016.

[13] Luc Bouquiaux and Jacqueline MC Thomas. *Enquête et description des langues à tradition orale*, volume 2. Peeters Publishers, 1976.

[14] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[15] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In *Advances in Neural Information Processing Systems*, pages 577–585, 2015.

[16] Long Duong, Antonios Anastasopoulos, David Chiang, Steven Bird, and Trevor Cohn. An attentional model for speech translation without transcription. In *Proceedings of NAACL-HLT*, pages 949–959, 2016.

[17] Micha Elsner, Sharon Goldwater, Naomi Feldman, and Frank Wood. A joint learning model of word segmentation, lexical acquisition, and phonetic variability. In *Proc. EMNLP*, 2013.

[18] Joerg Franke, Markus Mueller, Fatima Hamlaoui, Sebastian Stueker, and Alex Waibel. Phoneme boundary detection using deep bidirectional lstms. In *Speech Communication; 12. ITG Symposium; Proceedings of*, pages 1–5. VDE, 2016.

[19] Pierre Godard, Gilles Adda, Martine Adda-Decker, Alexandre Allauzen, Laurent Besacier, Helene Bonneau-Maynard, Guy-Noël Kouarata, Kevin Löser, Annie Rialland, and François Yvon. Preliminary experiments on unsupervised word discovery in mboshi. In *Interspeech 2016*, 2016.

[20] Sharon Goldwater, Thomas L Griffiths, and Mark Johnson. A bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112(1):21–54, 2009.

[21] Sharon J Goldwater. *Nonparametric Bayesian models of lexical acquisition*. PhD thesis, Citeseer, 2007.

[22] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[23] Aren Jansen, Emmanuel Dupoux, Sharon Goldwater, Mark Johnson, Sanjeev Khudanpur, Kenneth Church, Naomi Feldman, Hynek Hermansky, Florian Metze, Richard Rose, et al. A summary of the 2012 jhu clsp workshop on zero resource speech technologies and models of early language acquisition. 2013.

[24] Mark Johnson and Sharon Goldwater. Improving nonparameteric bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 317–325. Association for Computational Linguistics, 2009.

[25] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[26] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics, 2007.

[27] Chia-ying Lee, Timothy J O'Donnell, and James Glass. Unsupervised lexicon discovery from acoustic input. *Transactions of the Association for Computational Linguistics*, 3:389–403, 2015.

[28] Graham Neubig. Neural machine translation and sequence-to-sequence models: A tutorial. *arXiv preprint arXiv:1703.01619*, 2017.

[29] Christopher Olah. Understanding lstm networks. http://colah.github.io/posts/2015-08-Understanding-LSTMs/. Published August 27, 2015. Accessed April 12, 2017.

[30] Lucas Ondel, Lukáš Burget, and Jan Černockỳ. Variational inference for acoustic unit discovery. *Procedia Computer Science*, 81:80–86, 2016.

[31] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

[32] David MW Powers. Applications and explanations of zipf's law. In *Proceedings of the joint conferences on new methods in language processing and computational natural language learning*, pages 151–160. Association for Computational Linguistics, 1998.

[33] Helmut Schmid. Probabilistic part-ofispeech tagging using decision trees. In *New methods in language processing*, page 154. Routledge, 2013.

[34] Maarten Versteegh, Xavier Anguera, Aren Jansen, and Emmanuel Dupoux. The zero resource speech challenge 2015: Proposed approaches and results. *Procedia Computer Science*, 81:67–72, 2016.

[35] Ron J Weiss, Jan Chorowski, Navdeep Jaitly, Yonghui Wu, and Zhifeng Chen. Sequence-to-sequence models can directly transcribe foreign speech. *arXiv preprint arXiv:1703.08581*, 2017.

[36] Marcely Zanon Boito, Laurent Besacier, and Aline Villavicencio. Unsupervised word discovery using attentional encoder-decoder models. In *Women and Underrepresented Minorities in Natural Language Processing ACL Workshop*, 2017.

[37] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*, 2014.

REFERÊNCIAS

ADAMS, O. et al. Learning a lexicon and translation model from phoneme lattices.

ADAMS, O. et al. Inducing bilingual lexicons from small quantities of sentence-aligned phonemic transcriptions. In: **12th International Workshop on Spoken Language Translation (IWSLT)**. [S.l.: s.n.], 2015.

ADDA, G. et al. Breaking the unwritten language barrier: The bulb project. **Procedia Computer Science**, Elsevier, v. 81, p. 8–14, 2016.

ANASTASOPOULOS, A.; CHIANG, D. A case study on using speech-to-translation alignments for language documentation. **arXiv preprint arXiv:1702.04372**, 2017.

AUSTIN, P. K.; SALLABANK, J. **The Cambridge handbook of endangered languages**. [S.l.]: Cambridge University Press, 2011.

BAHDANAU, D.; CHO, K.; BENGIO, Y. Neural machine translation by jointly learning to align and translate. **arXiv preprint arXiv:1409.0473**, 2014.

BANSAL, S. et al. Weakly supervised spoken term discovery using cross-lingual side information. **arXiv preprint arXiv:1609.06530**, 2016.

BARTELS, C. et al. Toward human-assisted lexical unit discovery without text resources. In: IEEE. **Spoken Language Technology Workshop (SLT), 2016 IEEE**. [S.l.], 2016. p. 64–70.

BÉRARD, A. et al. Listen and translate: A proof of concept for end-to-end speech-to-text translation. **arXiv preprint arXiv:1612.01744**, 2016.

BESACIER, L.; ZHOU, B.; GAO, Y. Towards speech translation of non written languages. In: IEEE. **Spoken Language Technology Workshop, 2006. IEEE**. [S.l.], 2006. p. 222–225.

BLACHON, D. et al. Parallel speech collection for under-resourced language studies using the lig-aikuma mobile device app. **Procedia Computer Science**, Elsevier, v. 81, p. 61–66, 2016.

BOITO, M. Z. et al. Unwritten languages demand attention too! word discovery with encoder-decoder models. **arXiv preprint arXiv:1709.05631**, 2017.

BOITO, M. Z.; BESACIER, L.; VILLAVICENCIO, A. Unsupervised word discovery using attentional encoder-decoder models. In: **Women and Underrepresented Minorities in Natural Language Processing ACL Workshop**. [S.l.: s.n.], 2017.

DUONG, L. et al. An attentional model for speech translation without transcription. In: **Proceedings of NAACL-HLT**. [S.l.: s.n.], 2016. p. 949–959.

ELSNER, M. et al. A joint learning model of word segmentation, lexical acquisition, and phonetic variability. In: **Proc. EMNLP**. [S.l.: s.n.], 2013.

FRANKE, J. et al. Phoneme boundary detection using deep bidirectional lstms. In: VDE. **Speech Communication; 12. ITG Symposium; Proceedings of**. [S.l.], 2016. p. 1–5.

GODARD, P. et al. Preliminary experiments on unsupervised word discovery in mboshi. In: **Interspeech 2016**. [S.l.: s.n.], 2016.

GOLDWATER, S.; GRIFFITHS, T. L.; JOHNSON, M. Contextual dependencies in unsupervised word segmentation. In: **Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics**. Sydney, Australia: [s.n.], 2006. p. 673–680.

GOLDWATER, S.; GRIFFITHS, T. L.; JOHNSON, M. A bayesian framework for word segmentation: Exploring the effects of context. **Cognition**, Elsevier, v. 112, n. 1, p. 21–54, 2009.

GOLDWATER, S.; GRIFFITHS, T. L.; JOHNSON, M. A Bayesian framework for word segmentation: Exploring the effects of context. **Cognition**, v. 112, n. 1, p. 21–54, 2009.

GOLDWATER, S. J. **Nonparametric Bayesian models of lexical acquisition**. Thesis (PhD) — Citeseer, 2007.

JANSEN, A. et al. A summary of the 2012 jhu clsp workshop on zero resource speech technologies and models of early language acquisition. IEEE, 2013.

JOHNSON, M.; GOLDWATER, S. Improving nonparameteric bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics**. [S.l.], 2009. p. 317–325.

KOEHN, P. et al. Moses: Open source toolkit for statistical machine translation. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions**. [S.l.], 2007. p. 177–180.

KOEHN, P.; KNOWLES, R. Six challenges for neural machine translation. **CoRR**, abs/1706.03872, 2017. Available from Internet: <http://arxiv.org/abs/1706.03872>.

LEE, C.-y.; O'DONNELL, T. J.; GLASS, J. Unsupervised lexicon discovery from acoustic input. **Transactions of the Association for Computational Linguistics**, v. 3, p. 389–403, 2015.

LEE, J.; CHO, K.; HOFMANN, T. Fully character-level neural machine translation without explicit segmentation. **CoRR**, abs/1610.03017, 2016. Available from Internet: <http://arxiv.org/abs/1610.03017>.

LIGNOS, C.; YANG, C. Recession segmentation: simpler online word segmentation using limited resources. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the fourteenth conference on computational natural language learning**. [S.l.], 2010. p. 88–97.

PAPINENI, K. et al. Bleu: a method for automatic evaluation of machine translation. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the 40th annual meeting on association for computational linguistics**. [S.l.], 2002. p. 311–318.

POWERS, D. M. Applications and explanations of zipf's law. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the joint conferences on new methods in language processing and computational natural language learning**. [S.l.], 1998. p. 151–160.

SCHMID, H. Probabilistic part-ofispeech tagging using decision trees. In: ROUTLEDGE. **New methods in language processing**. [S.l.], 2013. p. 154.

VERSTEEGH, M. et al. The zero resource speech challenge 2015: Proposed approaches and results. **Procedia Computer Science**, Elsevier, v. 81, p. 67–72, 2016.

WANG, L. et al. Morphological segmentation with window lstm neural networks. In: **AAAI**. [S.l.: s.n.], 2016. p. 2842–2848.

WEISS, R. J. et al. Sequence-to-sequence models can directly transcribe foreign speech. **arXiv preprint arXiv:1703.08581**, 2017.