Exploiting Semantic and Topic Context to Improve Recognition of Proper Names in Diachronic Audio Documents

Imran Sheikh

UNIVERSITÉ DE LORRAINE

École doctorale IAEM Lorraine

Exploitation du contexte sémantique pour améliorer la reconnaissance des noms propres dans les documents audio diachroniques

(Exploiting Semantic and Topic Context to Improve Recognition of Proper Names in Diachronic Audio Documents)

THÈSE

présentée et soutenue publiquement le 24 novembre 2016

pour l'obtention du

Doctorat de l'Université de Lorraine

(mention informatique)

par

Imran Sheikh

Composition du jury

Rapporteurs :	Dietrich Klakow	Prof., Universität des Saarlandes
	Kris Demuynck	Prof., Universiteit Gent
Examinateurs :	Christian Wellekens	Prof. Emer., Institut Eurécom
	Dominique Fohr	Dr., CNRS Nancy
	Georges Linarès	Prof., Université d'Avignon
Directeur de thèse :	Irina Illina	MCF HDR, Université de Lorraine

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Résumé

La nature diachronique des bulletins d'information provoque de fortes variations du contenu linguistique et du vocabulaire dans ces documents. Dans le cadre de la reconnaissance automatique de la parole, cela conduit au problème de mots hors vocabulaire (*Out-Of-Vocabulary*, OOV). La plupart des mots OOV sont des noms propres. Les noms propres sont très importants pour l'indexation automatique de contenus audio-vidéo. De plus, leur bonne identification est importante pour des transcriptions automatiques fiables. Le but de cette thèse est de proposer des méthodes pour récupérer les noms propres manquants dans un système de reconnaissance. Ces mots seront intégrés au lexique du système de reconnaissance pour effectuer une deuxième passe de reconnaissance. Comme dans la littérature, nous allons utiliser des documents textuels récupérés sur Internet pour sélectionner de nouveaux mots. Les méthodologies existantes sont fondées sur des matrices terme-document ou des co-occurrences de mots pour retrouver des nouveaux mots. Dans cette thèse nous proposons de modéliser le contexte sémantique et d'utiliser des informations thématiques contenus dans les documents audio à transcrire. Des modèles probabilistes de thème (*topic model*) et des projections dans un espace continu obtenues à l'aide de réseaux de neurones (*word embeddings*) sont explorés pour la tâche de récupération des noms propres pertinents. Une évaluation approfondie de ces représentations contextuelles a été réalisée. Pour modéliser le contexte de nouveaux mots plus efficacement, nous proposons des réseaux de neurones qui maximisent la récupération des noms propres pertinents. Les modèles de neurones (*Neural Bag-of-Words*, NBOW) modélisant les représentations contextuelles au niveau du document obtiennent de très bonnes performances. En s'appuyant sur ce modèle, nous proposons un nouveau modèle (*Neural Bag-of-Weighted-Words*, NBOW2) qui permet d'estimer un degré d'importance pour chacun des mots du document et a la capacité de capturer des mots spécifiques à ce document. Des expériences de reconnaissance automatique de bulletins d'information télévisés montrent l'efficacité du modèle proposé. L'évaluation de NBOW2 sur d'autres tâches telles que la classification de textes, l'analyse des critiques des films et la classification thématique des textes issus de groupes des discussions, montre des bonnes performances. Ce modèle donne des meilleurs résultats que les modèles utilisant des sac-de-mots.

Mots-clés: Reconnaissance de la parole, noms propres, OOV, sémantique distributive

Abstract

The diachronic nature of broadcast news causes frequent variations in the linguistic content and vocabulary, leading to the problem of Out-Of-Vocabulary (OOV) words in automatic speech recognition. Most of the OOV words are found to be proper names whereas proper names are important for automatic indexing of audio-video content as well as for obtaining reliable automatic transcriptions. New proper names missed by the speech recognition system can be recovered by a dynamic vocabulary multi-pass recognition approach in which new proper names are added to the speech recognition vocabulary based on the context of the spoken content. Existing methods for vocabulary selection rely on web search engines and adaptation corpora and choose the new vocabulary words using term-document frequency and co-occurrence based features. Open vocabulary systems based on sub-word units are an interesting solution but they face the problem of producing a reliable text transcription. The goal of this thesis is to model the semantic and topical context of new proper names in order to retrieve those which are relevant to the spoken content in the audio document. Training semantic/topic models is a challenging problem in this task because (a) several new proper names come with a low amount of data and (b) the context model should be robust to word errors in the automatic transcription. Probabilistic topic models and word embeddings from neural network models are explored for the task of retrieval of relevant proper names. A thorough evaluation of contextual representations from these models is performed. It is argued that these representations, which are learned in an unsupervised manner, are not the best for the given retrieval task. Neural network context models trained with an objective to maximise the retrieval performance are proposed. A Neural Bag-of-Words (NBOW) model trained to learn context vector representations at a document level is shown to outperform the generic representations. The proposed Neural Bag-of-Weighted-Words (NBOW2) model learns to assign a degree of importance to input words and has the ability to capture task specific key-words. Experiments on automatic speech recognition on French broadcast news videos demonstrate the effectiveness of the proposed models. Further evaluation of the NBOW2 model on standard text classification tasks, including movie review sentiment classification and newsgroup topic classification, shows that it learns interesting information about the task and gives the best classification accuracies among the bag-of-words models.

Keywords: speech recognition, OOV, proper names, distributional semantics

Acknowledgements

I would like to thank all the people who contributed in their own particular way to the achievement of this doctoral thesis.

First and foremost, I thank my doctoral supervisors Irina Illina, Dominique Fohr and Georges Linarès. I greatly appreciate the freedom they gave me to do my research, and the encouragement and support that they always offered me. The enthusiasm they had for this research topic was highly motivational. Their insights, advice, supervision and discussion have all been important for the materialisation of this thesis.

I want to thank my thesis committee members, Dietrich Klakow, Kris Demuynck, Christian Wellekens and Jean Lieber, for investing their time and providing valuable feedback. It was really great to have them in my thesis committee.

It was a great experience to be part of two amazing research groups: the Multispeech group at LORIA-INRIA, Nancy and the language processing group at LIA, Avignon. I would like to thank all the members of these groups for the friendly and fruitful environment they created. I owe special thanks to Emmanuel Vincent for his thoughtful discussions and guidance. My heartfelt thanks to Aditya Arie Nugraha and also to Jen-Yu Liu, Sunit Sivasakaran and Dung Tran for all those white board discussions which created an enjoyable and stimulating research environment in our lab. Thank you Yann Prono and Siddharth Dalmia for your contributions in my experiments. I also want to take a moment to thank Aghilas Sini, Motaz Saad and Arseniy Gorin for being great friends and offering their extended support.

My thanks also go out to the sources of financial support that I received during my doctoral studies. The research in this thesis was mainly supported by the French National Research Agency (ANR) ContNomina project under contract ANR-12-BS02-0009. I would also like to thank Sunil Kumar Kopparapu and TCS Innovation Labs for encouraging and supporting me to pursue my doctoral studies.

Most of all, I want to express my deepest gratitude to my family who supported me throughout. Especially, I want to thank my loving wife Atiya for her unconditional support and encouragement. Without her patience, love and sacrifices I could not have achieved this thesis. I am forever indebted to my parents for giving me all those opportunities that have brought me here.

Contents

List of Tables

List of Figures

Acronyms

AM	Acoustic Model
ASR	Automatic Speech Recognition
BOW	Bag-Of-Words
CBOW	Continuous Bag-Of-Words
IV	In Vocabulary
LDA	Latent Dirichlet Allocation
LM	Language Model
LSA	Latent Semantic Analysis
LVCSR	Large Vocabulary Continuous Speech Recognition
MAP	Mean Average Precision
NBOW	Neural Bag-Of-Words
NBOW2	Neural Bag-Of-Weighted-Words
OOV	Out-Of-Vocabulary
OOVPNER	Out-Of-Vocabulary Proper Name Error Rate
PLSA	Probabilistic Latent Semantic Analysis
PMI	Pointwise Mutual Information
PN	Proper Name
PNER	Proper Name Error Rate
POS	Part-Of-Speech
RP	Random Projections
SVD	Singular Value Decomposition
TF-IDF	Term-Frequency Inverse-Document-Frequency
VSM	Vector Space Model
WER	Word Error Rate
WFST	Weighted Finite State Transducer

Synthèse en Français

1 Introduction

Les bulletins d'information sont diachroniques par nature et sont caractérisés par des changements continus de thèmes et de contenus. Les variations fréquentes dans le contenu linguistique et le vocabulaire posent un défi pour la reconnaissance automatique de la parole (*Large Vocabulary Continuous Speech Recognition*, LVCSR). Tous les mots existants dans un langage ne peuvent pas être inclus dans le vocabulaire et le *Modèle de Langage* (*Language Model*, LM) d'un système de LVCSR, car

- il y a beaucoup des mots nouveaux/rares, particulièrement des noms propres (*Proper Names*, PN) ;
- leur inclusion augmenterait l'espace de recherche du LVCSR et sa complexité sans garantir une diminution du taux d'erreur de mots (*Word Error Rate*, WER).

Le choix pratique consiste à ajouter qu'une partie de ces mots au vocabulaire. Ce qui conduit au problème des mots hors vocabulaire (*Out-Of-Vocabulary*, OOV) pour le LVCSR. Une analyse des mots OOV révèle que la majorité des mots OOV (56-72% [Sheikh et al., 2015b]) sont des PN. Ces noms propres sont très importants pour l'indexation automatique des contenus audio-vidéo, ainsi que pour l'obtention des transcriptions automatiques précises et fiables. Dans cette thèse, nous étudions le problème suivant : comment récupérer de nouveaux OOV PN à partir de documents audio diachroniques.

Dans le cadre de la reconnaissance de la parole, les méthodes de récupération des mots OOV peuvent être classées en deux catégories :

- les approches fondées sur la détection des OOVs ;
- les approches fondées sur la sélection de vocabulaire.

Les approches fondées sur la *détection des OOVs* [Rastrow et al., 2009b, Qin and Rudnicky, 2012, Parada et al., 2010a, Kombrink et al., 2012, Chen et al., 2013b] ont pour but de détecter la présence de mots OOV et/ou localiser des

régions OOV dans l'hypothèse du LVCSR. Ensuite, le mot correspondant à l'OOV est recherché. Ces approches utilisent principalement des modèles de langage hybrides : modèles de mots et de sous-mots (par exemple, des syllabes). Ces mêmes idées sont utilisées pour les systèmes de reconnaissance avec un vocabulaire ouvert [Bisani and Ney, 2005, Shaik et al., 2015]. Cependant, ces méthodes de détection des OOVs utilisent l'information obtenue à la fin du processus de reconnaissance, par exemple, les scores a posteriori ou des réseaux de confusion de mots. Des systèmes fondé sur des modèles de langage hybrides peuvent nécessiter une sélection rigoureuse des unités de sous-mots. Parfois, cela peut conduire à une augmentation des taux d'erreurs [Shaik et al., 2015].

Les approches fondées sur la *sélection de vocabulaire* proposent un vocabulaire pertinent en utilisant des textes complémentaires. Pour un corpus spécifique à un domaine, des méthodes intéressantes [Allauzen and Gauvain, 2005a, Liu et al., 2007, Jouvet and Langlois, 2013, Ming Sun, 2015] ont été proposées pour sélectionner le vocabulaire afin de réduire le taux de OOVs. [Martins et al., 2007] propose une méthodologie pour une mise à jour quotidienne du vocabulaire. Les méthodes de sélection de vocabulaire spécifique à un document [Seneff, 2005, Oger et al., 2008b, Meng et al., 2010, Maergner et al., 2012] sont plus dynamiques que les méthodes fondées sur la détection des OOVs, car elles proposent un vocabulaire spécifique au contexte du document.

Dans notre travail, nous avons recueilli un corpus de documents textuels à partir d'Internet. Ce corpus est utilisé pour modéliser le contexte sémantique des noms propres nouveaux apparaissant dans le corpus. Pour un document audio de test, nous déduisons le contexte sémantique du contenu de ce document. Puis, à partir des documents recueillis sur Internet, nous proposons une liste de noms propres OOV qui sont pertinents pour le contexte du document de test. Notre motivation pour explorer des modèles sémantiques et contextuels vient d'une question plus générale et ouverte : *Est-ce que le contexte sémantique pourrait être profitable pour améliorer le processus de transcription automatique ?*

Nous proposons d'analyser la séquence de mots entourant les noms propres OOV et de l'utiliser pour modéliser le contexte sémantique local. Il est également possible d'utiliser un contexte de niveau plus élevé obtenu par une analyse sémantique ou thématique du document entier. Cependant, les transcriptions automatiques du LVCSR contiennent des erreurs de reconnaissance. D'autre part on ne dispose pas d'information sur la position des OOVs. Les approches fondées sur la détection d'OOVs peuvent localiser les régions où se trouvent les OOVs. Dans cette thèse, nous nous intéressons plutôt aux méthodes qui ne nécessitent pas de données étiquetées manuellement. Par conséquent, dans notre approche nous nous appuyons sur le contexte sémantique/thématique de la

transcription du document audio pour récupérer les noms propres OOV qui sont pertinents.

Les modèles sémantiques/thématiques ont été largement étudiés dans le domaine de la Linguistique Informatique et du Traitement Automatique du Langage Naturel, plus particulièrement dans le domaine de la *Sémantique Distributionnelle* [Turney and Pantel, 2010]. Parmi ceux-ci, l'analyse sémantique latente (*Latent Semantic Analysis* LSA) [Deerwester et al., 1990] et l'allocation de Dirichlet latente (*Latent Dirichlet Allocation* LDA) [Blei et al., 2003] sont les plus importantes pour extraire automatiquement des représentations sémantiques/thématiques à partir des documents textuels. Pour obtenir des espaces sémantiques, la LSA utilise des méthodes de décomposition des matrices. La LDA apprend des mélanges et des distributions de thèmes en utilisant une analyse Bayésienne hiérarchique. Récemment, les modèles appelés *Word Embeddings* ont été proposés et permettent de représenter des informations sémantiques ou syntaxiques des mots [Mikolov et al., 2013c].

L'une des principales contributions de cette thèse est le développement d'un ensemble de méthodes permettant de récupérer les noms propres OOV pertinents à partir de documents diachroniques. Ces méthodes s'appuient sur différentes représentations de l'espace sémantique/thématique. Nous procédons à une analyse approfondie de ces représentations pour savoir si elles peuvent être exploitées pour notre tâche de récupération des noms propres OOVs. Nous concentrons notre travail sur deux aspects importants :

- sur la robustesse de ces représentation à des erreurs de reconnaissance, car pour inférer le contexte sémantique/thématique nous nous appuyons sur des hypothèses du LVCSR ;
- sur l'efficacité de ces représentations pour les noms propres OOVs peu fréquents, parce que la grande partie des noms propres OOVs sont peu observés dans les données diachroniques.

La deuxième contribution principale de cette thèse est un ensemble de représentations discriminantes de contexte. Les modèles de *Word Embeddings* et la LDA apprennent des représentations sémantiques/thématiques en optimisant des fonctions de probabilité estimées sue les données d'apprentissage. Tout en maximisant la performance de la récupération des noms propres OOV, nous proposons différentes représentations discriminantes des contextes. Un des nouveaux modèles proposés est est un modèle neuronal de sac de mots pondérés *Bag-of-Weighted-Words* (NBOW2). Ce modèle estime un degré d'importance de chaque mot du document. Ce modèle a la capacité de capturer les mots importants. Outre notre tâche de récupération des noms propres OOV, nous

évaluons ce modèle sur des tâches d'analyse de critiques des films et de classification thématique de textes.

La section 2 présente une description générale de notre approche pour le traitement des OOVs et donne un aperçu des méthodes pour récupérer les OOVs pertinents en utilisant la LDA et les modèles fondées sur des *word embeddings*. La section 3 décrit des représentations contextuelles discriminantes, apprises en utilisant des réseaux de neurones. Le protocole expérimental et la description du système sont donnés dans la section 4. Les résultats de récupération des OOV PN sont discutés dans la section 5. Cette section est suivie des résultats de la reconnaissance de la parole (section 6). La section 7 donne quelques conclusions.

2 Approches proposées

Les méthodes existantes de sélection de vocabulaire spécifique à un document [Seneff, 2005, Oger et al., 2008b, Meng et al., 2010, Maergner et al., 2012, Nkairi et al., 2013] proposent souvent d'utiliser des moteurs de recherche du Web pour récupérer des documents pertinents. Elles choisissent les nouveaux mots à ajouter au vocabulaire en utilisant la fréquence des termes, la fréquence des documents et des calculs de co-occurrences. Dans notre travail pour récupérer les OOV PN pertinents pour un document audio de test, nous proposons de modéliser le contexte des OOV PN. La figure 1 présente un schéma de notre approche. Les articles diachroniques de presse sont collectées à partir d'Internet pour construire un *corpus diachronique de textes* contenant des nouveaux PN. Ce corpus diachronique de textes est utilisé pour apprendre un modèle de contexte qui capture les relations entre les mots du vocabulaire du LVCSR (*In-Vocabulary*, IV) et les OOV PN. Cela constitue la phase d'apprentissage. Pendant la phase de test, le document audio de test est reconnu par le LVCSR (en utilisant le vocabulaire et le LM de base) pour obtenir une transcription (premier passage). En utilisant cette transcription, le contexte du document est inféré par le modèle de contexte. Puis les OOV PN du corpus diachronique sont classés en fonction du contexte. Ensuite, les OOV PN sélectionnés sont utilisés pour mettre à jour le vocabulaire et le LM du LVCSR. Finalement, une deuxième passe de reconnaissance est effectuée avec ce nouveau vocabulaire et ce nouveau modèle de langage. A la place de la deuxième passe de reconnaissance, il est possible d'effectuer une recherche de mots clés, comme décrit dans [Sheikh et al., 2016c].

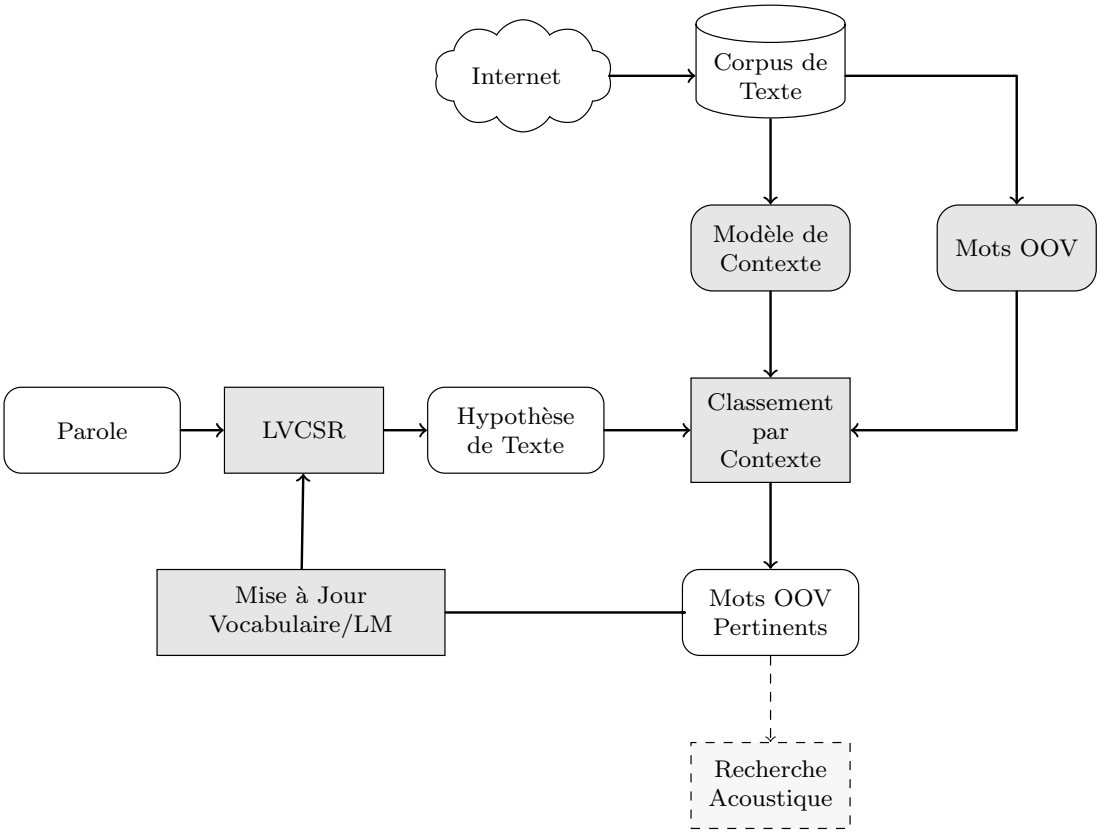


Figure 1: Méthodologie de reconnaissance des mots hors vocabulaire (OOV).

2.1 Travaux connexes

Dans le cadre de notre tâche de reconnaissance des documents diachroniques audio, la modélisation du contexte sémantiques/thématique est un problème difficile, parce que

- plusieurs nouveaux noms propres peuvent être présents dans une faible quantité de données ;
- le modèle de contexte doit être robuste aux erreurs de transcription du LVCSR.

[Senay et al., 2013] modélise les PN avec le modèle de type LDA. Une approche similaire basée sur la LSA a été expérimentée dans [Bigot et al., 2013]. Cependant, ces approches estiment un modèle pour chaque PN, ce qui les limite seulement aux PN très fréquents.

Nos représentations contextuelles discriminantes sont liées aux approches récentes de classification de textes en utilisant les réseaux de neurones. Dans le contexte de la classification de textes, plusieurs architectures fondées sur les réseaux de neurones ont été proposées : *Feedforward Neural Networks* [Iyyer et al., 2015, Nam et al., 2014], réseaux convolutifs (*Convolutional Neural Networks*, CNN) [Kim, 2014, Johnson and Zhang, 2015, Wang et al., 2015] et réseaux récurrents (*Recurrent Neural Networks*, RNN) [Socher et al., 2013, Hermann and Blunsom, 2013, Dong et al., 2014, Tai et al., 2015, Dai and Le, 2015]. Pour notre tâche, nous nous appuyons sur des méthodes de type sac-de-mots (*Bag-of-Words*) et construites au niveau de document. Nous avons choisi de travailler au niveau du document d'une part car cela permet d'être moins influencé par les erreurs des transcription du LVCSR et d'autre part car nous n'avons pas d'information sur la position des noms propres manquants. Par rapport aux autres travaux dans le domaine de la classification de textes, notre spécificité est que nous avons un grand nombre de classes de sorties (les OOV PN) et la distribution des documents par OOV PN est très asymétrique [Sheikh et al., 2015a].

2.2 Modèles de contexte sémantique et de thème

Les modèles de contexte sémantique ont une longue histoire dans le traitement automatique du langage naturel [Turney and Pantel, 2010]. Les modèles fondées sur la *Latent Dirichlet Allocation* (LDA) [Blei et al., 2003] ont été les méthodes les plus importantes pour représenter la distribution des thèmes des documents. La LDA permet de dériver des thèmes en utilisant une analyse bayésienne hiérarchique. La LDA est un modèle génératif. Il a été montré que la LDA surpasse la LSA pour la tache de classification des documents [Blei et al., 2003] et la tache de prédiction de mots [Griffiths et al., 2007].

Soit un ensemble de D documents composés d'un vocabulaire de V mots et (K) thèmes à modéliser, la distribution conjointe correspondante au processus génératif du modèle LDA est :

$$p(w, z, \theta, \phi|\alpha, \beta) = p(\phi|\beta)\ p(\theta|\alpha)\ p(z|\theta)\ p(w|\phi_z) \quad (1)$$

où z est le thème (caché) attribué au mot w dans un document d, $\theta = [\theta_{dk}]_{D\times K}$ est la distribution multinomiale de thèmes dans chaque document d. $\phi = [\phi_{vk}]_{V\times K}$ est la distribution multinomiale des mots dans un thème. α et β sont deux probabilité *a priori* (*Dirichlet priors*) pour θ et ϕ. Les paramètres du modèle LDA, θ et ϕ, et les thèmes attribués aux mots z peuvent être estimés en utilisant un algorithme d'échantillonnage de Gibbs [Griffiths and Steyvers, 2004].

Plus récemment, d'autres méthodes de représentations des mots (*word embeddings*) et des contextes sont devenus populaires. Ce sont des méthodes fondées

sur la prédiction du contexte dans lequel les mots apparaissent [Mikolov et al., 2013b, Pennington et al., 2014]. Ces représentations ont été efficacement appliquées à diverses tâches de traitement de textes [Baroni et al., 2014]. Les modèles de Mikolov et al. [Mikolov et al., 2013b, Mikolov et al., 2013a] sont devenus populaires en raison de leur capacité à gérer de grandes quantités de données non structurées avec un faible coût de calculs. Dans notre travail, nous utilisons le modèle *Skip-gram* de Mikolov et al. car les *Word Embeddings*, générés par ce modèle, sont plus performants que ceux du modèle CBOW. L'objectif du modèle Skip-gram consiste à maximiser la probabilité de prédire les mots qui apparaissent à proximité d'un mot donné. Soit $C(w)$ le contexte d'un mot w du corpus, la fonction objectif est[1] :

$$\arg\max_{\Theta} \prod_{w \in corpus} \left[\prod_{c \in C(w)} p(c|w;\Theta) \right] \tag{2}$$

où Θ sont les paramètres du modèle (poids de la couche d'entrée et de la couche de sortie).

2.3 Récupération des OOV PN en utilisant des représentations du contexte

Pour récupérer les OOV PN pertinents à un document audio, nous construisons un espace sémantique/thématique qui capture les relations entre les mots IVs et les OOV PN. Ensuite, la meilleure hypothèse du LVCSR est projetée dans l'espace de contexte pour déduire des OOV PN pertinents. Dans cette section, nous présentons les méthodes de récupération des OOV PN en utilisant les modèles LDA et Skip-gram.

2.3.1 Méthode I : Proximité entre l'hypothèse du LVCSR et les OOV PN dans l'espace contextuel

Pour récupérer les OOV PN, nous proposons d'utiliser la proximité entre l'hypothèse du LVCSR et des OOV PN dans l'espace contextuel. Un modèle de type LDA ou LSA peut etre appris sur les données diachroniques. Ce modèle représente la distribution des thèmes et prend en compte tous les OOV PN apparus dans les données diachroniques. Ce modèle peut également être utilisé

[1]Pour améliorer l'efficacité des calculs, une autre fonction avec le même objectif est possible [Goldberg and Levy, 2014].

pour inférer les thèmes à partir des transcriptions du LVCSR. Donc nous pouvons en déduire la proximité entre les OOV PN dans les hypothèses du LVCSR et attribuer un certain score à chaque mot OOV PN.

Utilisation des représentations LDA : Tout d'abord, le modèle LDA est construit en utilisant les documents textuels du corpus diachronique. Si nous notons l'hypothèse du LVCSR par h, la probabilité d'un OOV PN ($\tilde{v}_i$) est obtenu ainsi :

$$p(\tilde{v}_i|h) = \sum_{k=1}^{K} p(\tilde{v}_i|k)\, p(t|h) \tag{3}$$

Pour récupérer les OOV PN pertinents, nous calculons $p(\tilde{v}_i|h)$ pour chaque $\tilde{v}_i$, puis nous l'utilisons comme score de classement des OOV PN pertinents.

Utilisation des Word Embeddings : Le modèle Skip-gram ne permet pas d'apprendre une représentation par document mais seulement une représentation pour chaque mot. Pour représenter un document, nous proposons d'utiliser la propriété de linéarité des word embeddings : grâce au fait qu'il n'y pas de linéarité dans la première couche du modèle neuronal du modèle Skip-gram, il est valide d'additionner les représentations de différents mots.

Pendant l'apprentissage, un projection (Skip-gram word embedding) est apprise pour chaque mot présent dans le corpus diachronique. En utilisant les *word embeddings* et leur propriété de linéarité, la représentation d'un document est obtenue en faisant la moyenne des *embeddings* des tous les mots de ce document. Nous appelons cette représentation *AverageVec*. Cette représentation vectorielle de dimension K pour le document h est comparée avec l'*embedding* $\tilde{v}_i$ de chaque OOV PN. Ensuite, le score est calculé de la façon suivante :

$$s_i = \frac{\sum_{k=1}^{K} h_k\, \tilde{v}_{ik}}{\sqrt{\sum_{k=1}^{K}(h_k)^2}\, \sqrt{\sum_{k=1}^{K}(\tilde{v}_{ik})^2}} \tag{4}$$

Le score s_i est utilisé pour classer les OOV PN $\tilde{v}_i$.

2.3.2 Méthode II : Représentation spécifique au document

La méthode I est fondée sur les représentations de l'espace sémantique/thématique des noms propres OOV. Les points faibles de cette méthode sont les suivants :

- les noms propres OOV qui ont un petit nombre d'observations peuvent avoir des représentations non fiables ;

- si les noms propres OOV qui apparaissent dans des contextes trop variables, leurs représentations globales peuvent être sous-optimales.

Au lieu d'utiliser comme représentation d'un nom propre OOV estimé, nous pouvons exploiter des représentations estimées avec des documents dans lesquels ce OOV PN apparaît. La méthode proposée dans la suite utilise cette idée.

Pendant la phase d'apprentissage, les documents textuels diachroniques sont indexés avec des noms propres OOV qui s'y trouvent. Une représentation sémantique est apprise pour chaque document du corpus diachronique. Cette représentation est associée à chaque nom propre OOV de ce document. Un OOV PN qui figure dans plusieurs documents diachroniques aura de multiples représentations. Pendant la phase de test, la représentation vectorielle (T-dimensionnelle) de l'hypothèse du LVCSR h est comparée aux q vecteurs de contexte (C_q^i) de chaque OOV PN $\tilde{v}_i$ pour calculer un score, en utilisant la similitude cosinus :

$$\begin{aligned} s_i &= \max_q \{\text{Cosine\_Similarity}(h, C_q^i)\} \\ &= \max_q \left\{ \frac{h.C_q^i}{\|h\| \left\|C_q^i\right\|} \right\} \\ &= \max_q \left\{ \frac{\sum_{k=1}^{T} h_k \, C_{qk}^i}{\sqrt{\sum_{k=1}^{T}(h_k)^2} \sqrt{\sum_{k=1}^{T}(C_{qk}^i)^2}} \right\} \end{aligned} \tag{5}$$

où s_i est le score de $\tilde{v}_i$ qui sera utilisé pour classer les OOV PN pertinents pour h.

3 Représentations discriminantes

Dans cette section, nous présentons des modèles discriminants. Dans ces modèles, le critère d'apprentissage que nous voulons maximiser est directement la probabilité des OOV PN en fonction des mots présents dans un document.

3.1 Modèle neuronal sac-de-mots

Le modèle neuronal sac-de-mot (*Neural Bag-of-Words* NBOW) [Kalchbrenner et al., 2014, Iyyer et al., 2015] prend en entrée un texte X contenant un ensemble de mots w et génére des estimations de probabilité pour les L classes de sortie. Le réseau compte un couche d'entrée de taille V (taille du vocabulaire) ; une couche

cachée de taille K ; une couche de sortie de taille L correspondant au nombre de mots OOV PN. La figure 2 illustre le modèle NBOW. Les paramètres devant être estimés sont la matrice W^I de taille $[V \times K]$, la matrice W^O de taille $[K \times L]$ et b est un vecteur de biais. En entrée, on utilise une représentation sac-de-mot : si le mot i est présent dans le document analysé alors le neurone d'entrée i est mise à 1, sinon il est mis à zéro. Les valeurs de la couche cachée sont obtenues par le produit matriciel de W^I et de la couche d'entrée. En pratique, la moyenne des vecteurs de mots est utilisée :

$$z = \frac{1}{|X|} \sum_{w \in X} v_w \tag{6}$$

Cette moyenne des vecteurs est utilisée pour estimer les probabilités de sortie : $\hat{y} = \text{softmax}(zW^O + b)$, où W^O est une $[K \times L]$ matrice et b est un vecteur de biais, et $\text{softmax}(l) = \exp(l)/\sum_{j=1}^{L} \exp(l_j)$.

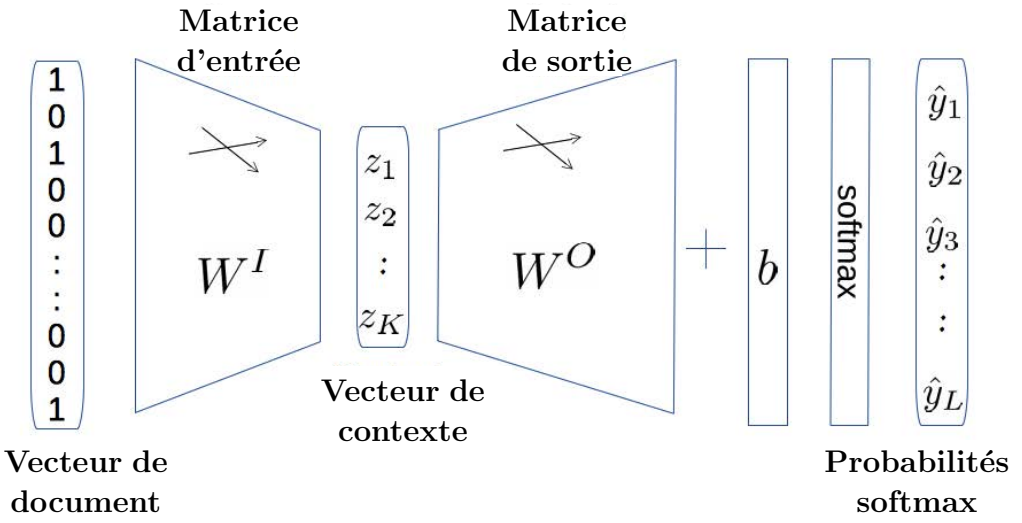


Figure 2: Modèle neural bag-of-words (NBOW).

La moyenne des vecteurs $z \equiv \{z_1, z_2 ... z_K\}$ représente le vecteur de contexte pour ce document. Le produit entre ce vecteur contexte et la matrice de sortie (W^O) est équivalent à la comparaison du document d'entrée et des OOV PN dans l'espace de contexte.

Pendant l'apprentissage du modèle, les mots d'un document du corpus diachronique sont fournis à l'entrée du modèle et un neurone de sortie correspondant à un OOV PN de ce document est mis à 1. Le critère d'apprentissage choisi est la minimisation de l'entropie croisée [Goldberg, 2015]. Si un document contient plus

d'un OOV PN, ce document d'apprentissage est dupliqué pour chaque OOV PN. Pendant le test, pour récupérer des OOV PN pertinents, les mots de la meilleure hypothèse du LVCSR sont donnés à l'entrée du système ; les probabilités softmax de sortie sont utilisées en tant que scores pour classer les OOV PN.

3.2 Modèle Neural Bag-of-Weighted-Words (NBOW2)

Nous pensons que le modèle NBOW présenté précédemment ne parvient pas à utiliser *explicitement* le fait que certains mots peuvent être plus importants que d'autres. En conséquence, nous proposons un nouveau modèle, appelé NBOW2. Ce modèle permet d'apprendre l'importance de certains mots en leurs associant des poids. Pour apprendre ces poids d'importance, une somme pondérée des mots d'entrée X est utilisé :

$$z = \frac{1}{|X|} \sum_{w \in X} \alpha_w \; v_w \tag{7}$$

où α_w sont des poids d'importance pour chaque mot $w \in X$. Les poids α_w sont obtenus en introduisant un nouveau vecteur a de dimension K, défini de la façon suivante :

$$\alpha_w = f(v_w \cdot a) \tag{8}$$

où $(\cdot)$ représente le produit scalaire. La fonction f permet de projeter les poids d'importance dans l'intervalle $[0, 1]$. Les poids d'importance α_w représentent une distance entre w et a dans l'espace de contexte. Cela permet de garantir que le calcul des α_w tient compte des similitudes de mots contextuels. Pour f, différentes fonctions d'activation peuvent être utilisées, par exemple, sigmoide, softmax (comme dans [Sheikh et al., 2015c]) ainsi que la tangente hyperbolique. Selon nos expériences, la fonction sigmoide $f(x) = (1 + e^{-x})^{-1}$ est un meilleur choix en termes de convergence et de la précision. Nous discutons plus en details le choix de f dans la Section 5.

La figure 3 représente le modèle NBOW2. Les entrées, la matrice W^I et la matrice W^O sont similaires à celles de NBOW. En revanche, le vecteur de contexte du document est obtenu différemment. Un produit scalaire est calculé entre chaque vecteur de mot d'entrée et le vecteur a. Les sorties du produit scalaire sont modifiées par la fonction f. Les poids d'importance des mots sont multipliés par les vecteurs des mots du document d'entrée, et on obtient la somme pondérée représentant le vecteur de contexte du document.

Le travail de Ling [Ling et al., 2015] est proche de notre proposition consistant à utiliser des poids des mots en s'appuyant sur un réseau de neurones. Cependant, pour améliorer les vecteurs de contexte, les auteurs utilisent des poids en fonction des positions des mots dans le document *Continuous Bag-Of-Words*

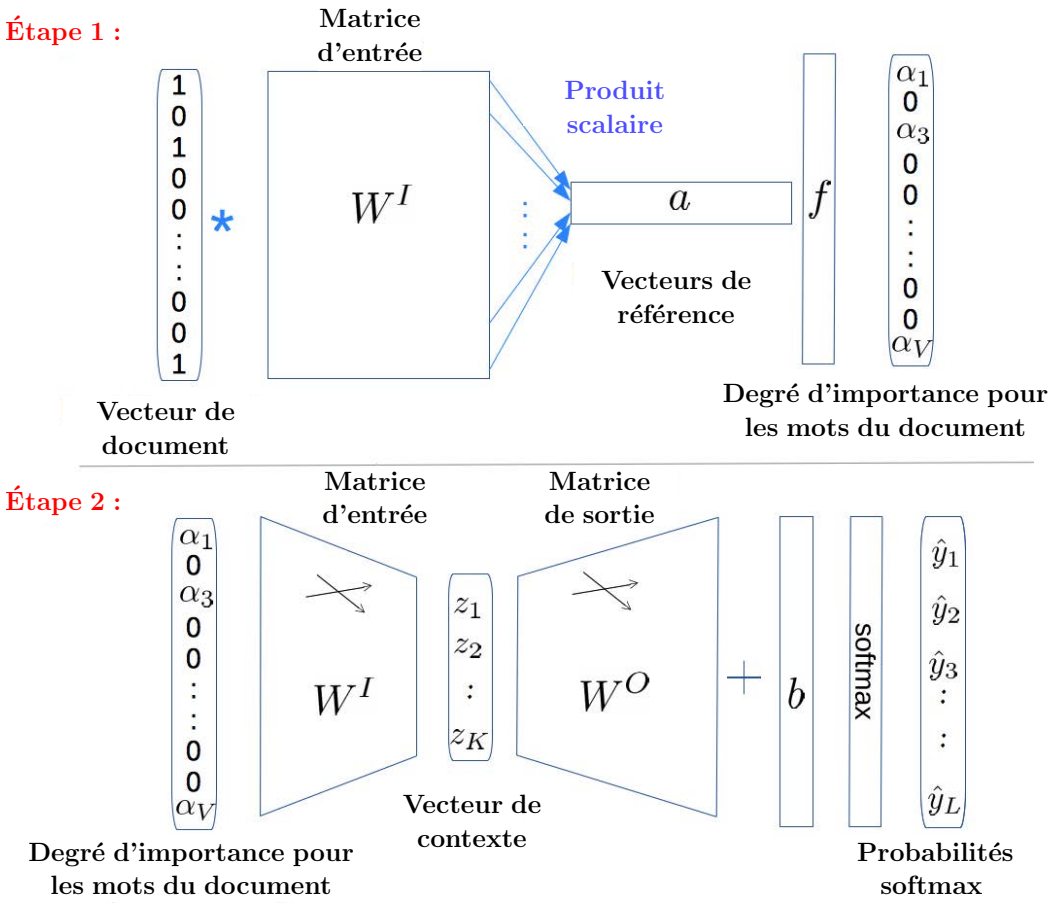


Figure 3: Modèle neural bag-of-weighted-words (NBOW2).

(CBOW) [Mikolov et al., 2013a]. Notre modèle NBOW2 permet d'apprendre un vecteur contextuel et d'attribuer des poids d'importance à des mots spécifiques à la tâche étudiée. Quelques travaux similaires sont développés dans le domaine de la traduction automatique [Bahdanau et al., 2014] de la reconnaissance de la parole [Chan et al., 2015], de sous-titrage des images [Xu et al., 2015] et de l'analyse des séquences protéiques [Sønderby et al., 2015].

3.3 Combinaison des Modèles NBOW et NBOW2

Nous proposons un nouveau modèle, appelé NBOW2+, dans lequel, pour un document donné, les vecteurs de contexte sont concaténés. NBOW2+ possède deux matrices d'entrée (W_1^I, W_2^I) et utilise deux vecteurs v_w^1 et v_w^2, de K dimensions pour chaque mot d'entrée w. Ce modèle a un vecteur a de K dimensions, simi-

laire à NBOW2, une matrice W^O et un vecteur de biais b. Le vecteur de contexte z du document est obtenu par la concaténation des deux vecteurs de contexte z_1 et z_2 comme suit :

$$z_1 = \frac{1}{|X|} \sum_{w \in X} v_w^1 \quad , \quad z_2 = \frac{1}{|X|} \sum_{w \in X} \alpha_w \, v_w^2$$
$$z = [z_1 z_2] \tag{9}$$

Le vecteur de contexte de document est la concaténation des deux vecteurs de contexte de dimension K chacun. Donc les paramètres de la couche de sortie (W^O, b) ont une dimension $2K$. La procédure d'apprentissage et la fonction à optimiser sont les mêmes que dans le cas de NBOW et de NBOW2.

4 Protocole expérimental

4.1 Corpus

Le Table 1 présente trois ensembles de données d'actualité réalistes qui sont utilisés pour l'apprentissage, la validation et le test dans notre étude. Ces ensembles de données justifient notre motivation pour la traitement des OOV PN. Ces ensembles de données ont été recueillis à partir de deux sources différentes : (a) le journal français *L'Express*[2] et (b) le site français[3] de la chaîne de télévision *Euronews*. Le corpus *L'Express* contient des articles de journaux alors que le corpus Euronews contient des articles ainsi que des vidéos de bulletins d'information et leurs transcriptions textuelles. Dans notre étude, les données de *L'Express* seront utilisées comme corpus diachronique pour estimer des modèles de contexte ou de thème et pour choisir les OOV PN pertinents pour les vidéos d'Euronews (le corpus de test). Les documents textuels d'*Euronews*, désigné par 'valid' dans Table 1, seront utilisés comme un ensemble de validation dans nos expériences.

4.1.1 Prétraitement du corpus diachronique pour former des modèles de contexte

Le logiciel TreeTagger [Schmid, 1994b] est utilisé pour détecter automatiquement les PN dans le texte. Les mots et PN qui apparaissent dans le lexique de notre système de LVCSR sont désignés par 'IV' (dans le vocabulaire) et les mots et PN restants sont étiquetés comme 'OOV' (hors vocabulaire). Pour la construction

[2]http://www.lexpress.fr/
[3]http://fr.euronews.com/

Table 1: Ensembles de données d'actualité

	L'Express (train)	*Euronews* (valid)	*Euronews* (test)
Type de Documents	Text	Text	Video
Période	Janvier - juin 2014		
Nombre de documents[1]	45K	3.1K	3K
Taille du vocabulaire[2]	150K	42K	45K
Taille du Corpus (nombre de mots)	24M	550K	700K
PN unigrammes[2]	57K	12K	11K
Nombre des PN	1.45M	54K	42K
OOV unigrammes[3]	12.4K	4.9K	4.3K
Documents avec OOV[3]	32.3K	2.25K	2.2K
Nombre des OOV[3]	141K	9.1K	8K
OOV PN unigrammes[3]	9.3K	3.4K	3.1K
Documents avec OOV PN[3]	26.5K	1.9K	1.9K
Nombre des OOV PN[3]	107K	6.9K	6.2K

[1]K désigne *Mille* et M désigne *Million*

[2] Les unigrammes de *L'Express* qui apparaissent moins de 3 fois sont ignorés

[3] Les unigrammes de *L'Express* qui apparaissent dans moins de 3 documents sont ignorés ; documents avec plus de 20 et moins de 500 mots

des modèles de contexte, les mots du corpus diachronique sont lemmatisés et filtrés (suppression des PN et des mots non PN apparaissant moins de 3 fois). De plus, les mots outils sont supprimés et seuls les mots des classes grammaticales nom propre, nom, adjectif, verbe et acronyme sont conservés. Les modèles de contexte et de thème sont construits à partir de ce vocabulaire filtré.

4.1.2 Statistiques des OOV PN

Comme indiqué dans la Table 1, 72% (3.1K sur 4.3K) des OOV dans le corpus Euronews video sont PN, et 64% (1.9K sur 3K) des vidéos contiennent des OV PN. Le nombre total de OOV PN à retrouver pour le corpus *Euronews* vidéos (obtenu en comptant les OOV PN uniques par vidéo), est 4694. Sur ces 4694 OOV PN, seulement 2010 (c'est-à-dire 42%) peuvent être récupérés avec le corpus

diachronique*L'Express*. La couverture des OOV PN pourrait être augmentée en ajoutant des documents textuels extraits de sites d'informations supplémentaires comme indiqué dans [Sheikh et al., 2016a].

4.2 Systèmes de reconnaissance de la parole

Dans nos expériences, nous utilisons deux systèmes de LVCSR. Notre système de LVCSR basé sur des GMM-HMM a un WER plus élevé que celui fondé sur des DNN-HMM. Ces deux systèmes de LVCSR sont utilisés pour démontrer l'effet des erreurs de reconnaissance sur les performances des approches proposées. Ces systèmes effectuent une segmentation et une transcription automatique de la parole de fichiers audio de bulletins d'information.

4.2.1 Logiciel ANTS (Automatic News Transcription System)

Le système ANTS [Illina et al., 2004] est fondé sur des modèles GMM-HMM appris sur 200 heures de fichiers audio de bulletins d'information. Il utilise le moteur de reconnaissance Julius [Lee and Kawahara, 2009]. Le lexique a été sélectionnée à partir des mots les plus fréquents dans des documents textuels antérieur à 2009. Il contient 122K mots (260K prononciations). En utilisant le SRILM toolkit [Stolcke, 2002], un modèle de langage 4-grammes est estimé sur des corpus de texte d'environ 1800 millions de mots. Les transcriptions automatiques obtenues par ANTS pour les vidéos de *Euronews* ont un WER de 41.7%.

4.2.2 Logiciel KATS (Kaldi Automatic Transcription System)

Le système KATS est basé sur des modèles DNN-HMM appris sur les mêmes corpus que ANTS. Il utilise le moteur de reconnaissance de la parole Kaldi [Povey et al., 2011]. Le lexique est identique à celui de ANTS. Un modèle de langage bigramme est estimé sur le même corpus de textes que celui utilisé pour construire le modèle de langage de ANTS. Les transcriptions automatiques obtenues par KATS ont un WER de 16.4%, sur les vidéos de *Euronews*.

4.3 Modèle de référence : Pointwise Mutual Information

La *pointwise mutual information* (PMI) est utilisé comme une mesure de corrélation statistiques dans la théorie de l'information. En linguistique informatique (Computational linguistics), PMI a été utilisé pour trouver des corrélations et des associations entre les mots [Church and Hanks, 1990]. Nous l'utilisons pour

mesurer les associations entre OOV PN et des mots de vocabulaire LVCSR. Désignant v_x et v_y deux mots apparaissant dans un document, les PMI est calculée ainsi :

$$pmi(v_x, v_y) = \log \frac{p(v_x, v_y)}{p(v_x)p(v_y)} \tag{10}$$

où $p(v_x, v_y)$ désigne la probabilité de co-occurrence des termes v_x et v_y dans un document, $p(v_x)$ et $p(v_y)$ désignent les probabilités d'occurrence des termes v_x et v_y dans tout du corpus. Pour une hypothèse de reconnaissance h contenant des mots $\{w_1, w_2, w_3, ...\}$, le score de chaque OOV PN $\tilde{v}_i$ est calculé de la façon suivante :

$$s(\tilde{v}_i) = \sum_{i=1}^{|h|} \log \frac{p(\tilde{v}_i, w_i)}{p(\tilde{v}_i)\ p(w_i)} \tag{11}$$

Cette méthode ne modélise pas explicitement des informations sémantiques ou thèmatiques et elle sera utilisée comme notre méthode de référence (baseline).

4.4 Mesures de performance pour la récupération des OOV PN

Pour mesurer la performance de récupération des OOV PN pertinents pour un document audio, nous utilisons des mesures fondées sur le rappel et la précision. Spécifiquement, nous utilisons le *rappel* et la *MAP* (*Mean Average Precision* : moyenne de la précision moyenne) [Manning et al., 2008a] qui sont couramment utilisés pour évaluer les systèmes de recherche d'information. Pour un document audio, de nombreux OOV PN peuvent être pertinents. Nous appelons 'target OOV PN' les OOV PN effectivement présents dans un document. Pour notre tâche, nous calculons le rappel (R) :

$$R = \frac{\text{nombre des targets OOV PN récupérés}}{\text{nombre total des targets OOV PN}}$$

Le calcul des MAP pour un ensemble de documents Q se fait ainsi :

$$MAP = \frac{\sum_{q=1}^{Q} \overline{P}(q)}{Q} \tag{12}$$

où $\overline{P}(q)$ est le score moyen de précision pour chaque document q. Avec la liste classée des OOV PN pour un document, $\overline{P}(q)$ est calculé par :

$$\overline{P}(q) = \frac{\sum_r P(r)\ rel(r)}{\text{nombre des targets OOV PN dans } q} \tag{13}$$

où $P(r)$ est la précision au rang r, calculée comme

$$P@r = \frac{\text{nombre des targets OOV PN récupérées jusqu'à } r}{r}$$

$rel(r)$ est une fonction indicatrice égale à 1 si le OOV PN au rang r est une target OOV PN et égale à 0 sinon.

Les courbes de rappel et de MAP permettent des interprétations différentes des résultats. Après la récupération des OOV PN pertinents, les N premiers OOV PN pertinents sont sélectionnés. Après avoir ajouté ces OOV PN au vocabulaire du système de reconnaissance, une deuxième passe de reconnaissance ([Fohr and Illina, 2015, Oger et al., 2008b]) ou une recherche de mots clés ([Parada et al., 2010b]) peut être effectuée. Alors que la valeur MAP tient compte des rangs des OOV récupérés, la valeur de rappel pour un *point de fonctionnement* (N choisi n'est pas sensible au rang des OOV choisis ; par exemple, dans les expériences de la figure 4) pour $N = 465$, toutes les méthodes indiquent le même rappel, mais les MAP sont différentes.

Pour une analyse détaillée, les résultats de récupération seront présentés sous forme de graphique de rappel et MAP (Figure 4). En calculant de la MAP less target OOV PN qui ne sont pas dans le liste des top-N meilleures OOV PN obtenir un score de précision ($P(r)$) de zéro. Pour permettre de comparer tous les modèles, on utilisera la valeur MAP calculée au point de fonctionnement 128 (MAP@128).

Pour déterminer si la différence entre les valeurs MAP@128 obtenues par les deux méthodes est statistiquement significative, nous utilisons un test de Student mesurée à l'aide du Student's paired t-test ou un ré-échantillonnage [Smucker et al., 2007]. L'*hypothèse nulle* est qu'il n'y a pas de différence entre les deux méthodes, et qu'elles produisent des résultats identiques. L'hypothèse nulle est rejetée si la valeur de p est inférieure à 0.05 pour les deux tests [Smucker et al., 2007]. Pour le ré-échantillonage, nous générons 100,000 permutations aléatoires des résultats des deux méthodes à comparer.

4.5 Sélection des hyper-paramètres des modèles

Le modèle LDA a trois hyper-paramètres, α : Dirichlet prior pour les distributions documents-thèmes, β : Dirichlet prior pour les distributions thèmes-mots et T : le nombre de thèmes, qui est aussi la dimension du vecteur représentant un mots ou un document. Il y a des travaux dans la littérature [Griffiths and Steyvers, 2004, Wallach et al., 2009] qui discutent de la sélection des hyper-paramètres de la LDA et ils sont généralement basés sur la probabilité obtenue par le modèle

sur un ensemble de données. Dans notre tâche, nous choisissons des probabilités symétriques pour les priors et nous sélectionnons les hyper-paramètres qui donnent les meilleurs résultats sur les données de validation au point de fonctionnement 128 mots (MPA@128).

En plus de la taille du word embedding, le modèle Skip-gram a un hyper-paramètre crucial : la taille de la fenêtre contextuelle. Nous avons essayé jusqu'à la taille de fenêtre de 40 (longueur du plus petit des documents dans nos données).

D'après les résultats obtenus sur les données de validation, nous avons choisi 400 pour le nombre de thèmes de la LDA, 400 pour la taille du word embedding du modèle Skip-gram, 20 pour la taille de fenêtre de contexte du modèle Skip-gram, $\alpha = 0.01$ et $\beta = 0.01$ pour la LDA.

Les modèles NBOW, NBOW2 et NBOW2+ avec des word embeddings de taille 400 ont également donné la meilleure performance sur le corpus de validation. En plus de la taille de word embeddings, il y a d'autres hyper-paramètres importants à choisir pour ces modèles. Ceux-ci seront discutés en détail dans les sections 4.6 et 5.2.

4.6 Apprentissage des Modèles NBOW, NBOW2, NBOW2+

Dans cette section, nous discutons des choix faits pour la construction et l'apprentissage des modèles NBOW, NBOW2 and NBOW2+. Certains hyper-paramètres jouent un role crucial qui affectent significativement les performances des modèles. Une discussion plus spécifique est présentée dans la section 5.2.

4.6.1 Initialisation

Il est bien connu qu'une bonne initialisation des poids du réseau est cruciale pour l'apprentissage des réseaux de neurones profonds [Larochelle et al., 2009, Goldberg, 2015]. Bien que le modèle NBOW n'est un modèle profond, nous avons examiné si l'initialisation affecte la performance du modèle NBOW dans notre tâche. Nous présenterons les résultats pour le modèle NBOW avec des vecteurs de mots d'entrée initialisée (a) de manière aléatoire et (b) avec des word embeddings issus du modèle Skip-gram. Les vecteurs correspondant aux OOV PN sont initialisés de manière aléatoire.

4.6.2 Apprentissage en une passe et apprentissage en deux passes

Nous explorons deux méthodes d'apprentissage des modèles NBOW, NBOW2 et NBOW2+ : (a) *Apprentissage en une passe* et (b) *Apprentissage en deux passes*. Dans l'apprentissage en une passe, tous les paramètres du réseau sont appris et mis à jour en même temps. Le modèle NBOW (Section 3.1) comporte une matrice d'entrée, une matrice de sortie et un vecteur de biais de sortie. Les modèles NBOW2 et NBOW2+ (section 3.2 et 3.3) comportent en plus un vecteur a.

La méthode à deux phases comporte une première phase d'apprentissage dans laquelle seuls les paramètres de sortie (W^O, b), et le vecteur a pour les modèles NBOW2 et NBOW2+, sont mis à jour en gardant les vecteurs d'entrée fixes à leurs valeurs d'initialisation (word embeddings calculés par le modèle Skip-gram). Dans la deuxième phase, tous les paramètres du modèle, y compris les vecteurs d'entrée, sont mis à jour. La motivation de cet apprentissage en deux phases est l'espoir d'obtenir une meilleure convergence. La première phase est utilisée pour mieux apprendre les paramètres de sortie (qui ont été initialisés au hasard).

4.6.3 Taux d'apprentissage et critère d'arret

Tous les modèles NBOW sont appris en utilisant l'algorithme de descente de gradient avec ADADELTA [Zeiler, 2012]. ADADELTA fournit un taux d'apprentissage (*learning rate*) adaptatif et il est robuste aux fluctuations du gradient. Nous avons testé deux valeurs pour la paramètre (ρ) d'ADADELTA, 0.99 et 0.95. Nous avons choisi ρ =0.99 dans toutes nos expériences, parce qu'il donne de bons résultats sur les données de validation.

Pour contrôler l'apprentissage des modèles NBOW, nous avons choisi un critère d'arrêt [Bengio, 2012] basé sur l'erreur sur les données de validation : *Early stopping*. Le critère d'early stopping est utilisé pendant l'apprentissage en une passe ainsi que lors des deux passes de l'apprentissage en deux passes.

4.6.4 Dropout

Le technique de *dropout* [Srivastava et al., 2014] réduit le sur-apprentissage et donne des améliorations par rapport aux autres méthodes de régularisation pour les réseaux neuronaux profonds. Les modèles NBOW, NBOW2 et NBOW2+ ne sont pas des architectures profondes, mais nous sommes intéressés à analyser si le mécanisme de dropout pouvait éviter le sur-apprentissage et ajouter de la robustesse aux modèles BOW. Nous avons appliqué le dropout à la couche

d'entrée pour simuler des erreurs d'omissions d'un système de reconnaissance. Cette méthode dropout a été récemment essayée et a donné des améliorations dans les tâches de classification de texte [Dai and Le, 2015, Iyyer et al., 2015].

5 Résultats des expériences et Discussion

5.1 Performance de récupération des OOV PN

Le Figure 4 illustre les performances (rappel et MAP) de récupération des OOV PN pour les différents modèles. Les résultats sur des transcriptions de référence sont présentés à gauche, ceux sur les transcriptions de ANTS LVCSR au centre et ceux sur les transcriptions de KATS LVCSR à droite. Ce triptyque permet de montrer l'effet des erreurs de reconnaissance sur la récupération des OOV PN. Les axes horizontaux représentent le nombre des OOV PN choisis dans le corpus diachronique. L'axe vertical représente le rappel (en haut) et la MAP (en bas) pour les target OOV PN. Pour chacune des méthodes, les modèles qui donnent la meilleure performance sur les données de validation ont été choisis (Section 4.5 et 5.2).

En comparant les performances de la Figure 4, nous pouvons faire les observations suivantes :

- La méthode de PMI, qui ne modélise pas explicitement le contexte sémantique, obtient la plus mauvaise performance. Les méthodes basées sur des modèles de contexte sémantique et le thème donnent de bien meilleurs résultats.
- Les méthodes basées sur LSA et Skip-gram donnent de meilleures performances que la LDA. En revanche, la LDA est plus robuste aux erreurs de reconnaissance.
- Les représentations spécifiques au document donnent de meilleurs rappel et MAP que les représentations globales apprises par chacun des modèles.
- Les modèles NBOW, NBOW2 et NBOW2+ montrent des résultats similaires (leurs graphiques se superposent). Ils obtiennent la meilleure performance de récupération des OOV PN et ils sont robustes aux erreurs de reconnaissance.

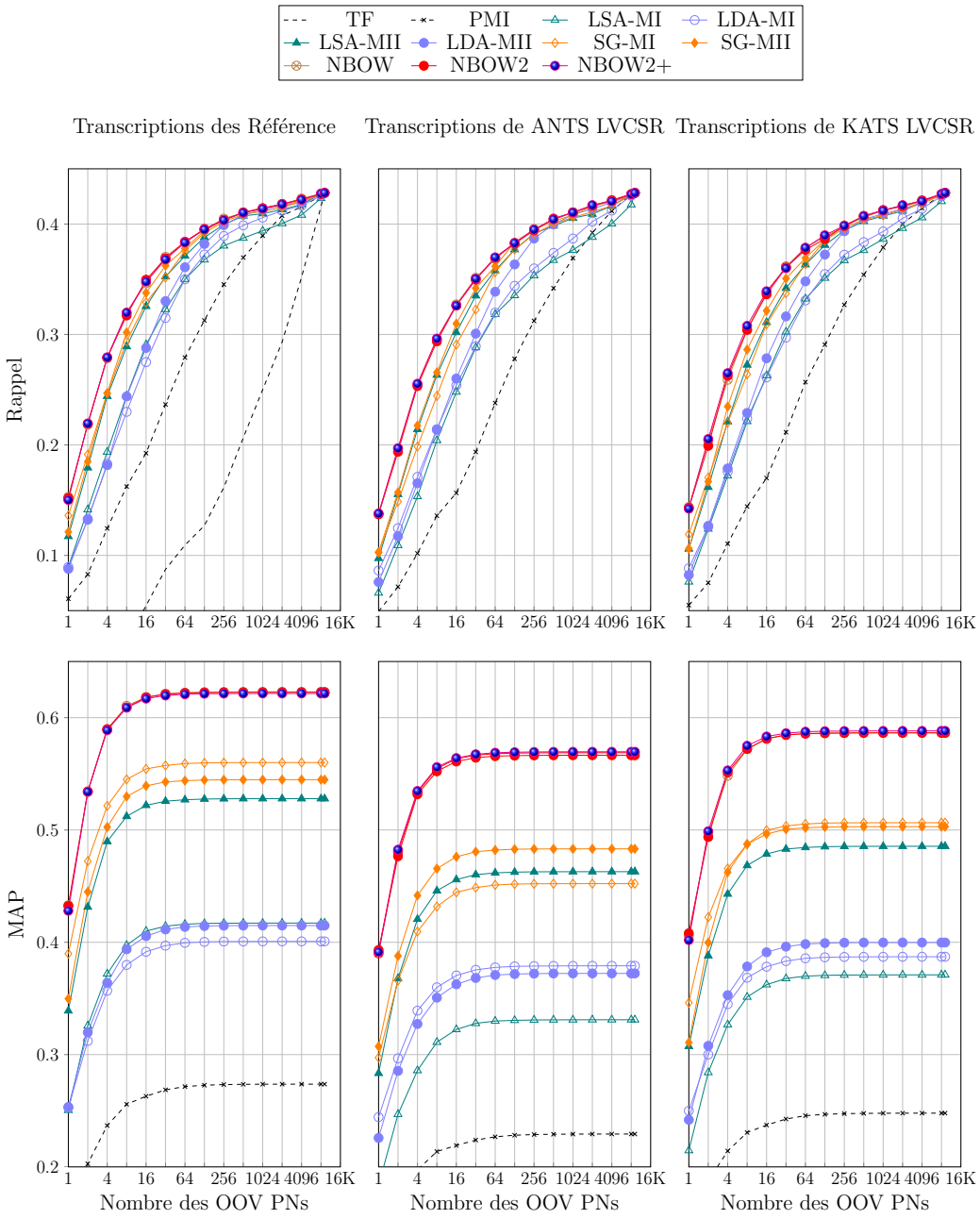


Figure 4: Rappel et moyenne de la précision moyenne (Mean Average Precision, MAP) de récupération des OOV PN pour l'ensemble de données audio *Euronews*. La *Term Frequency* (TF, fréquence d'un terme) est une sélection des k plus fréquents OOV PN. Les modèles NBOW, NBOW2 et NBOW2+ sont initialisés avec les word embeddings du Skip-gram et sont appris en deux phases.

5.2 Analyse de l'apprentissage des modèles NBOW, NBOW2 et NBOW2+

Dans cette section, nous analysons d'abord comment (a) l'utilisation du dropout et (b) l'apprentissage en deux phases affectent les performances du modèle NBOW.

5.2.1 Robustesse avec dropout

L'effet de l'application du dropout peut être observé sur la Table 2. Il est clair que le dropout améliore les résultats MAP. Nous pouvons observer que le modèle NBOW initialisé avec les embeddings du Skip-gram (Sg-1p) converge plus vite et donne de meilleures performances que le modèle NBOW avec initialisation aléatoire (Rand-1p). Mais l'utilisation du dropout donne plus d'amélioration pour Rand-1p que pour Sg-1p. Par exemple, la valeur de MAP pour la transcriptions de référence (MAP-TR) est améliorée de 15% pour Rand-1p et de 6.75% pour Sg-1p avec un dropout de 0.9, par rapport aux résultats obtenus sans dropout. Deuxièmement, nous pouvons observer que l'amélioration de la MAP avec le dropout est plus grand pour les transcriptions LVCSR. Par exemple, si l'on compare la valeur MAP pour la référence manuelle et pour les transcriptions de ANTS (MAP-TR et MAP-TA), les améliorations sont de 15% versus 25% pour Rand-1p, de 6.75% versus 11.8% pour Sg-1p et de 3.3% versus 8.2% pour Sg-2p.

5.2.2 Apprentissage en deux phases et améliorations avec le modèle NBOW2+

Dans la section 4.6.2 nous avons proposé d'apprendre les modèles NBOW en deux phases. Les résultats du MAP dans la Table 2 montrent que le meilleur résultat est obtenu avec l'apprentissage en deux phases. Cependant, il faut un plus grand nombre d'époques d'apprentissage comparé à l'apprentissage en une seule phase (Sg-1p). Avec l'aide de la figure 5, nous illustrons que ce problème est résolu par le modèle NBOW2+. Cette figure montre les évaluations des erreurs sur les données de validation pour les modèles NBOW, NBOW2 et NBOW2+ lors de l'apprentissage. On peut observer que les trois modèles (NBOW, NBOW2 et NBOW2+) convergent vers un même point, mais pas à la même vitesse : le modèle NBOW2+ obtient une convergence plus rapide sans compromis sur le taux d'erreur. Pour appuyer cet argument, nous présentons la table 3 qui compare le MAP atteint par les modèles NBOW, NBOW2 et NBOW2+ (dimension des vecteurs des mots de 400 et dropout de 0,9).

A partir de ces expériences, on peut conclure que (a) deux phases d'apprentissage conduisent à de meilleures performances avec les modèles NBOW et NBOW2

Table 2: MAP@128 obtenu par le modèle NBOW (vecteurs des mots de dimension 400) en utilisant le critère d'arrêt early stopping. les Suffixes V, TR, TA et TK désignent respectivement les performances sur les données de Validation, sur les Transcriptions de Référence, sur les Transcriptions de ANTS et KATS. Rand (resp. Sg) désigne l'initialisation avec des vecteurs aléatoires (resp. embeddings de Skip-gram). 1p et 2p désignent le nombre de passes d'apprentissage. La meilleure configuration est mise en évidence en caractères gras. * dénote une différence statistiquement non-significative par rapport à la meilleure configuration.

		probabilité dropout(p)				
		0.0	0.25	0.5	0.75	0.9
Rand-1p	époques	175	217	249	320	276
	MAP-V	0.458	0.482	0.502	0.537	0.530
	MAP-TR	0.500	0.522	0.549	0.578	0.576
	MAP-TA	0.419	0.435	0.464	0.505	0.526
	MAP-TK	0.457	0.473	0.500	0.533	0.542
Sg-1p	époques	112	147	152	149	155
	MAP-V	0.511	0.522	0.535	0.541	0.543
	MAP-TR	0.563	0.569	0.576	0.587	0.601
	MAP-TA	0.491	0.483	0.502	0.531	0.549
	MAP-TK	0.523	0.522	0.532	0.551	0.566
Sg-2p	époques	481	482	398	417	410
	MAP-V	0.551	0.553	0.562	0.574	**0.585**
	MAP-TR	0.602	0.598	0.605	0.615*	**0.622**
	MAP-TA	0.525	0.519	0.533	0.561*	**0.568**
	MAP-TK	0.555	0.552	0.561	0.578*	**0.586**

mais cela nécessite un apprentissage plus long, et (b) le modèle NBOW2+ permet de réduire considérablement ce temps d'apprentissage, sans compromis sur la performance MAP.

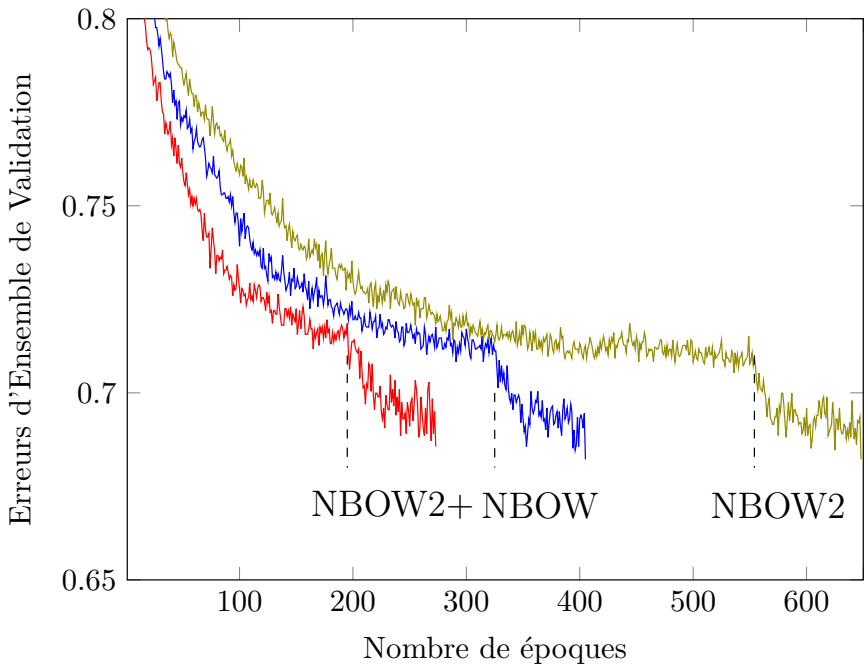


Figure 5: Erreurs sur le corpus de validation, au cours de l'apprentissage en deux phases des modèles NBOW, NBOW2 et NBOW2+. (Les pointillés indiquent la fin de la première phase d'apprentissage.)

5.3 Importance des mots appris par le modèle NBOW2

Nous présentons la figure 6 pour discuter (a) de l'importance des poids α_w appris par le modèle NBOW2, et (b) du choix de la fonction f pour le modèle NBOW2 (équation (8)). Il montre le degré d'importance des mots dans un document. Dans la figure 6, le graphique de gauche montre les poids attribués par le modèle NBOW2 avec une activation sigmoide et le graphique de droite montre les poids attribués par une activation softmax.

Tout d'abord, il est clair à partir de ces graphiques que le modèle NBOW2 apprend et affecte différents degrés d'importance pour les différents mots. Par exemple, ce document de test a pour sujet l'accident du pilote de Formule 1 Michael Schumacher et il a un OOV PN manquant *'Kehm'* (*Sabine Kehm* est la porte-parole de Michael Schumacher). Si nous analysons la liste des mots selon le graphique de gauche, les quatre mots les plus importants sont *michael*, *formule*, *critique* et *hospitaliser* et les quatre mots les moins importants sont *rester*, *tenir*, *monde* et *présent*. Dans cet exemple, il est évident que le modèle NBOW2 attribue un poids plus élevé à des mots qui sont importants pour la récupération des OOV PN. Cela est également vrai pour le modèle NBOW2 avec softmax. La

Table 3: Résulatts MAP@128 obtenus par les modèles NBOW, NBOW2 et NBOW2+ (dimension des vecteurs de mots 400, dropout de $p = 0{,}9$ et early stopping). V :données de validation, TR :transcriptions de référence, TA : transcriptions ANTS, TK : transcriptions KATS, Rand : initialisation aléatoire, Sg : initialisation avec les embeddings du Skip-gram, 1p : une seule passe, 2p :deux passes. La meilleure configuration est mise en évidence en caractères gras. * dénote une différence statistiquement non-significative par rapport à la meilleure configuration.

		NBOW	NBOW2	NBOW2+
R-1p	époques	276	123	210
	MAP-V	0.530	0.474	0.519
	MAP-TR	0.576	0.507	0.574
	MAP-TA	0.526	0.402	0.526
	MAP-TK	0.542	0.440	0.546
Sg-1p	époques	155	166	161
	MAP-V	0.543	0.541	0.547
	MAP-TR	0.601	0.599	0.601
	MAP-TA	0.549	0.549	0.545
	MAP-TK	0.566	0.566	0.566
Sg-2p	époques	410	648	273
	MAP-V	0.585*	0.587*	**0.593**
	MAP-TR	0.622*	0.622*	**0.621**
	MAP-TA	0.568*	0.566*	**0.569**
	MAP-TK	0.586*	0.586*	**0.588**

deuxième observation est que le modèle NBOW2 avec softmax tend à attribuer un poids plus élevé à moins de mots et un poids proche de zéro à la plupart des autres mots. Nous faisons l'hypothèse que le modèle NBOW2 avec softmax ignore (c'est-à-dire donne une valeur d'importance faible) trop de mots, ce qui affecte sa capacité discriminative, en particulier quand l'hypothèse de reconnaissance contient des mots erronés, ou que le document est multi-thématique, par exemple un mélange de sport et de politique.

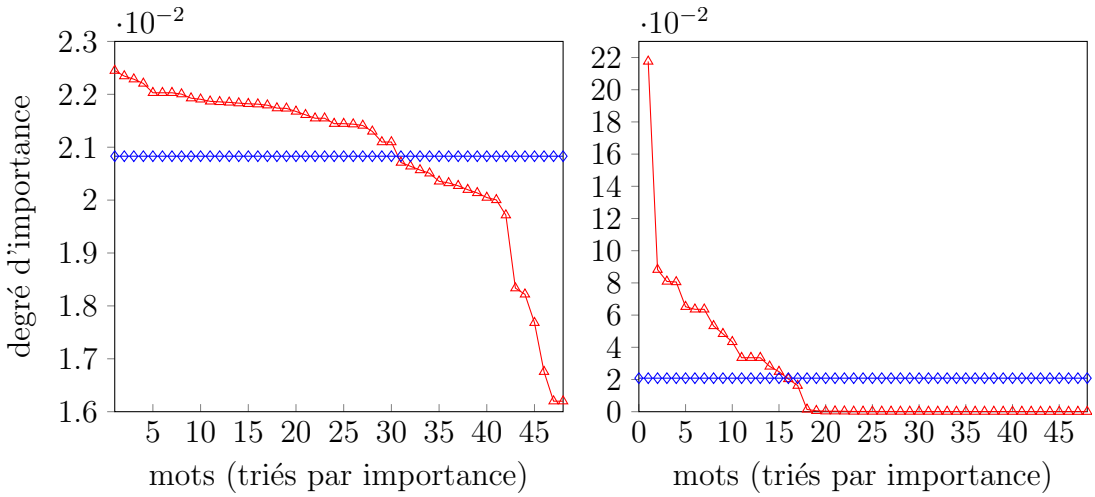


Figure 6: Degré d'importance des mots affectés par le modèle NBOW2, dans un document contenant 48 mots. Deux variantes du modèle NBOW2 sont présentées : à gauche la fonction f est une sigmoide et à droite f est un softmax. La ligne bleu horizontale représente l'importance utilisée par le modèle NBOW (même importance donnée à tous les mots).

6 Expérience de reconnaissance

La liste des OOV PN pertinents, récupérée par le modèle de contexte, doit être utilisée pour diminuer le taux d'erreur du système de reconnaissance. Dans nos travaux précédents, nous avons évalué l'efficacité de la liste des OOV PN obtenus à partir de modèles de contexte, en effectuant une recherche acoustique/phonétique. Dans [Sheikh et al., 2016b], nous avons effectué une recherche phonétique pour les N premiers OOV PN dans la meilleure hypothèse du système de reconnaissance. Dans [Sheikh et al., 2016c], nous avons effectué une recherche acoustique dans le réseau d'hypothèse de LVCSR basé sur un transducteur à états finis (FST *Finite State Transducer*). La récupération basée sur la recherche de mots clés permet une évaluation plus rapide, mais il en résulte de nombreuses fausses alarmes. Dans cette thèse, nous effectuons une seconde passe de reconnaissance en ajoutant les nouveaux noms propres récupérés au lexique.

6.1 Ajout des noms propres dans le système de reconnaissance

Pour ajouter les nouveaux noms propres au système de reconnaissance il est nécessaire d'ajouter ces mots dans le lexique phonétique et dans le modèle de langage. Pour générer les prononciations de ces nouveau noms propres, nous avons

utilisé le générateur automatique G2P (*Grapheme-to-Phoneme*) Sequitur [Bisani and Ney, 2008]. L'estimation des probabilités n-grammes de LM, pour de nouveaux mots est un problème non trivial et ouvert. La plupart des méthodes proposées reposent sur la similitude entre les nouveaux mots et des mots du vocabulaire [Orosanu and Jouvet, 2015, Qin, 2013, Lecorvé et al., 2011] ou utilisent des classes de mots dans le modèle de langage [Allauzen and Gauvain, 2005b, Pražák et al., 2007, Naptali et al., 2012]. Nous avons décidé d'ajouter seulement des probabilités unigrammes pour ces nouveaux mots. Les probabilités unigrammes sont ajustées en prenant une partie de la probabilité de <*unk*> et la probabilité unigramme est calculée comme suit :

$$p_{oov-pn-unigram} = p_{<unk>} \times \frac{\delta}{\#\text{ OOV PN}} \tag{14}$$

où δ est la fraction de probabilités de <*unk*> qui est attribué à tous les OOV PN à ajouter. Cette approche est similaire à l'utilisation d'une classe de mots.

6.2 Configuration de l'expérience pour la reconnaissance

Pour réduire le temps de calcul, nous avons défini un ensemble de tests plus petit pour ces expériences. A partir des 3000 vidéos de *Euronews* (Table 1), nous avons choisi un sous-ensemble de vidéos qui apparaissent dans 4 semaines choisies au hasard. Ce sous-ensemble de test comprend un total de 467 vidéos, parmi lesquelles 318 ont un ou plusieurs noms propres manquants (target OOV PN). On peut noter qu'il y a 149 vidéos dans ces données de test qui sont sans OOV PN. Comme ce serait le cas dans une configuration réelle, on ne sait pas à l'avance si la vidéo contient ou non des OOV PN. Les 318 vidéos contiennent un total de 1023 OOV PN (non uniques), parmi lesquels 483 peuvent être récupérés avec le corpus diachronique *L'Express*.

Nous effectuons la seconde passe de la reconnaissance de parole avec le système ANTS car il est rapide d'effectuer une mise à jour du modèle de langage pour chaque document[4]. Notre système de base (*baseline*) sera ANTS avec le lexique de base (asns OOV PN), désigné par *No-OOV*. Le système noté *LX-All* utilise ANTS dans lequel touts les 9300 OOV PN du corpus *L'Express* ont été ajoutés. Nous avons créé deux systèmes de reconnaissance pour lequels nous avons ajouté, lors de la deuxième passe, les 128 premiers OOV PN pertinent pour le document vidéo, récupéré par les modèles LDA et les NBOW2+. Ceux-ci seront désignés *LDA-128* et *NBOW2+-128*. Nous avons utilisé un autre sous-ensemble

[4]Le système KATS est basé sur Kaldi qui nécessite une longue ($\sim$ 6 heures) compilation des (HCLG) FST.

de vidéos *Euronews* (ne faisant pas partie du sous-ensemble de test) pour ajuster le paramètre δ dans equation (14). Après différents essais, nous choisissons une valeur de 0,001 pour δ, ce qui a donné une performance optimale pour chacune des méthodes. Une valeur plus élevée de δ permet d'améliorer la reconnaissance des target OOV PN mais elle entraîne une augmentation des fausses alarmes.

6.3 Résultats des reconnaissance

La Table 4 représente le taux d'erreur de noms propres (*Proper Name Error Rate* PNER) après la deuxième passage de reconnaissance de la parole. Le taux PNER est obtenu en alignant d'abord la référence et la transcriptions automatique au niveau du mot, puis en calculant les erreurs de substitution, d'omission et d'insertion en ne prenant en compte que les noms propres. A partir de la table 4, on peut observer que l'ajout de tous les OOV PN du corpus diachronique (LX-All) conduit à une augmentation de PNER. Les modèles contextuels LDA et NBOW2+ permettent la sélection de PN pertinents et la réduction du PNER. LDA et NBOW2+ montrent une performance PNER similaire, mais une analyse des erreurs a révélé que le modèle NBOW2+ conduit à une meilleure reconnaissance des noms propres et à moins de fausses alarmes. En outre, l'ajouter des nouveaux PN dans le vocabulaire et le modèle de langage n'a eu aucun impact négatif sur le WER.

Table 4: Les résultats de la deuxième passe de reconnaissance de la parole. PNER désigne Proper Name Error Rate.

		(OOV PN ajoutée)		
	No-OOV	LX-All	LDA-128	NBOW2+-128
% PNER	61.6	67.8	57.0	56.8

7 Conclusion

Les modèles de contexte sémantique permettent d'améliorer la récupération des noms propres hors vocabulaire en sélectionnant des OOVs pertinents pour un document audio. Nous avons analysé des méthodes basées sur la LSA, sur des modèles thématiques LDA et sur des embeddings obtenus à l'aide d'un réseau neuronal. Nous avons également étudié le modèle NBOW pour la tâche de récupération des OOV PN. Nous avons proposé une nouvelle extension du modèle NBOW qui permet de pondérer les mots importants pour notre tâche.

Lors d'expériences sur des vidéos d'actualité en français, nous avons montré que nos méthodes basées sur le modèle thématique LDA et les modèles NBOW peuvent récupérer jusqu'à 85% à 90% des noms propres manquants extraits d'un corpus diachronique. Les méthodes proposées basées sur des modèles de LDA et NBOW sont robustes aux erreurs du système de reconnaissance.

Les modèles NBOW et NBOW2 donnent une amélioration des performances de récupération par rapport à une méthode fondée uniquement sur des embeddings. L'apprentissage en deux phases et la méthode dropout permettent aux modèles NBOW d'obtenir de meilleures performances. La combinaison de modèles NBOW et NBOW2 conduit à une convergence plus rapide lors de l'apprentissage. Les OOV PN pertinents, récupérés par les modèles de contexte, ont été évalués en effectuant un deuxième passage de reconnaissance. Cette seconde passe de la reconnaissance montre une réduction absolue de 4,8 % du taux d'erreur de noms propres. Si tous les OOV PN du corpus diachronique sont simplement ajoutés au vocabulaire du système de reconnaissance, une forte dégradation est observée. D'autres améliorations sont possibles en utilisant des données diachroniques de plusieurs sources.

Dans un autre travail [Sheikh et al., 2016d], nous avons évalué le modèle NBOW2 sur les tâches de classification de texte, d'analyse de critiques de film et de classification thématique de texte issus de groupe de discussion. Nous avons montré la capacité d'apprentissage du modèle NBOW2 et qu'il surpasse les modèles utilisant des sac-de-mots (Bag-Of-Words.

CHAPTER 1

Introduction

Automatic Speech Recognition (ASR) systems are software programs which can automatically transcribe a spoken utterance into written text. Most ASR systems follow a statistical approach and try to predict the most likely sequence of sounds given the observed acoustic evidence from the speech recording. Instead of simply predicting a sequence of sounds or phonemes, which lacks word boundaries and are difficult to interpret, it is favourable to predict a sequence of words. This requires the speech recognition system to maintain a vocabulary of words, a pronunciation lexicon to map a sequence of phonemes to a word and a language model which defines the possible word sequences in that language. This hierarchical approach introduces some linguistic and grammatical knowledge into speech recognition, leading to better results and more interpretable transcriptions.

Automatic speech recognition systems are known as Large Vocabulary Continuous Speech Recognition (LVCSR) systems when they can transcribe a large set of words, typically ranging between 50,000 to 200,000 words. Given a reliable acoustic model, the vocabulary of the LVCSR system can grow larger if there is significant amount of text data available to train the language model. Words which do not have enough occurrences in the training data cannot be relied upon to learn reliable language model statistics. A practical choice is to leave out these words from the language model. However, it turns out that there are many such un-modelled words and they comprise important words, like names and entities, which carry important information.

This chapter serves as an introduction to this dissertation. It begins with an overview of the problem, and presents the motivation to study and address it. This includes a discussion on the part of the problem being focused in this dissertation[1]. This is followed by a brief discussion on adopted methodology and proposed models, and finally a description of the layout and remaining chapters of this dissertation.

[1]This thesis work is carried out under the ContNomina project supported by the French National Research Agency (ANR) under the contract ANR-12-BS02-0009. (Details available on the webpage: http://www.agence-nationale-recherche.fr/?Projet=ANR-12-BS02-0009)

LVCSR Transcription

le dirigeant nord-coréen troisièmes représentants de la dynastie au pouvoir depuis 1900 48 n'a pas explicitement nommé son oncle et en sont à être exécuté le 12 décembre officiellement pour crimes contre le parti des travailleurs au pouvoir et activités nuits en a l'international

Reference Transcription

Le dirigeant nord-coréen, troisième représentant de la dynastie au pouvoir depuis 1948, n'a pas explicitement nommé son oncle Jang Song-thaek, exécuté le 12 décembre, officiellement pour crimes contre le Parti des travailleurs au pouvoir et activités nuisant à l'intérêt national.

Figure 1.1: LVCSR and reference transcriptions of a sentence from a French broadcast news video from Euronews.

1.1 Overview of the Problem

1.1.1 Out-of-Vocabulary (OOV) Words

LVCSR systems cannot recognise words which are not present in their vocabulary, leading to the problem of Out-of-Vocabulary (OOV) Words. As an example, Figure 1.1 shows LVCSR and reference transcriptions of a sentence from a French broadcast news video[2]. The words highlighted in red color indicate the errors made by the LVCSR system. The name *Song-thaek* is not in the vocabulary of the LVCSR system and hence the LVCSR predicts a sequence of similar sounding words, *sont à être*, which are in the LVCSR vocabulary. On the other hand, the first part of the name *Jang* is present in the LVCSR vocabulary but it is mis-recognised. This is a common phenomenon where OOV words spread errors into the adjacent in-vocabulary words and affect the overall LVCSR performance.

[2]available at https://www.youtube.com/watch?v=YtM0mc_x0bo

1.1.2 Can we simply accumulate new words?

A simpler solution to the OOV problem would be to find new words and a significant amount of text data for these words to train the language model, and finally accumulate the new words into the LVCSR system. To discuss on this possibility we present Figure 1.2, which is taken from [Hetherington, 1995]. Hetherington plotted a graph of the vocabulary size obtained by varying the amount of training data taken from a corpus. As shown in Figure 1.2, he plotted this graph for nine different corpora, comprising different languages (English, French, Italian) and different communication styles (human-computer interactions, conversational speech and news corpora).

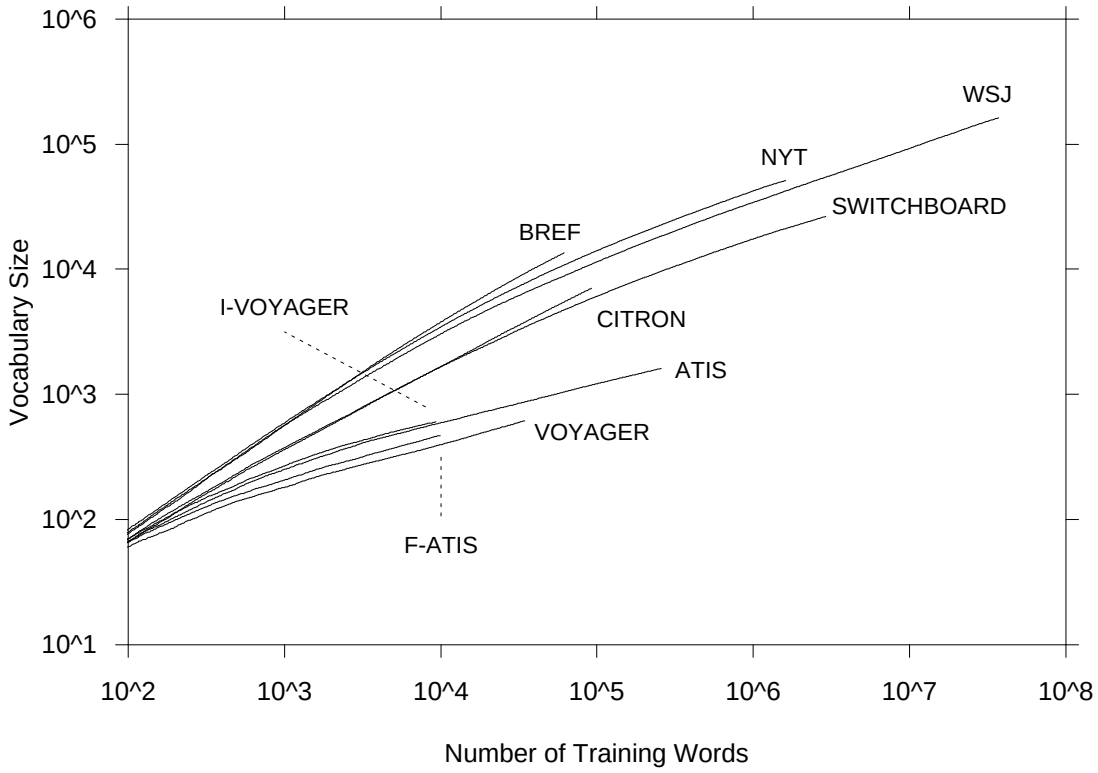


Figure 1.2: Vocabulary size versus amount of training data, taken from [Hetherington, 1995]. The datasets are in different languages with F-ATIS and BREF in French, I-VOYAGER in Italian and the remaining in English. They comprise different application domains including human-computer interactions (VOYAGER, ATIS), conversational speech (CITRON, SWITCHBOARD) and news corpora (BREF, NYT, WSJ).

A flatter curve in Figure 1.2 indicates that, as the amount of training data is increased the increase in vocabulary is lesser. This is true for corpora from limited vocabulary tasks such as ATIS, F-ATIS, VOYAGER and I-VOYAGER[3]. The vocabulary growth rate are much higher for the less restricted conversational speech corpora, CITRON and SWITCHBOARD, and highest for news corpora BREF, NYT and WSJ[4]. The trend of the graph for news corpora indicates that its vocabulary growth rate will not flatten and new words will keep increasing with the amount of training data. As an example, one could imagine names of all politicians, all possible places from various countries, the names of organisations and products, and other entities. A crawl on the internet for news articles in an particular language, like English, French, Italian, leads to a corpus with vocabulary of millions of words. Thus **continuously adding more words into the LVCSR system is not a practical solution**, as also discussed in previous works [Hetherington, 1995, Parada, 2011, Qin, 2013].

The growth in vocabularies is due to the accumulation of new words introduced by constantly evolving topics in the news. This brings our discussion to a more specific issue that we discuss in our research, that of diachronic audio documents.

1.1.3 Diachronic Documents and Proper Names

Documents like broadcast news and Youtube videos are diachronic in nature and are characterised by a variety of topics which change frequently with time. The appearance and disappearance of new events leads to many new words which are ultimately OOVs. To illustrate this phenomenon we present Figure 1.3. This figure shows a plot of count of new words appearing in news articles, from a French newspaper, during a period of 25 weeks. From these examples we can see that some OOV words are common and could be present throughout the timeline but others span a short time and can have fewer instances. Simply accumulating such new words into the LVCSR system would increase the LVCSR search space and complexity and lead to confusion with in-vocabulary words.

Similar to our illustration in Figure 1.3, previous works have reported that the majority of the new or OOV words are proper names (PNs). Percentage of proper names in OOV words been reported as: 56% by Qin [Qin, 2013], 66% by Parada [Parada et al., 2011a], 57.6% by Palmer [Palmer and Ostendorf, 2005],

[3] ATIS stands for Air Travel Information Services [Hemphill et al., 1990] and VOYAGER is a system for locating some services in a specific area [Zue et al., 1990].

[4] WSJ is for Wall Street Journal [Paul and Baker, 1992], NYT for New York Times, and BREF has read speech from French newspaper Le Monde [Lamel et al., 1991]

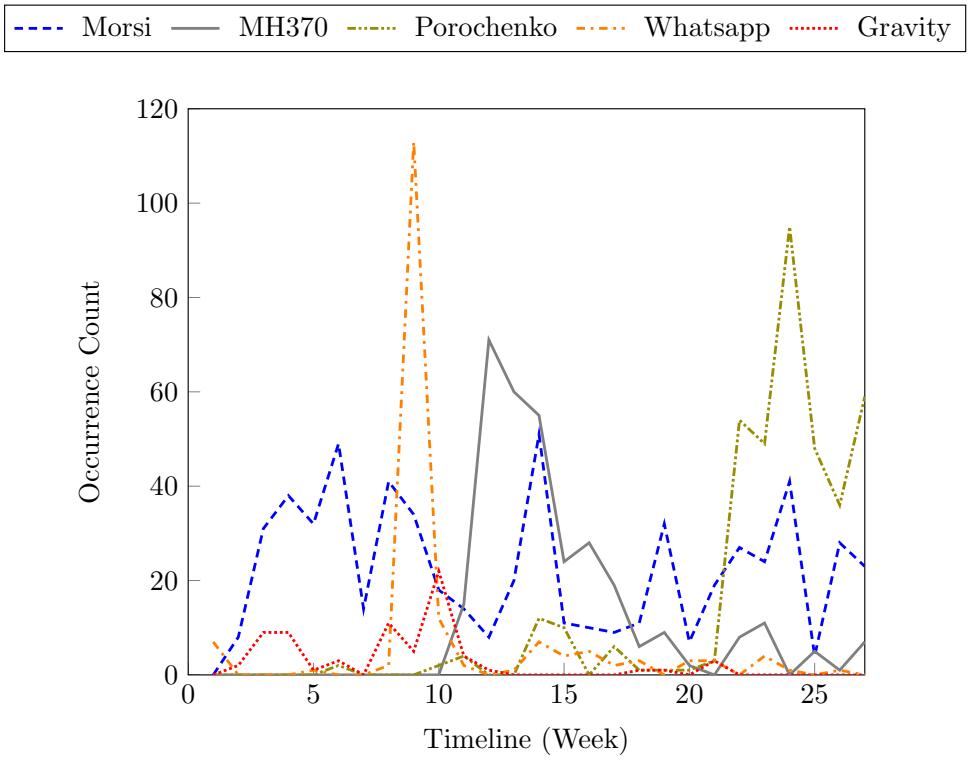


Figure 1.3: Time versus frequency distribution of new words in a news corpus. The words are new and OOV with respect to our automatic news transcription system described in Section 2.4.2. Occurrence count shown were obtained from news articles appearing in the French newspaper *L'Express* during January-June 2014. (*Mohamed Morsi* is an Egyptian politician, *MH370* is the Malaysia airlines flight which disappeared, *Petro Porochenko* is the fifth and current President of Ukraine, *WhatsApp* is a mobile application, *Gravity* is the name of Hollywood movie.)

70% by Allauzen [Allauzen and Gauvain, 2005b], 72% by Bechet [Béchet et al., 2000]. We observed similar statistics in our analysis. As discussed in Section 2.4.1, about 72% of OOV words in our test set, comprising of broadcasts videos from Euronews, are proper names and about 64% of the videos contain OOV proper names. Speaking in terms of number of OOV proper names, there are about 10K new proper names when looking at 6 months of recent news articles from only one website. This number increases to 18K for articles from two news websites (see Section 4.6.1). With this trend the total number of OOV proper names will be in hundreds of thousands for an LVCSR system trained on the

French Gigaword corpus with 16 years of news data[5]. (It should also be noted that for highly inflected languages like Portuguese this may not be completely true. Inflections of verbs contribute majority of OOV tokens and reduce the percentage of OOV proper names. For instance as reported in [Martins et al., 2006], proper names contribute about 29% of the OOVs as compared to 57% of verb inflections.)

Broadcast news transcription is one of the most widely studied LVCSR setup, and as discussed above it faces the problem of OOV proper names. Proper names in audio/video news are of prime importance for content based indexing and browsing applications as well as to produce accurate and reliable transcriptions, as partly evident from the example in Figure 1.1. This motivates us to study the problem of OOV proper names in broadcast news, i.e. names which appear in diachronic broadcast news audio but are not present in the LVCSR vocabulary and language model and hence cannot be recognised by the LVCSR system.

1.1.4 Approaches to Address the OOV Problem

Approaches addressing the OOV problem in LVCSR systems can be put into two categories as follows.

- **OOV detection based approaches** [Qin and Rudnicky, 2012, Parada et al., 2010a, Kombrink et al., 2012], which aim to detect the presence of OOV words and/or locate OOV regions in the LVCSR hypothesis. These approaches mainly learn from outputs of one or more speech recognition engine. A related and more interesting approach is based on hybrid language models, which enable hypothesis of both word and sub-word units. This approach forms the motive for most open vocabulary ASR systems [Bisani and Ney, 2005, Shaik et al., 2015].

- **Vocabulary selection approaches**, which propose a relevant vocabulary for speech recognition using additional text data. They try to minimise the OOV rate for a domain specific corpus [Allauzen and Gauvain, 2005a, Liu et al., 2007] or for a daily update system [Martins et al., 2007]. Document specific vocabulary selection methods [Oger et al., 2008b, Meng et al., 2010] are more dynamic as they propose context specific vocabulary.

The problem with OOV detection approaches which learn from outputs of speech recognition is that it requires speech datasets with manual and automatic LVCSR transcriptions, and features obtained from LVCSR decoding, to train

[5]https://catalog.ldc.upenn.edu/LDC2011T10

classifiers for detecting OOVs. Modelling together word and sub-word units is more promising as it does not necessarily depend on OOV detection or knowledge of possible OOV words. However this approach requires an optimal selection of sub-word units, so that it does not impact the recognition of in-vocabulary words.

Vocabulary selection based approaches are more flexible as they enable dynamic vocabulary selection for speech recognition. They require (a) the knowledge of possible OOV words, as also required for OOV word recovery post OOV detection, and (b) related text data for training the selection methods. However, it is easier to meet these requirements in most LVCSR tasks, especially with the availability of text data on the internet. One may argue that this data might as well be used to train language models to include the OOV words in the LVCSR system. However, this would bring us back to the questions (a) is the amount of data sufficient for training reliable language model, and (b) how many of the new words should be added in the LVCSR models.

1.2 Adopted Methodology

In this dissertation, **we adopt a document specific vocabulary selection approach and focus on the problem of retrieving relevant OOV words** in large vocabulary continuous speech recognition. Earlier proposed document specific vocabulary selection methods rely on web search engines to retrieve text documents containing relevant words [Oger et al., 2008a] and/or rely on methods based on term frequency, document frequency and word co-occurrence features for selecting new words [Allauzen and Gauvain, 2005b, Martins et al., 2007, Meng et al., 2010, Maergner et al., 2012, Nkairi et al., 2013]. As opposed to relying on ad-hoc methods and count based hand crafted features, we adopt unsupervised and theoretically well defined methods for document vocabulary selection.

In our approach, we collect a corpus of in-domain text documents from the web, which contain new/OOV proper names. This corpus, referred to as the *diachronic text corpus*, is the input to train our OOV proper name retrieval models. This model learns the semantic and topic context of the OOV proper names, rather than simply relying on search engines or text matching techniques. Given a test speech utterance, our models infer the semantic/topic context of the spoken content and hypothesise a list of context relevant OOV proper names. This list can then be used to recognise/recover the *target OOV proper names*[6]

[6]All new proper names in the diachronic text corpus, and not in LVCSR vocabulary, are referred as OOV proper names. Those actually present in the test speech are referred as target OOV proper names.

present in the speech. The recovery can be performed using a second pass of speech recognition with an updated language model or using keyword spotting techniques. The methods for recovery are not the problems that we focus on. Instead, for evaluation we perform a second pass speech recognition in which the relevant OOV proper names are added as unigrams into the language model. Throughout this dissertation, our aim is to model and exploit semantic/topic context to retrieve relevant OOV proper names. Our motivation to explore semantic/topic context models for addressing the OOV problem in LVCSR comes from a more fundamental question - *How can we leverage semantic and topic context to improve LVCSR transcriptions?*

Semantic and topic information can be exploited in different ways in our task. One could model the local level semantic information of OOV proper names, relying on its surrounding word sequences, or a more global document level topic information. However LVCSR transcriptions are prone to word errors and noise in word sequences, and secondly have no direct information about the position of OOVs. So it is less favourable to rely on local level semantic information of OOV proper names. In our approach **we rely on a document level semantic and topic context of LVCSR transcriptions to retrieve relevant OOV proper names**.

Semantic and topic models have been widely studied in the field of Computational Linguistics and Natural Language Processing, more specifically under Distributional Semantics [Turney and Pantel, 2010]. Among these, the Latent Semantic Analysis (LSA) [Deerwester et al., 1990] and Latent Dirichlet Allocation (LDA) [Blei et al., 2003] models have been the most prominent methods for automatically extracting underlying topic/semantic representations in documents. While LSA obtains semantic spaces using matrix decomposition method from linear algebra, LDA learns topic distributions using a hierarchical Bayesian analysis. More recent developments on Neural Network based language models [Mikolov et al., 2013c] have led to interesting semantic/syntactic word and context representations.

This dissertation focuses on two main issues: (a) robustness of semantic/topic representations to speech recognition errors, because in our task the context is inferred from an LVCSR hypothesis, and (b) effectiveness of the representations for less frequent OOV proper names, because most OOV proper names do not have many instances in the training data. **One of the main contribution of this dissertation is a methodology to retrieve relevant OOV proper names, which generalises to different semantic and topic space representations**, including LSA, LDA and word embedding spaces. We perform a thorough analysis of how these representations can be exploited for our task.

The second main contribution is models to learn discriminative context representations. LSA, LDA and word embedding models learn representations by maximising training data likelihood. Following the detailed analysis of these different representations, we argue that they are not the most optimal for our task. We propose discriminative context representations trained with an objective to maximise the performance of retrieval of OOV proper names. The proposed discriminative models outperform the different semantic/topic representations. Our Neural Bag-Of-Weighted-Words (NBOW2) model learns to assign task specific importance weights to words. The effectiveness of our NBOW2 model is also evaluated on standard topic classification and sentiment analysis tasks.

In this dissertation, we present experiments performed on retrieval of OOV proper names in French broadcast news videos. However, our proposed methodology and the discriminative context representation models are language independent and readily apply to other types of diachronic audio/video documents as well as non proper name OOVs. We would also like to highlight that the proposed methodology does not require any supervision or labelled data.

1.3 Thesis Layout

The dissertation is organised as follows. In Chapter 2 we present a background, providing a very generic and technical description of automatic speech recognition and a glimpse of the major revivals in LVCSR research. This is followed by a survey of previous works addressing the OOV problem and finally the details about our LVCSR setup, corpora, tasks and evaluation measures.

As we would like to exploit semantic and topic context for our task of OOV proper name retrieval, we dedicate Chapter 3 to introduce models from distributional semantics. In this chapter we describe the LSA model, the LDA topic model and the Continuous Bag Of Words (CBOW) and Skip-gram word embedding models [Mikolov et al., 2013b].

In Chapter 4 we present the proposed methodology for retrieval of OOV proper names. We introduce our retrieval methods using topic representations from LDA and extend these to semantic vectors from the LSA model and representations from LDA based Entity-Topic models. An evaluation of the proposed methods is performed and the performances for all these representations are compared. This chapter also presents a discussion on the problem of selection of diachronic text corpora from the internet, which is essential for training the context models and to retrieve OOV proper names.

In Chapter 5 the proposed retrieval methods are extended to word embedding space representations. Similar to chapter 4, we analyse the performance of the representations with different model hyper-parameters and finally compare the best performing semantic/topic representations. A detailed analysis reveals the inadequacies with these representations.

Chapter 6 presents the proposed discriminative context representations. It describes the model architectures and the training procedure for our Neural Bag-Of-Words (NBOW) model and the Neural Bag-Of-Weighted-Words (NBOW2) model. This is followed by a detailed analysis of the proposed methods to boost robustness and performance of these models. After presenting the improvement in retrieval performance with these models, we try to evaluate it in terms of recovery of the target OOV proper names in audio documents. We also evaluate the NBOW2 model on standard topic and sentiment classification tasks.

The conclusions and future perspectives drawn from this dissertation are presented in Chapter 8.

CHAPTER 2

Background

This chapter provides a general background relevant to this dissertation. It is divided into three sections. In the first section we give an overview of Large Vocabulary Continuous Speech Recognition (LVCSR) systems, providing a very generic technical background as well as a glimpse of the major revivals in LVCSR research. The second section briefly presents the Out-of-Vocabulary (OOV) problem faced by LVCSR systems and mainly discusses the previous works addressing the OOV problem, followed by the approach that we have adopted. The last section provides details and description of our study, including the LVCSR setup, corpora, tasks and evaluation measures.

2.1 Large Vocabulary Continuous Speech Recognition

Automatic Speech Recognition (ASR) systems produce a transcription of a spoken utterance into the corresponding string of words. These systems are known as large vocabulary continuous speech recognition systems when they can transcribe a large set of possible words, typically ranging between 50,000 to 200,000 words. The *continuous* attribute of LVCSR indicates that the speech signal is processed to obtain a sequence of words in a continuous manner as, opposed to isolated word recognition systems. Historically, most LVCSR systems were developed during the 1990's[1,2] under the continuous speech recognition programmes funded by the Advanced Research Projects Agency (ARPA) [Young and Chase, 1998].

Given the task of transcribing spoken utterances, sentences and even dialogs and discourse into the corresponding sequence of words, LVCSR systems are evaluated with a measure termed as Word Error Rate (WER). The WER measure

[1] In the years before, systems with thousand or more words were sometimes referred as large vocabulary systems.

[2] The Broadcast News (BN) and the late Wall Street Journal (WSJ) and North American Business (NAB) news tasks during the mid-1990's had a vocabulary of about 65,000 words.

is itself derived from the edit distance [Navarro, 2001] or Levenshtein distance (attributed to Levenshtein [Levenshtein, 1966]). Denoting N as the total number of words in the reference transcription, D as the number of words missed or deleted by the LVCSR, I as the number of erroneous words inserted by the LVCSR itself and S as the number of words in the reference transcription that are substituted by the LVCSR with some other words, WER is calculated as:

$$WER = (D + I + S)/N \tag{2.1}$$

Most practical and state-of-the-art LVCSR systems take a statistical approach and pose speech recognition as a problem to estimate the most likely sequence of words $\widehat{W} = \hat{w_1}\hat{w_2}\hat{w_3}\cdots\hat{w_n}$ given the observed acoustic evidence A. This can be formally written as:

$$\widehat{W} = \underset{W \in \mathcal{L}}{argmax} \quad P(W|A) \tag{2.2}$$

where $P(W|A)$ denotes the probability of a word sequence W, drawn from a language $\mathcal{L}$, given that acoustic evidence A was observed. Using Baye's rule, Equation 2.2 becomes:

$$\widehat{W} = \underset{W \in \mathcal{L}}{argmax} \quad P(A|W) \times P(W)/P(A) \tag{2.3}$$

$$\approx \underset{W \in \mathcal{L}}{argmax} \quad P(A|W) \times P(W) \tag{2.4}$$

where we can see that the problem of finding the best word sequence relies on maximising the acoustic likelihood $P(A|W)$ given the language prior $P(W)$. The acoustic likelihood $P(A|W)$ of a word is obtained from an *Acoustic Model* (AM) whereas the prior $P(W)$ comes from a *Language Model* (LM) which defines the possible word sequences in that language. These models are learned from hundreds of hours of speech recordings and their transcriptions. To perform speech recognition, the LVCSR system includes a *decoder* module which performs the search for the best likely word sequence.

Instead of modelling the acoustics of each possible word separately words are split into smaller units, most commonly a sequence of phonemes looked up from a pronunciation *lexicon*. The total number of these sub-word units is fix, for example 40-45 phonemes for English and French. Composing words from a fixed set of smaller units gives the flexibility to include new words into the speech recognition vocabulary. The language model is most commonly built on word units and gives the flexibility to recognise different sequence of words. This hierarchical approach adds linguistic and grammatical constraints to the speech recognition problem, leading to more accurate results as compared to transcribing a speech signal into a sequence of characters or phonemes.

Figure 2.1 illustrates a generic processing hierarchy of large vocabulary continuous speech recognition systems. A generic description of this hierarchy, independent of the specific underlying modelling techniques, is as follows.

- *Signal Level*: At the lowest level is the input speech signal, which is nothing but the acoustic signal captured by a microphone and digitised into a discrete waveform. Since the speech signal is non-stationary and changes continuously with time, it is split into short overlapping frames, generally of 25 milliseconds length and 10 milliseconds overlap, for further processing.

- *Feature Level*: In the next level, important acoustic features are extracted from each speech frame and each frame is now represented by a feature vector. This step discards the useless information from the raw frame level representation of the speech signal and reduces its dimensionality for statistical modelling. To capture low level temporal dynamics, feature vectors corresponding to each frame can use information from feature vectors corresponding to previous few frames.

- *Sequence of States*: The temporal patterns in the feature vectors are captured by a sequence learning model, for example Hidden Markov Model (HMM). A sequence learning model maintains a particular sequence of states for each pattern to be recognised, for example a sequence of feature vectors corresponding to a phoneme. The sequence of observed feature vectors remain in the same state as long as they are similar to previous ones. Changes in temporal pattern of the observed feature vectors cause a transition from the current state to the next state in the state sequence.

- *Sequence of Sub-word Units*: A sub-word unit is composed of a particular sequence of states. Depending on the choice of sub-word units, a sequence of feature vectors may trigger the sequence of states corresponding to two acoustically closer sub-word units, for example 'p' and 'b'. The context of preceding and following sub-word units can disambiguate (but cannot eliminate) such possibilities. Generally, there is a lexicon which maps a sequence of sub-word units to words.

- *Sequence of Words*: A language model built on word units, or sometimes even on sub-word units, is at the top most level of the LVCSR processing hierarchy. It models the possible words (or word sequences) that could be present in a speech utterance.

In the LVCSR processing hierarchy, the acoustic model links the feature level to the sub-word sequence level whereas the language model covers the word level

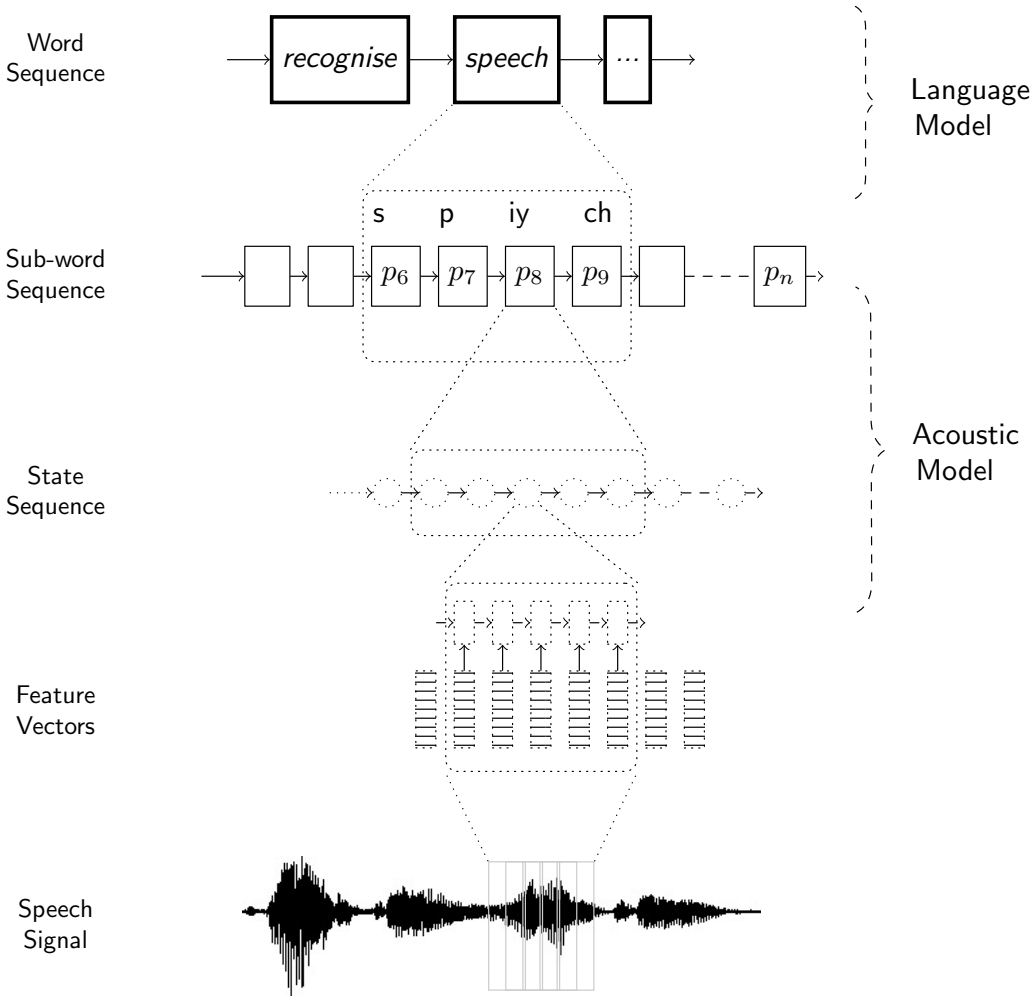


Figure 2.1: A generic processing hierarchy of large vocabulary continuous speech recognition systems.

and sometimes also the sub-word sequence level. It must be noted that, as opposed to first predicting a sequence of sub-words and then fitting a sequence of words over it, most LVCSR systems follow an integrated approach. In this approach, a speech recognition decoder loads a language model in the form of a graph, with words as nodes in the graph and arcs connecting those words which can appear in a sequence. The words are eventually decomposed into sub-word and then state sequences. Given a sequence of feature vectors representing the speech utterance, the decoder performs the search for the most likely state sequences and hence hypotheses of the most likely word sequences. On the contrary, some very recent systems [Hannun et al., 2014] based on Connectionist Temporal Classification first recognise a sequence of phonemes/characters from speech and then use a slightly sophisticated language model to obtain the best word sequence.

The earliest LVCSR frameworks were discussed in [Rabiner and Juang, 1993, Young, 1996, Jelinek, 1997] and until a few years ago the state-of-the-art LVCSR systems were based on this framework [Gales and Young, 2007, Hinton et al., 2012a]. In the recent few years the progress in machine learning and speech recognition has led to improved acoustic models and language models as well more efficient decoders for LVCSR systems. We will present a brief overview of the most prominent acoustic modelling and language modelling methods that have have led to major breakthroughs in LVCSR technology in the past few decades. Since the main focus of this dissertation is on modelling context for dynamic vocabulary selection in LVCSR, the description does not go into the technical depth of individual acoustic and language models.

2.1.1 LVCSR Acoustic Modelling

In the early days of speech processing and recognition, extracting feature vectors from speech signal was a hot research topic. This lead to interesting developments including the Linear Predictive Coding (LPC) [Atal and Hanauer, 1971], and later Mel-Frequency Cepstral Coefficients (MFCC) [Davis and Mermelstein, 1980] and Perceptual Linear Prediction (PLP) coefficients [Hermansky, 1990]. See [Anusuya and Katti, 2011] for a detailed review. Except for some very recent LVCSR systems, acoustic modelling involves training a sequence learning model on these automatically extracted, but specifically engineered, sequence of acoustic feature vectors.

Hidden Markov Models have been the most common and widely used approach for modelling sequences of acoustic feature vectors. Figure 2.2 shows a diagrammatic representation of an HMM. The HMM has two types of variables

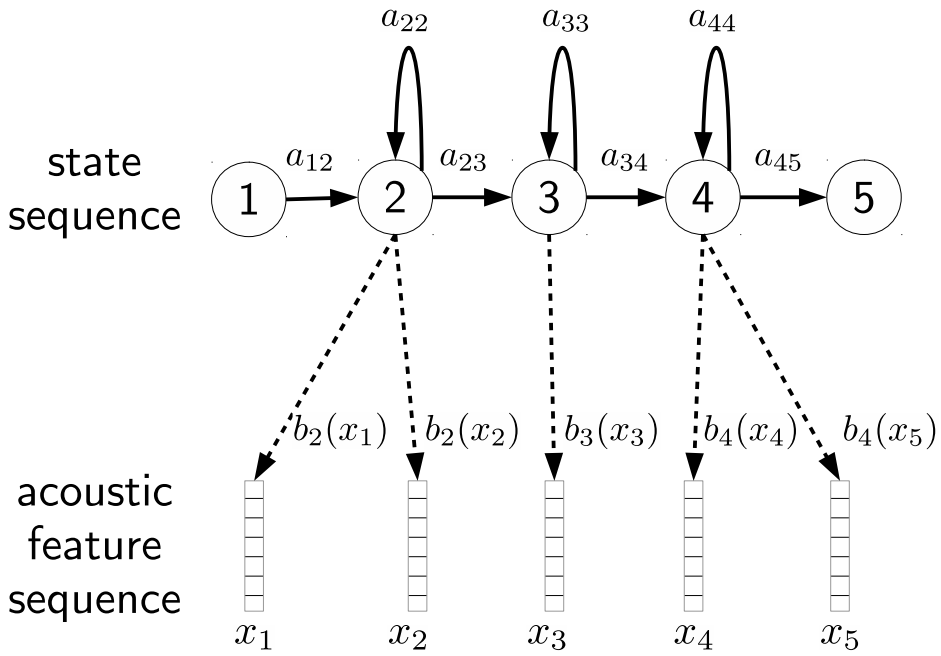


Figure 2.2: Hidden Markov Model (HMM) based acoustic modelling.

(a) the observed variables (denoted as $x_1, x_2...x_5$ in Figure 2.2), which are the acoustic feature vectors in speech recognition applications, and (b) the hidden variables or the hidden states (denoted as s_1, s_2, s_3 in Figure 2.2), where each state represents a probability distribution over possible feature vectors. While the HMM is a generative model we will present a simpler description in terms of decoding during speech recognition. We refer to [Rabiner, 1989] for a more detailed discussion on operation and training of HMMs. At every time step or feature vector in the incoming sequence of feature vectors, the HMM takes a transition from the current state to either the next state or to the current state itself. The probability of making a transition from one state s_i to another state s_j is given by the probabilities a_{ij}. The probability of a feature vector belonging to, or in generative sense output vector being generated by, a state s_j is given by $b_j()$.

For many years (about 1990's - 2012), the state-output distribution for each state was modelled by a Gaussian Mixture Model where each component of the mixture model is a Gaussian probability density function. We refer to [Gales and Young, 2007] for more details. The complete GMM-HMM acoustic model, including model parameters $[\{a_{ij}\}, \{b_j()\}]$ are efficiently estimated from a corpus of spoken utterances and their corresponding transcriptions, using the Baum-Welch algorithm [Baum et al., 1970] which is a special case of the more general

Expectation-Maximisation (EM) algorithm [Dempster et al., 1977]. Over the years, many refinements in the GMM-HMM acoustic model have been presented, some of them including discriminative parameter estimation, acoustic feature projections, vocal tract length normalisation and algorithms for adaptation and noise compensation [Gales and Young, 2007]. As a result these GMM-HMM based models became very successful and they could not be easily outperformed by other acoustic modelling techniques, especially in LVCSR systems.

Despite the continuing success of GMM-HMM acoustic models it was known that these models had inherent shortcomings. The GMMs, which were mainly driving the HMM based acoustic models, amount to several thousands of Gaussian densities and lead to statistically inefficiencies, as pointed out in more recent works on acoustic modelling [Mohamed et al., 2012, Hinton et al., 2012b, Dahl, 2015]. There were early attempts to replace the GMMs with an artificial neural network [Bourlard and Morgan, 1993, Ellis and Morgan, 1999]. However after many years, the idea and success on deep learning and Deep Neural Networks (DNN)[3] [LeCun et al., 2015, Schmidhuber, 2014], lead to a promising DNN-HMM hybrid acoustic model [Mohamed et al., 2009]. Record breaking results were presented on small scale speech recognition task [Mohamed et al., 2012] and on LVCSR [Dahl et al., 2011]. These acoustic models were commonly know as DBN-HMMs in those days as Deep Belief Networks (DBN) [Hinton et al., 2006] were used for pre-training the multiple layers of the DNN. This training mechanism was replaced by other techniques resulting in further improvements [Dahl, 2015]. Since 2012, DNN-HMM acoustic models have shown striking gains in performance, as witnessed by several speech recognition groups [Hinton et al., 2012b], leading to state-of-the-art LVCSR systems. We refer interested readers to [Yu and Deng, 2014] for a detailed discussion on DNN-HMM acoustic models and their comparison with GMM-HMM acoustic models.

With the continuing development in the field of neural networks and deep learning there have been several new proposals for acoustic modelling. Some of the prominent ones being the use of Convolutional Neural Networks (CNN) for extracting features and acoustic modelling [Sainath et al., 2013, Abdel-Hamid et al., 2014] and the use of Recurrent Neural Networks (RNN) for performing end-to-end speech recognition without requiring feature extraction and HMM acoustic models [Graves and Jaitly, 2014].

[3]multiple layers of artificial neural networks

2.1.2 LVCSR Language Modelling

The task of the language model is to capture the possible sequence of symbols (usually words) in a given language. In small vocabulary applications, the expected word sequences can be specified by a finite-state or context-free grammar [Bahl et al., 1978]. However for LVCSR systems a more feasible, and a long standing, solution has been a Statistical Language Model (SLM)[Bahl et al., 1983]. An SLM is trained over a (sufficiently large) text corpus to capture a probability distribution over sequences of words. Once an SLM is trained it can be used to calculate the likelihood of a new sequence of words or to predict the next word for the given word sequence. The probability of a sequence of N words is given as:

$$p(w_1, w_2, w_3, ..., w_N) = \prod_{i=1}^{N} p(w_i | w_1, w_2, ..., w_{i-1}) \tag{2.5}$$

Assuming an n^{th} order Markov property, the probability of the word w_i can be approximated by the probability of observing this word after the preceding $n-1$ words as:

$$p(w_1, w_2, w_3, ..., w_N) \approx \prod_{i=1}^{N} p(w_i | w_{i-(n-1)}, ..., w_{i-1}) \tag{2.6}$$

This is referred to as the n-gram language model, with unigram, bigram and trigram being a common term referring to the case where n is 1, 2 and 3 respectively. The conditional probabilities can be directly obtained with the n-gram counts from the training text corpus. For instance the unigram, bigram and trigram probabilities can be obtained with the following equations:

$$\begin{aligned} p(w_1) &= \frac{\text{count}(w_1)}{\text{total word count of corpus}} \\ p(w_2|w_1) &= \frac{\text{count}(w_1, w_2)}{\text{count}(w_1)} \\ p(w_3|w_1, w_2) &= \frac{\text{count}(w_1, w_2, w_3)}{\text{count}(w_1, w_2)} \end{aligned} \tag{2.7}$$

This kind of counting can be problematic because many valid, but unseen, word sequences will have a zero n-gram count and hence the probability estimates are going to be zero. This problem is addressed by discounting or smoothing techniques, with Good-Turing smoothing [Katz, 1987] and modified Kneser-Ney smoothing [Kneser and Ney, 1995] being the more widely used smoothing techniques.

The simple smoothed n-gram language models have several shortcomings for which they receive great criticism from linguists. On the other hand they have

been quite successful as building blocks in LVCSR and other natural language processing systems. Over the years, researchers have presented interesting modifications for the standard n-gram language models and we will briefly mention some of the popular ones. For more details, we refer the interested readers to [Rosenfeld, 2000, Goodman, 2001, Mikolov, 2012].

One of the problems with n-gram models is to capture long term history. For instance the model could make a better prediction of the next word if the topic was known rather than simply relying on a sequence of handful of words[4]. Cache models [Jelinek et al., 1991] address this problem by dynamically estimating an n-gram model by interpolating the recent long term history (about hundreds of words) with a standard n-gram model. The latent semantic analysis based language model [Bellegarda, 2004a] is an interesting variant of the cache language model which takes into account the semantic representation of the text. Even the trigger based model [Lau et al., 1993] can be seen an a variant of a cache model. Another problem with n-gram models is that of data sparsity in higher order n-grams. A very simple example of this problem could be that of the name of the months and days in a week. While they can have some specific word histories, for example in case of specific events, it is easy to imagine that they can appear interchangeably and hence share their word history contexts. The class based model is proposed as a solution to this problem and it has many existing variants [Goodman, 2001]. A class based model replaces words by their classes, a class of months and a class of days of the week in our example, and estimates an n-gram model on these classes instead of those words. If a class is encountered during word prediction, then the words contained in that class are chosen based on their probability within the class. More linguistically motivated statistical language models are the structured language models in which a sentence is seen as a tree generated from a context free grammar. Probabilistic context free grammars are learned during the training phase from parsed or annotated text corpora. A slightly different kind of language model worth mentioning is the maximum entropy (or exponential) language model which expresses the probability of a word given a history in terms of arbitrary features corresponding to the word-history pair using an exponential model[5]. Important contributions in maximum entropy language models were presented in [Rosenfeld, 1996, Chen, 2009].

Similar to the research in acoustic models for speech recognition, even the statistical n-gram language model were challenged by statistical language models based on neural networks. The main idea behind these models was to represent words as vectors in a continuous space and to predict a word by comparing its vec-

[4]Simply increasing the order n of the n-gram model will start overfitting the data and does not help.

[5]more commonly known as logistic regression among the machine learning community

tor with the vector corresponding to the context of the word. The context vector is composed from the vectors of the words in the context. Under this representation and learning framework, similar words were arranged close to each other in the continuous vector space and shared similar context. Without requiring an exact match in context, this helped to overcome the problem of exponential increase of parameters as faced by an n-gram model.

The first successful neural network language model was based on a feed-forward neural network architecture [Bengio et al., 2001, Bengio et al., 2003] and it was later tried for LVCSR [Schwenk and Gauvain, 2002]. The feed-forward neural network was later replaced by a recurrent neural network architecture [Mikolov et al., 2010], giving the ability to capture long term contexts, leading to significant improvements in the language model. The training problems related to RNN language models were addressed by the Long-Short Term Memory (LSTM) based recurrent neural network language model [Sundermeyer et al., 2015], leading to further improvements. More recent works have explored several other neural network architectures [Józefowicz et al., 2016], also including character level information.

Irrespective of the methods used, a common challenge with language models is to adapt it to newer scenarios. Adapting a language model to new domain and datasets has been of interest to researchers in speech recognition, almost as much as the language modelling problem itself [Bellegarda, 2004b, Mikolov et al., 2010]. A related and more severe problem is that of handling new words which were not seen before and/or when there is not enough data to include them into the language model. As this dissertation deals with this specific issue, we devote the next section to discuss more about it.

2.2 The Out-of-Vocabulary Problem in LVCSR

LVCSR systems give the flexibility to include new words into the speech recognition vocabulary as well as to recognise new sequence of words. However a limitation to this is the amount of text data available to train the language model, irrespective of the underlying method. The words which do not have enough occurrences in the language model training data cannot be relied upon for getting n-gram statistics and as a practical choice they are excluded from the language model. This leads to the Out-of-Vocabulary (OOV) problem, in which the LVCSR cannot recognise words which are not present in the LVCSR vocabulary and language model. As a result the LVCSR substitutes the spoken OOV word with a similar sounding In-vocabulary (IV) word, or group of in-vocabulary words, in the output text. This dissertation deals with the problem of OOV

words for LVCSR systems. We devote this section to discuss about the previous works addressing the OOV problem, before discussing the severity of the problem and our task setup in Section 2.4.

Previous works addressing OOV words in LVCSR systems can be put into two main categories (a) OOV detection based approaches and (b) vocabulary selection based approaches. The OOV detection based approaches aim to detect presence of OOV words and/or locate OOV regions in the speech recognition hypothesis. On the other hand, vocabulary selection based approaches try to directly recognise or recover OOV words using knowledge of possible OOV words and/or task specific information. We present different techniques that were proposed along these two kind of approaches, followed by a discussion summarising the advantages and dis-advantages of each which finally led to the choice of our adopted approach.

2.2.1 OOV Detection Based Approaches

OOV detection based approaches try to detect and/or locate OOV regions in the speech recognition hypothesis. Most of these approaches can be used without any knowledge of the OOV words. The earliest works in OOV detection explored a *generic word model* which could be augmented to the IV words in the speech recogniser [Asadi et al., 1991, Katunobu et al., 1992, Suhm et al., 1993, Hayamizu et al., 1995, Fetter, 1998, Bazzi and Glass, 2000]. The generic word model could hypothesise a sequence of sub-word units and therefore could represent any possible word. The idea was to recognise the IV words when a known sequence of phones (or features) are encountered or simply substitute a sequence of sub-word units otherwise. While there were other works which relied only on confidence scores from the usual word only speech recognition [Suhm et al., 1993, Young, 1994], the idea of modelling and recognising sub-word units formed the base for most works which address the OOV problem in LVCSR. We have further divided the OOV detection approaches into three categories based on the methods and information used for OOV detection.

2.2.1.1 Hybrid Language Models

Hybrid language models are statistical n-gram language models trained with word and sub-word units to enable recognition of fragments of OOV words in between IV words. In this approach, the choice of the sub-word units is very crucial and there have been several works discussing and comparing different type of sub-word units [Yazgan and Saraclar, 2004, Choueiter, 2009, Rastrow

et al., 2009a, Qin et al., 2011]. Automatically generated sub-word fragments [Klakow et al., 1999, Bazzi and Glass, 2000, Qin et al., 2011, Parada et al., 2011b] have been shown to be an optimal choice. Furthermore, some of the works have studied combining and mixing of different type of sub-word units [Bazzi and Glass, 2002, Qin et al., 2012, Qin and Rudnicky, 2012].

In the context of hybrid LM speech recognition systems, also worth mentioning are open vocabulary systems which model sub-word units. Some works in this direction are that of [Bisani and Ney, 2005, Rastrow et al., 2009b, Shaik et al., 2012, Shaik et al., 2015]. In contrast to the above methods supporting hybrid LM, one of the previous works [Gerosa and Federico, 2009] showed that simply expanding the lexicon using a *Grapheme-to-Phoneme* (G2P) system and incorporating these words in LM training could reduce the word error rate.

2.2.1.2 Word and Sub-word Recogniser Output Combination

While the hybrid LM systems combine word and sub-word units before the recognition process there are works which combine the outputs of independent word and sub-word unit recognisers. To detect OOV regions, some methods perform alignment of the independently generated word and sub-word hypothesis [Lin et al., 2007, White et al., 2008] and other methods train classifiers with recognition posteriors and confidence scores as input features [Ketabdar et al., 2007, Kombrink et al., 2009]. These classification models have been used along with an alignment error model [Hannemann et al., 2010] and further extended to include scores from a hybrid word sub-word recogniser [Kombrink et al., 2012].

2.2.1.3 Including Language Context for OOV Detection

Words in any language follow linguistic and grammatical constraints and thus exhibit some contextual features, for example part-of-speech information, typical neighbouring words, appearance in topics, etc. Vice versa, these features can be used to verify discrepancies in the text and hence to detect speech recognition error regions and presence of OOV words. Thus features based on acoustic scores and confusion have been combined with language context features to detect OOV words. Some very earlier works in this category are those of [Suhm et al., 1993] and [Young, 1994], which used simple n-gram LM and semantic parsing techniques along with acoustic scores to detect OOV words.

Following the developments in natural language text processing, later works adopted approaches similar to text sequence tagging and classification to address the OOV detection problem. Apart from the basic acoustic and language model

scores, several new features were used. For instance context words, part-of-speech tags, features from the LVCSR decoding graph, word graphone disagreement, etc. Different classification methods were used to perform the OOV detection with these features, including boosting classification algorithm [Lecouteux et al., 2009], Conditional Random Fields (CRF) [Parada et al., 2010a], MaxEnt model [Kumar et al., 2012, Chen et al., 2013b]. Intensely trained systems involving slot level as well as sentence level classification along with syntactic parsing on word confusion networks from LVCSR were also proposed [Marin et al., 2012].

2.2.2 Vocabulary Selection Based Approaches

Detecting the presence of OOVs or finding the location of OOVs in speech suites a scenario when the OOV words are not known. In many LVCSR transcription tasks the majority of the OOV words could be known beforehand, either from the statistics of text data used for LM training, or from a domain/task specific corpus, or from the World Wide Web. With the knowledge of possible OOV words, methods can be developed to directly recognise or recover the OOV words, instead of detecting OOV regions first. For instance techniques like OOV word phone sequence matching or acoustic keyword search or a second pass LVCSR with updated vocabulary and language model can be used. Even the methods which first detect the OOV region and then perform OOV recovery also require selection of appropriate OOV words [Pan et al., 2005, Parada et al., 2010b, Oger et al., 2008b] for better recovery. Hence, with each of these methods it is crucial to balance the number of OOV words being searched, to avoid un-necessary false alarms. The underlying problem tackled by vocabulary selection based approaches is to infer a relevant vocabulary and/or reduced list of OOV words. We have grouped the related previous works under the following categories.

2.2.2.1 Vocabulary Selection to Reduce OOV Rate

There are several works in literature on selection of an optimal vocabulary for speech recognition. The main objective of these methods is to reduce mismatch between the trained LM and the utterances expected in the test speech while minimising the OOV rate. Some methods [Wang, 2003, Allauzen and Gauvain, 2005a] focus on learning a linear or vectorial combination of words in several existing corpora in order to re-estimate and interpolate the count of words in a domain/task specific corpus which ideally resembles the test speech data. New vocabulary is chosen based on the interpolated word counts or word weights. Neural network based methods have also been proposed [Liu et al., 2007, Jouvet and Langlois, 2013]. In these methods a neural network is trained on features

based on word frequencies, document frequency, TF-IDF and part-of-speech tag features, to infer word weights used for selection of speech recognition vocabulary.

Systems for daily update of LVCSR vocabulary have been proposed [Bertoldi and Federico, 2001, Federico and Bertoldi, 2001, Martins et al., 2006, Martins et al., 2007]. These systems uses news articles from the internet to update the LVCSR vocabulary. Given this new text data, the simpler scheme was to chose the new LVCSR vocabulary from the most recent and most frequent new words [Bertoldi and Federico, 2001, Federico and Bertoldi, 2001, Martins et al., 2006] or to use the earlier proposed linear interpolation schemes [Martins et al., 2007]. Furthermore, document specific vocabulary selection approaches have also been proposed. These methods adapt the vocabulary and language model for each test audio document. Allauzen and Gauvain [Allauzen and Gauvain, 2005b] used the video meta-data to add new words in the vocabulary and language model of the speech recognition system used for indexing videos. Meng et al. [Meng et al., 2010] used meta-data along with some selected phrases from the speech recognition hypothesis for querying the internet. They employed a neural network classifier using word frequencies, document frequency, TF-IDF and POS tag features for selection of new words. A similar approach was used for vocabulary selection for recognition of lectures, by using information from the lecture presentation slides [Maergner et al., 2012].

2.2.2.2 Querying the Internet for Recovery of OOV Words

Similar to the vocabulary selection techniques discussed in the previous section, these works also query the internet for relevant documents. They form queries based on the speech recognition hypothesis and then choose OOV words from the retrieved documents. The approaches mainly differ in techniques used to form search queries and/or the methods used to select the OOVs from the retrieved documents. TF-IDF based techniques were used to form search queries by [Parada et al., 2010b] and [Pan et al., 2005]. [Oger et al., 2008a] discussed different techniques for query formulation and later presented the results of integration of the chosen OOV words into speech recognition [Oger et al., 2008b, Oger et al., 2009]. The target OOV candidates were chosen from the retrieved documents using phone sequences observed in the identified OOV region [Pan et al., 2005, Parada et al., 2010b] or by using the words adjacent to the OOV region [Oger et al., 2008b].

2.2.2.3 Acoustic Search for OOV Words

When the known list of OOV words is small, acoustic keyword search techniques can be used to directly recover the OOV words. [Seneff, 2005] used a two pass strategy in which OOV candidates were filtered in the first pass by performing a phone search and the second pass performed LVCSR with OOV candidates included in the vocabulary. [Chen et al., 2013a] proposed WFST techniques to generate IV words and phone sequence proxies for each OOV word for searching the LVCSR lattice for OOVs. Similarly, [Karakos and Schwartz, 2014] presented a fuzzy phonetic search employing syllables and phone sequences of variable lengths for spotting OOV words. A combination of the proxy method and fuzzy-phonetic search was later shown to perform even better [Karakos and Schwartz, 2015].

2.3 Our Approach

Following the arguments and motivation presented in Section 1.2, in this dissertation we explore semantic and topic context models for handling the OOV problem in LVCSR systems. Fig. 2.3 shows a block diagram of our adopted approach. Text documents are collected from the web/internet to build a *diachronic text corpus* which contains documents with new i.e., OOV words. The diachronic text corpus is used to learn a context model which captures relationships between the LVCSR in-vocabulary words and the OOV words. This is the training or OOV learning phase. During normal operation, i.e. the test phase, speech is processed by the LVCSR system (with the base vocabulary and LM) to obtain the (first pass) LVCSR hypothesis. Given this text hypothesis and the context models, the context of the spoken content is inferred and a context based ranking is performed to choose only the relevant OOV words from the full list of OOV words. The list of relevant OOV words is then used to update the vocabulary and LM of the LVCSR to perform a second pass of LVCSR[6], which can now recognise the OOV words. Alternatively, an acoustic search can be performed for each of the relevant OOV words. However, unlike LVCSR decoding, the simpler acoustic search for OOV words is not constrained by the local n-gram word sequence (or grammatical order). For purpose of evaluation we perform a second pass of speech recognition using a simple language model update, given that updating the LVCSR language model is not in the scope of this dissertation.

For our study we have chosen the scenario of large vocabulary continuous

[6]Dynamically including words in the single pass of LVCSR decoding itself is possible and has been discussed in some works [Allauzen and Riley, 2015, Ma et al., 2015b], but this is a separate open problem and it is not in the scope of this dissertation.

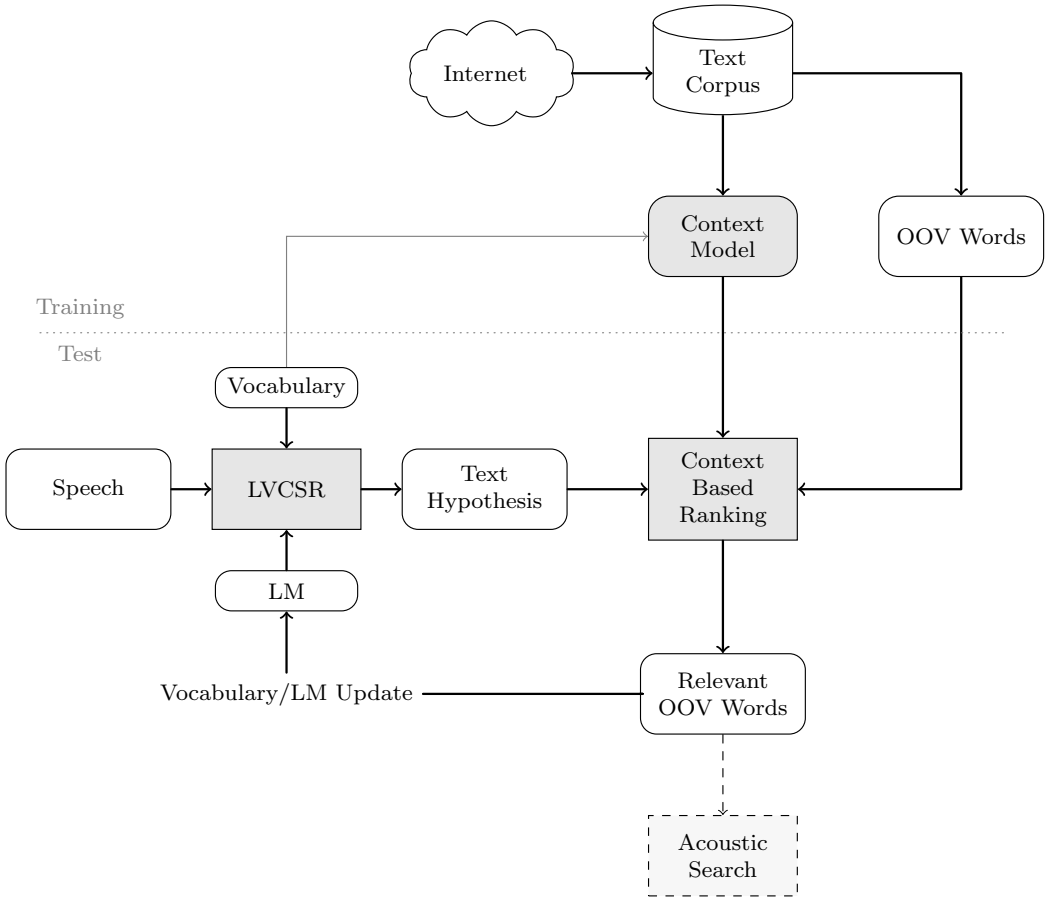


Figure 2.3: Block diagram of our approach for the recognition of Out-of-Vocabulary (OOV) words.

speech recognition systems applied to automatic transcription of French broadcast news videos. However, our proposed methods can be directly applied to LVCSR systems for other languages and domains. Moreover, as will be highlighted in Section 2.4.1 the majority of the OOV words in French broadcast news data are proper names or named entities. So we will focus mainly on proper name OOV words. But our methods do not depend on any specific linguistic or grammatical property of proper names and are readily applicable to other type of OOV words.

It must be noted that unlike the OOV detection methods, our context models are trained using only text data available from the web. Furthermore, our approach can be directly applied to process the hypothesis from LVCSR systems with sub-word units and hybrid language models, however this extension will not be studied in this dissertation.

2.3.1 Related Works

The idea of understanding and interpreting spoken utterances is not new. There is a fully fledged sub-field under spoken language research and development, generally referred as Spoken Language Understanding (SLU) [Mori et al., 2008, Tur and De Mori, 2011]. However it mainly deals with the problems of understanding and extracting the intent from spoken utterances, which is essential in tasks like spoken dialog systems for querying information, question answering systems and voice search.

Long span language models capturing semantic information [Bellegarda, 1999, Chien and Chueh, 2008, Bengio et al., 2003, Mikolov et al., 2010, Bayer and Riccardi, 2014] have shown success but it has been limited to generating better n-gram models and rescoring LVCSR n-best outputs. There have been several attempts on using long term context directly in LVCSR decoding [Chueh and Chien, 2009, Bayer and Riccardi, 2014]. Similarly, as opposed to separately modelling the speech recognition and the understanding modules, there have been efforts to improve the language model for both speech recognition and understanding [Riccardi and Gorin, 1998, Bayer and Riccardi, 2012]. Other related methods and techniques will be discussed alongside our proposed methods in the following chapters.

2.4 Task, Corpora and Transcription Systems

Section 2.3 presented a generalised description of our approach to handle OOV words. In this section we will present some specific details of our study, including a discussion on the task that will be focused on in this dissertation, as well as the evaluation measures, experiment corpora and the LVCSR systems used.

2.4.1 Diachronic News Corpora

Table 2.1 presents realistic diachronic news corpora which will be used as the training, validation and test datasets in our study. These corpora also highlight the motivation for handling OOV proper names in broadcast news transcription. The corpora are collected from two different sources: (a) website of the French newspaper *L'Express*[7], and (b) the French website[8] of the *Euronews* television

[7] http://www.lexpress.fr/
[8] http://fr.euronews.com/

Table 2.1: French broadcast news datasets used in experiments

	L'Express (train)	*Euronews* (validation)	*Euronews* (test)
Type of Documents	Text	Text	Video
Time Period	Jan - Jun 2014	Jan - Jun 2014	Jan - Jun 2014
Number of Documents[1]	45K	3.1K	3K
Vocabulary Size[2]	150K	42K	45K
Corpus Size (word count)	24M	550K	700K
Number of PN unigrams[2]	57K	12K	11K
Total PN count	1.45M	54K	42K
Number of OOV unigrams[3]	12.4K	4.9K	4.3K
Documents with OOV[3]	32.3K	2.25K	2.2K
Total OOV count[3]	141K	9.1K	8K
Number of OOV PN unigrams[3]	9.3K	3.4K	3.1K
Documents with OOV PN[3]	26.5K	1.9K	1.9K
Total OOV PN count[3]	107K	6.9K	6.2K

[1]K denotes *Thousand* and M denotes *Million*

[2] *L'Express* unigrams occurring less than 2 times are ignored

[3] *L'Express* unigrams occurring in less than 3 documents are ignored; documents with more than 20 and less than 500 terms

Note: OOV, OOV PN statistics are computed after term-document filtering

channel. The *L'Express* dataset contains text news whereas the Euronews dataset contains text news as well as news videos and their text transcriptions.

In our study the *L'Express* dataset will be used as the diachronic text corpus to train context/topic models in order to infer the OOV proper names relevant to Euronews videos which is our test set. *Euronews* text documents, denoted as 'validation' in the Table 2.1, will be used as a validation set in our experiments. To train the context models, the *L'Express* diachronic corpus vocabulary is lemmatised and filtered by removing proper names occurring only once, non proper name words occurring less than 4 times, and using a stop-list of common and non-content French words. Moreover, a POS based filter is employed to choose only words tagged as proper name, noun, adjective, verb and acronym. The filtered vocabulary has about 50K terms.

TreeTagger[9] [Schmid, 1994a] is used to automatically tag the proper names in the text. The words and proper names which occur in the vocabulary of our LVCSR system are tagged as in-vocabulary and the remaining words and proper names are tagged as out-of-vocabulary. The vocabulary of our LVCSR system was formed with the 122,000 most frequent words in a corpus comprising articles from the French newspaper *LeMonde* and the French Gigaword corpus. These two corpora containing data until the year 2008.

As shown in Table 2.1, 72% (3.1K out of 4.3K) of OOV words in the *Euronews* video dataset are proper names and about 64% (1.9K out of 3K) of the videos contain OOV proper names. An important statistic termed "*target OOV proper name coverage*" is not shown in Table 2.1. We use the term "target OOV proper name coverage" to refer to the percentage of OOV proper names in *Euronews* videos which can be recovered with the given diachronic corpus. For the *Euronews* videos the sum of the number of unique OOV proper names per video is 4694. Out of these 4694, up to 2010 i.e. 42% of the target OOV proper names can be recovered with the *L'Express* diachronic text corpus which introduces 9.3K new (OOV) proper names. So we say that the target OOV proper name coverage of *L'Express* diachronic text corpus is 42%.

The target OOV proper name coverage can be increased by augmenting text documents from additional news websites, as discussed in Section 4.6.3.2. For instance if additional news articles were collected from the website of the French news paper *Le Figaro*[10], and this for the same time period of January - June 2014, we can obtain a target OOV proper name coverage of 52%, but the total number of possible OOV proper names increase from 9.3K to 18.4K.

2.4.2 LVCSR and News Transcription Systems

Automatic broadcast news transcription systems use an LVCSR system with additional tools for pre-processing of the audio inputs and post-processing of the text outputs. Our broadcast news transcription system includes a set of tools that are trained to perform automatic segmentation of audio into different segments like music, jingles and speech from different genders and of different audio quality. Details on these pre-processors are available in [Illina et al., 2004]. Following this pre-processing step LVCSR is applied to the identified speech segments.

The LVCSR acoustic models are trained for speech-to-text transcription in the French language, with separate models to handle wide band (16 kHz) and narrow band (8kHz, telephone) quality audio. The transcription system LVCSR has a

[9]http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
[10]http://www.lefigaro.fr/

vocabulary based on articles that appeared in the French newspaper *LeMonde* and in the French Gigaword corpus, both until the year 2008. The LVCSR vocabulary contains the 122,000 most frequent words, which result into 260,000 entries in the pronunciation lexicon of the LVCSR system.

For our experiments, we will use two LVCSR systems which are based on different type of acoustic models and give different word error rates. Since our proposed methods infer the context of the spoken content from the LVCSR hypothesis, the LVCSR systems with different word error rates will demonstrate the robustness of our proposed methods under LVCSR errors. One is a GMM-HMM acoustic model based LVCSR system, which has a higher WER compared to the second LVCSR system based on DNN-HMM acoustic models. A brief description of the two systems follows.

2.4.2.1 Automatic News Transcription System (ANTS)

The ANTS [Illina et al., 2004] LVCSR system is based on context dependent GMM-HMM phone models trained on 200 hour broadcast news audio files. It uses the Hidden Markov Model toolkit (HTK) [Young et al., 2006] for training acoustic models and the Julius [Lee and Kawahara, 2009] speech recognition engine as the decoder backend. Using the SRILM language modelling toolkit [Stolcke, 2002], a 4-gram language model is estimated on text corpora of about 1800 million words. Automatic transcriptions obtained from ANTS have an average WER of 41.7% on the Euronews videos.

2.4.2.2 Kaldi Automatic Transcription System (KATS)

The KATS LVCSR system is based on context dependent DNN-HMM phone models trained on the same speech dataset used to train acoustic models for ANTS. It uses the Kaldi [Povey et al., 2011] backend for training acoustic models and for speech recognition decoding. A bi-gram language model is estimated on the same text corpora as that used for training ANTS language model. Automatic transcriptions obtained from KATS have an average WER of 16.4% on the Euronews videos.

2.4.3 Task Description and Evaluation Measures

As discussed earlier, the number of possible OOV proper names can be of the order of hundreds of thousands, depending on the mis-match from the LVCSR vocabulary. All these OOV proper names cannot be included in the vocabulary

and language model of the LVCSR system because (a) they are infrequent and not well represented in training data (b) it would increase the LVCSR search space and complexity, without guaranteeing correct recognition and on the contrary leading to false alarms which ultimately affect the recognition of in-vocabulary words. Even the acoustic search based recovery would face the problem of unnecessary false alarms. On the other hand, including OOV proper names with a non-optimal selection criterion may not show any improvements. Hence it is crucial to balance the number of OOV words included in the LVCSR. The main focus of this dissertation will be on learning context models which obtain OOV lists of highest relevance. This primary task will use its own evaluation measures, namely Recall and Precision. We further evaluate the effectiveness of the retrieved OOV list in terms of improvement in speech recognition and recovery of OOV words, as discussed in Chapter 6.

2.4.3.1 Primary Task: Retrieval of Relevant OOV PNs

Our primary task, to find the list of OOV proper names relevant to the spoken content, can be formulated as a retrieval task. Let us consider an automatic speech-to-text transcription setup with an LVCSR system having a base vocabulary $V = \{v_1, v_2, v_3, \cdots\}$, where v_i represents an in-vocabulary word. A set of OOV proper names, denoted as $\tilde{V} = \{\tilde{v}_k\}$, is obtained from a diachronic text corpus collected from the web. Given that in-vocabulary words and OOV proper names co-occur in the diachronic text corpus, a context model θ_C can be learnt to capture relationships and/or mapping between the in-vocabulary words and OOV proper names.

$$\theta_C \equiv f(V, \widetilde{V}) \tag{2.8}$$

In the simplest case θ_C could just capture mutual information between the words [Church and Hanks, 1990] or it could be a more complex model based on distributional semantics, as it will be discussed throughout this dissertation.

From a spoken utterance, the sequence of words hypothesised by LVCSR is denoted as $h = w_1, w_2, w_3, \cdots$ where $w_j \in V$ i.e. each word w in the LVCSR hypothesis h comes from the LVCSR base vocabulary V. Given the context model θ_C and the LVCSR hypothesis h, we can obtain a likelihood score s_k for each OOV proper name $\tilde{v}_k$ as:

$$\begin{aligned} s_k &= p(\tilde{v}_k \mid \theta_C, h) \\ &= p(\tilde{v}_k \mid \theta_C, w_1, w_2, w_3, \cdots) \end{aligned} \tag{2.9}$$

To retrieve OOV proper names we calculate s_k for each OOV proper name $\tilde{v}_k \in \tilde{V}$ and then use it as a score to rank OOV proper names relevant to h.

2.4.3.2 OOV PN Retrieval Performance Measures

To measure the performance of retrieval of relevant OOV proper names, we use measures based on *Recall* and *Mean Average Precision* (MAP) [Manning et al., 2008a], which are commonly used to evaluate information retrieval systems. As mentioned earlier, for a given audio document several OOV proper names can be relevant. The ones actually present in the audio are referred to as target OOV proper names. For our task we can calculate recall (R) as:

$$R = \frac{\text{\# of target OOV PNs retrieved}}{\text{\# total target OOV PNs}}$$

The MAP for a set of Q test (or validation) documents is calculated as:

$$MAP = \frac{\sum_{q=1}^{Q} \overline{P}(q)}{Q} \tag{2.10}$$

where $\overline{P}(q)$ is the average precision score for each test/validation document q. Given a ranked list of OOV proper names for q, the average precision $\overline{P}(q)$ is calculated as:

$$\overline{P}(q) = \frac{\sum_r P@r \;\; rel(r)}{\text{\# target OOV PNs in q}} \tag{2.11}$$

where $rel(r)$ is an indicator function whose value is 1 if the OOV proper name at rank r is a target OOV proper name and 0 otherwise. And $P@r$ is the precision at rank r, calculated for a given q as:

$$P@r = \frac{\text{\# of target OOV PNs retrieved until } r}{r}$$

The overall recall is not informative and a more useful metric would be:

$$R@N = \frac{\text{\# of target OOV PNs in top-}N\text{retrieved OOV PNs}}{\text{\# total target OOV PNs}}$$

which indicates how much of the recall is achieved with the top-N retrieved OOV proper names. This is because, after retrieval of the relevant OOV proper names, the top-N (relevant) OOV proper names are to be used for recovery/recognition of the target OOV proper names. Similarly, MAP@N gives useful insights on the retrieval results. In MAP@N, while calculating the average precision $\overline{P}(q)$ for a document q, the target OOV proper names not present in the top-N OOV proper name list get a precision score of zero.

For analysis of performance of retrieval of OOV proper names, and for the comparison of different context models, we plot a graph of recall and MAP for

the top-N retrieved OOV proper names. These recall and MAP curves present different information about the OOV proper retrieval results. To recover the target OOV proper names one can use an additional speech recognition pass ([Oger et al., 2008b]); or an acoustic spotting of the relevant proper names ([Parada et al., 2010b]). In each of these approaches, the retrieval ranks/scores may or may not be used. This is where plotting both the recall and MAP curves make a difference. The recall value at an *operating point* (N in the top-N choice) is not sensitive to the rank of the retrieved OOV proper names whereas the MAP value takes into account the retrieval ranks. As discussed later, in our experiments we choose an operating point of N=128 because around this point the MAP curves stop increasing and also this point roughly corresponds to about 1% of the total OOV proper names that we obtain in the diachronic text corpus used in our experiments. Thus for simple (non-detailed) comparison of two models, or model configurations, the MAP@128 (equivalently the maximum MAP) achieved by the model will be used.

The statistical significance of the difference between the (maximum) MAP values achieved by two models is measured using Student's paired t-test and randomisation test [Smucker et al., 2007]. The *null hypothesis* is that there is no difference between the two models and they produce identical retrieval results. The null hypothesis is rejected if the p-value is less than 0.05 for both the tests [Smucker et al., 2007]. For the randomisation test we generate 100,000 random permutations of the results of the two models under test.

CHAPTER 3

Topic and Semantic Context Models

Retrieval of relevant Out-of-Vocabulary (OOV) Proper Names (PNs) in LVCSR is the main goals of this dissertation. Our proposal is to model the semantic and topic context of the possible OOV proper names. If the semantic and topic context of the spoken content can be inferred, the relevant OOV proper names can be retrieved from the (long) list of possible OOV proper names. In this chapter we will present a background on existing topic and semantic context models proposed in the literature. These models are studied extensively in the field of Computational Linguistics and Natural Language Processing (NLP).

There have been attempts to use the rich representations learned by semantic and topic models in LVCSR, specifically to model long term context in language models [Bellegarda, 1999, Schwenk and Gauvain, 2002, Mikolov et al., 2010, Bayer and Riccardi, 2014] and in applications on classification of spoken queries and spoken documents [Tur and De Mori, 2011, Wintrode, 2011, Morchid et al., 2014]. However we want to study and evaluate these models for our task of OOV proper name retrieval and hence we dedicate a separate chapter to describe these models.

This chapter serves as introductory material to readers new to the area of modelling semantics and topics. It begins with a brief introduction to distributional semantics and vector space models. It then describes the most prominent models, including Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA), Continuous Bag-Of-Words (CBOW) and Skip-gram.

3.1 Semantic and Topic Space Representations

Semantic and topic spaces are in effect vector spaces that are used for computer (or machine) representation and interpretation of semantics and topics. The corresponding models and methods have a long history in the field of computational linguistics and natural language processing. A survey on this topic is available in [Turney and Pantel, 2010]. Almost every model is backed by the statistical

semantics hypothesis which states that: **statistical patterns of word usage, in pieces of text or around a word, can be used to describe the underlying semantics**. Turney and Pantel divide the statistical semantics hypothesis into specific cases: (a) bag of words hypothesis (b) the distributional hypothesis (c) the extended distributional hypothesis and (d) the latent relation hypothesis. While (a) seems related to semantics of documents or smaller contexts, and (b) to semantic representation of individual words, they are essentially two sides of the same coin, specifically when documents and contexts are treated as bag of words. In general, the term 'distributional hypothesis' encompasses both hypotheses and we will follow this terminology in this dissertation, except for the discussion in this section. (c) and (d) on the contrary, are concerned with semantic relations between pairs of words, for example *mason:stone* and *carpenter:wood*, and can extend to n-tuples of words. In accordance with the objectives of this dissertation (see Section 1.2) all discussed models will be based on (a) and (b), without focusing on local or relational semantics of names and entities.

3.1.1 Distributional Semantics

Distributional semantic models encompass the semantic and topic models based on **the bag of words hypothesis and the distributional hypothesis**. To develop a general understanding of these models we present an example in Figure 3.1. Figure 3.1 (a) shows a sample text corpus with six lines, where each line represents a context. As a generalisation context could be a document, paragraph, sentence or some other window of text. Each context is composed of words, and each word is an instance of a term in the vocabulary. Figure 3.1 (b) shows a term-context matrix, which is formed by counting the number of occurrences of each term in each context. It highlights the bag of words hypothesis which states that if contexts/documents have similar word counts then they tend to have similar meanings[1]. For instance we can see that the first and second contexts/documents are most relevant to each other. It can be verified that if we compute cosine similarity of the first column (or context vector) with the rest of them then the cosine similarity would be highest with the second column (or context vector). Similarly, it also highlights the distributional hypothesis which states that words that occur in similar contexts tend to have similar meanings[2]. And it can be verified that the vectors for the terms *vote* and *elect*, which share common semantics, are closest to each other.

[1]Bag of words, since it simply relies on word counts and ignores word sequence information
[2]*Distributional* signifies the distribution of words or word counts

Context 1: *Some People like to play and to watch cricket*
Context 2: *Other People like to watch and to play football*
Context 3: *Some People vote to elect Prime minister*
Context 4: *Other People vote to elect President*
Context 5: *Prime minister to watch cricket*
Context 6: *President to watch football*

(a) A sample text corpus.

	Context Index					
	1	2	3	4	5	6
and	1	1	0	0	0	0
cricket	1	0	0	0	1	0
elect	0	0	1	1	0	0
football	0	1	0	0	0	1
like	1	1	0	0	0	0
minister	0	0	1	0	1	0
Other	0	1	0	1	0	0
people	1	1	1	1	0	0
play	1	1	0	0	0	0
President	0	0	0	1	0	1
Prime	0	0	1	0	1	0
Some	1	0	1	0	0	0
to	2	2	1	1	1	1
vote	0	0	1	1	0	0
watch	1	1	0	0	1	1

(b) Term-Context co-occurrence matrix derived from the text corpus.

Figure 3.1: Example of statistical patterns of word usage

3.1.2 Distributional Modelling Approaches and Our Choice

Driven by the distributional hypothesis, the term-context matrix forms the starting point for machine interpretation and representation of semantics and topics. However as we move to real world problems and datasets it becomes apparent that its requires further processing, both linguistic as well as computational, to interpret and represent semantics and topics efficiently. Some of the common and popular processing steps are listed below with some examples.

- Linguistic processing
  - tokenisation
    e.g. *Prime_Minister* could be one term
  - text normalisation
    e.g. handling accented characters in French, like in *Égypte* v/s *Egypte*
  - annotation
    e.g. annotating capitonyms[3] like *Turkey* (the country) and *turkey* (the bird)
- Computational processing
  - type and scope of context
    e.g. topic models [Hofmann, 1999, Blei et al., 2003] and information retrieval applications commonly use complete documents as context
    e.g. some models use a pairwise word co-occurrence (square) matrix [Lund et al., 1995, Lund and Burgess, 1996]
  - weighting context matrix elements
    e.g. Term Frequency-Inverse Document Frequency (TF-IDF) variants [Manning et al., 2008b]
    e.g. Pointwise Mutual Information (PMI) variants [Church and Hanks, 1990, Turney and Pantel, 2010]
  - smoothing, sparsity and noise reduction
    e.g. matrix decomposition/factorisation [Deerwester et al., 1990, Lee and Seung, 1999]
    e.g. probabilistic smoothing [Hofmann, 1999, Blei et al., 2003]

[3] A word whose meaning changes based on whether it is capitalised or not.

Research in distributional semantics has led to several semantic word-context vector representations. The earliest success was seen with Latent Semantic Analysis (LSA) [Deerwester, 1988, Deerwester et al., 1990]. LSA derives a semantic vector space by applying truncated Singular Value Decomposition (SVD) on the word co-occurrence matrix. SVD reduces the dimensionality of the vector space, suppressing redundancy and sparsity, and results in a semantic space where semantically related documents/contexts are close to each other even if they do not have same set of terms. While the LSA remained prominent for many years, it faced scalability challenges and was criticised for its *uninterpretable* semantic space. A probabilistic version of LSA named Probabilistic Latent Semantic Analysis (PLSA) was later proposed [Hofmann, 1999], which expressed a document as a mixture of interpretable and non-orthogonal topics. But PLSA itself was soon succeeded by LDA, which learns probabilistic topic space representations with a Bayesian framework[4]. LDA has been shown to outperform PLSA and LSA for document classification [Blei et al., 2003] and word prediction [Griffiths et al., 2007] tasks. LDA, being a Bayesian network, has also been extended to included different attributes and variables [Blei, 2012]. The **prior evaluations and modelling capability motivate us to study LDA for learning context of OOV PNs**. On the other hand, LSA has been popular in information retrieval, so it is also evaluated alongside the LDA model. A description of LSA is presented in Section 3.2 and that of the LDA model follows in Section 3.3.

Recent developments in Neural Network based language models [Mikolov et al., 2013c] led to a renewed interest in the field of distributional semantics. More specifically in learning word embeddings: representation of words in a vector space describing syntactic and/or semantic properties. The most straightforward method to learn these representations is by predicting the word embedding using the context in which words appear [Mikolov et al., 2013b, Pennington et al., 2014], and this could be achieved with neural networks or matrix factorisation methods [Levy and Goldberg, 2014]. These representations were shown to perform effectively for a range of text processing tasks [Baroni et al., 2014]. Particularly, the Skip-gram and CBOW models of Mikolov et al. [Mikolov et al., 2013b, Mikolov et al., 2013a] have become very popular due to their ability to handle large amounts of unstructured text data with reduced computational costs. The **efficiency and semantic properties of the word embedding representations motivate us to explore word embeddings for our task** of OOV proper name retrieval and to compare its performance with that of LDA topic models. The CBOW and Skip-gram models are described in Section 3.4.

[4][Girolami and Kabán, 2003] showed that PLSI is a Maximum a Posteriori (MAP) estimated LDA model under a uniform Dirichlet prior and the shortcomings of PLSA can be resolved with the LDA framework.

3.2 Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA) [Deerwester et al., 1990] derives semantic representations of words and documents using linear algebra and matrix decomposition methods. The LSA model begins with a term-document co-occurrence matrix computed on a text corpus. This term-document matrix is similar to term-context matrix shown in Figure 3.1b. Each document is treated as context and the term-document matrix consists of Term Frequency-Inverse Document Frequency (TF-IDF) weights, corresponding to each word in each document, as its elements. Extending the discussion in Section 3.1.1, the columns of this matrix carry semantic information in documents and the rows carry semantic information in words. The dimensions of the document and word vectors are very high (of the size of the vocabulary and the number of documents in the corpus, respectively). In order to reduce the dimensionality of these vectors and to handle the sparsity and noise in the huge term-document matrix (X), a Singular Value Decomposition (SVD) is applied as:

$$X = U\Sigma V^{'} \tag{3.1}$$

where Σ is a diagonal matrix with the singular values of X; U and V are the left and right singular vectors for the corresponding singular values. Selecting the K largest singular values, and their corresponding singular vectors from U and V, we get the rank K approximation to X as:

$$X_K = U_K\Sigma_K V_K^{'} \tag{3.2}$$

U_K now contains the K dimensional semantic representation of each term and V_K now contains the K dimensional semantic representation of each document. This decomposition is depicted in Figure 3.2.

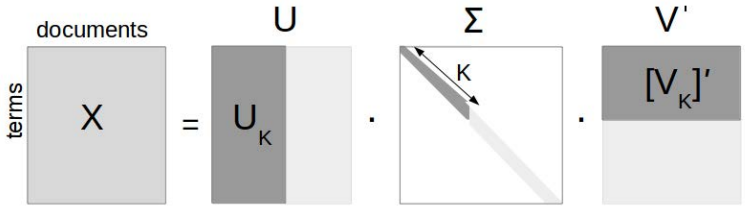


Figure 3.2: Matrix Decomposition in LSA.

Semantic space representation of a new unseen document h can be obtained as:

$$h_K = \Sigma_K^{-1} U_K^{'} h \tag{3.3}$$

3.3 Latent Dirichlet Allocation (LDA) Topic Model

Latent Dirichlet Allocation (LDA) [Blei et al., 2001, Blei et al., 2003] has been a prominent method for automatically learning underlying topic representations in text documents. LDA takes a generative probabilistic approach for modelling collections of text documents[5]. Each text document is described as a mixture of latent topics; for example the sentence "*With Olympics over, Haitian workers are leaving Brazil for the US.*" could be a mixture of about 70% economics, 20% politics and 10% sports. Each topic is a discrete distribution over the vocabulary. More specifically, each topic is a multinomial distribution of words and the mixture of topics in a document is also a multinomial distribution. Following a Bayesian framework each of these multinomial distributions have corresponding priors with a Dirichlet distribution, which give some control over the shape of the topic distributions.

For a better understanding we will describe the generative process of the model. Table 3.1 describes the list of mathematical symbols that will be used to formulate the LDA model. In addition, Figure 3.3 shows a plate diagram for the smoothed[6] LDA topic model. Given the notations in Table 3.1, the generative process for a collection (or a corpus) of text documents under the LDA model is as follows:

1. Draw T multinomials ϕ_z from a Dirichlet prior β, one for each topic z;
2. For each document d, draw a multinomial θ_d from a Dirichlet prior α;
3. For each of the N_d words w_i in document d;
   - $\rightarrow$ Draw a topic z_{di} from multinomial θ_d;
   - $\rightarrow$ Draw a word w_{di} from multinomial $\phi_{z_{di}}$

Given a corpus of text documents, the number of topics T to be modelled and choice of priors α and β, the LDA model can learn topic distributions in a unsupervised manner. This involves the procedure of *posterior inference*: learning the posterior distributions of the latent variables (θ, ϕ, z) of the LDA model.

[5]LDA can model any type of collections of grouped discrete data and has been tried for images [Fei-Fei and Perona, 2005], audio [Kim et al., 2009], social-network data [Cha and Cho, 2012], etc. It is interesting to note that the same model was also proposed independently in the study of population genetics [Pritchard et al., 2000].

[6]The Dirichlet prior α gives non-zero probability to words that do not appear in the training text and hence as in the original paper this version of the LDA model is also called smoothed LDA model. However, throughout this dissertation we will refer to the smoothed LDA model as simply the LDA model.

Table 3.1: Description of Symbols used for LDA Topic Model

Symbol	Description
T	number of topics
D	number of documents in the corpus
V	number of unique words in the corpus
N_d	number of word tokens in document d
θ, θ_d	the multinomial distribution of topics to documents (suffix d denotes to specific document $d = 1, 2, ..., D$)
ϕ, ϕ_z	the multinomial distribution of words (suffix z denotes specific topic $z = 1, 2, ..., T$)
z, z_{di}	topic assignment (suffix di denotes for the i^{th} token in document d)
w, w_{di}	token in a document (suffix di denotes the i^{th} token in document d)
α	Dirichlet prior to θ
β	Dirichlet prior to ϕ

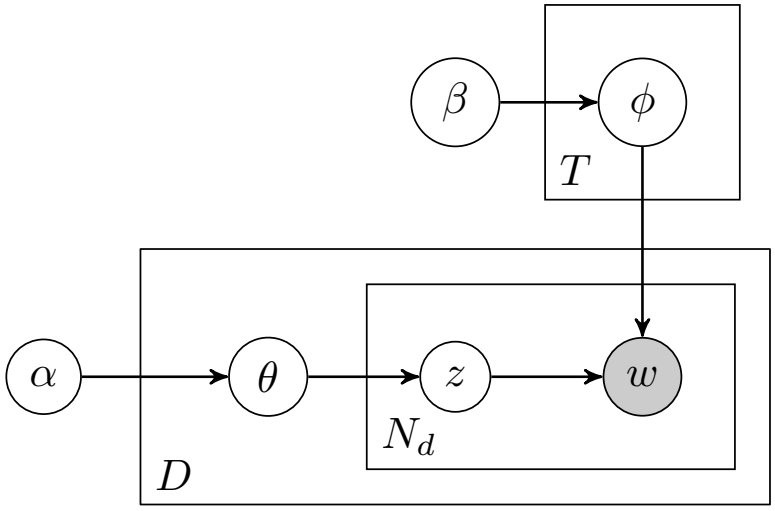


Figure 3.3: Plate Diagram for the LDA Topic Model. The circles, referred as nodes, represent variables of the model. The grey node represents observed variable (the words), whereas the other nodes represent hidden variables. The rectangles, referred as plates, represent replication and the replication count is shown at the bottom left corner of the plate.

3.3.1 Estimating LDA Model Parameters

Estimating these LDA model parameters requires to solve the equation:

$$p(\theta, \phi, z|w, \alpha, \beta) = \frac{p(\theta, \phi, z, w|\alpha, \beta)}{p(w|\alpha, \beta)} \tag{3.4}$$

The difficult part in this equation is the denominator which becomes computationally intractable, as it requires iteration over possible mixture of topics for every possible word. As a result, we need to use approximate inference techniques, such as mean-field variational expectation maximisation (as in the original work [Blei et al., 2001, Blei et al., 2003]), Gibbs Sampling [Griffiths and Steyvers, 2004] or expectation propagation [Minka and Lafferty, 2002]. While each of these methods have their own advantages, we adopt the Gibbs sampling method due to its simplicity. There are works in literature which present detailed discussion on Gibbs Sampling based estimation of LDA parameters [Griffiths and Steyvers, 2004, Heinrich, 2004, Carpenter, 2010]. For completeness we will present **a quick, but comprehensive, walkthrough of Gibbs Sampling estimation of LDA parameters, using a slightly different perspective**.

The LDA parameters to be estimated are θ, ϕ, z. We will start with an assumption that the topic assignment for each token in each document is known, and denoted as Z. Then it is straightforward to obtain the multinomial parameter sets θ and ϕ. According to their definitions as multinomial distributions with Dirichlet prior their posterior estimate turns out to be a Dirichlet distribution, as detailed in Appendix A.1 (Equation A.5).

$$\begin{aligned} p(\theta_d|Z, \alpha) &= \frac{1}{\Delta_{\theta_d}} \prod_{i=1}^{N_d} p(z_{di}|\theta_d)\ p(\theta_d|\alpha) \\ &= \text{Dirichlet}(\theta_d|\{n_{d,k} + \alpha\}_{k=1,2,\ldots,T}) \end{aligned} \tag{3.5}$$

$$\begin{aligned} p(\phi_k|Z, \beta) &= \frac{1}{\Delta_{\phi_k}} \prod_{v:z_v=k} p(v|\phi_k)\ p(\phi_k|\beta) \\ &= \text{Dirichlet}(\phi_k|\{n_{k,v} + \beta\}_{v=1,2,\ldots,V}) \end{aligned} \tag{3.6}$$

where, Δ are used to denote the corresponding normalising factors, $n_{d,k}$ is the count of tokens in document d which are assigned the topic k, $n_{k,v}$ is the number of times word v in the vocabulary is assigned the topic k. Using the expectation of the Dirichlet distribution, $\langle \text{Dirichlet}(\vec{a}) \rangle = a_i / \sum_i a_i$, we can estimate:

$$\begin{aligned} \theta_{d,k} &= \frac{n_{d,k} + \alpha}{\sum_{k=1}^{T} n_{d,k} + \alpha} \\ \phi_{k,v} &= \frac{n_{k,v} + \beta}{\sum_{v=1}^{V} n_{k,v} + \beta} \end{aligned} \tag{3.7}$$

Now the problem is to estimate the topic assignments $p(z|w, \alpha, \beta)$, which can be obtained by sampling from the joint distribution $p(z, w|\alpha, \beta)$, using Gibbs sampling. The Gibbs sampler samples the hidden variable (topic assignment) for each token in each document conditioned on all other hidden variables (topic assignments) sampled before it. If we denote this hidden variable as $z_{\mathfrak{i}}$ and use superscript $^{(-\mathfrak{i})}$ to denote leaving the $\mathfrak{i}^{\text{th}}$ token out of the calculation, then the Gibbs sampling equation would be:

$$\begin{aligned} p(z_{\mathfrak{i}}|z^{(-\mathfrak{i})}, w, \alpha, \beta) &= \frac{p(w, z|\alpha, \beta)}{p(w, z^{(-\mathfrak{i})}|\alpha, \beta)} \\ &\propto \frac{p(w, z|\alpha, \beta)}{p(w^{(-\mathfrak{i})}, z^{(-\mathfrak{i})}|\alpha, \beta)} \end{aligned} \tag{3.8}$$

The joint distribution expands as:

$$\begin{aligned} p(z, w|\alpha, \beta) &= \int\int p(\phi|\beta)\ p(\theta|\alpha)\ p(z|\theta)\ p(w|\phi_z)\ d\theta\ d\phi \\ &= \int p(z|\theta)p(\theta|\alpha)\ d\theta\ \int p(w|\phi_z)p(\phi|\beta)\ d\phi \end{aligned} \tag{3.9}$$

The first integral represents the Dirichlet-Multinomial distribution for the mixture of topics for each of the documents and the second integral represents the Dirichlet-Multinomial distribution of words in each topic. These integrals can be expressed in terms of the normalising constants of Dirichlet distributions of the posterior and prior, as discussed in Appendix A.1 Equation A.6. (The vector notation $\vec{}$ on symbols is skipped for simplicity.)

$$p(z, w|\alpha, \beta) = \prod_{d=1}^{D} \frac{B(n_{d,\cdot} + \alpha)}{B(\alpha)}\ \prod_{k=1}^{T} \frac{B(n_{k,\cdot} + \beta)}{B(\beta)} \tag{3.10}$$

$B(\alpha)$ is the multivariate Beta function, which can be expressed as: $B(\alpha) = \frac{\prod_{i=1}^{V}\Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{V}\alpha_i)}$, where $\Gamma(n) = (n-1)!$ for positive integer. $n_{d,\cdot}$ denotes topic specific word counts in document d and $n_{k,\cdot}$ denotes word counts for each topic k.

Using Equation 3.10 in Equation 3.8:

$$\begin{aligned} p(z_{\mathfrak{i}}|z^{(-\mathfrak{i})}, w, \alpha, \beta) &\propto \frac{\prod_{d=1}^{D} \frac{B(n_{d,\cdot}+\alpha)}{B(\alpha)}\ \prod_{k=1}^{T} \frac{B(n_{k,\cdot}+\beta)}{B(\beta)}}{\prod_{d=1}^{D} \frac{B(n_{d,\cdot}^{(-\mathfrak{i})}+\alpha)}{B(\alpha)}\ \prod_{k=1}^{T} \frac{B(n_{k,\cdot}^{(-\mathfrak{i})}+\beta)}{B(\beta)}} \\ &\propto \prod_{d=1}^{D} \frac{B(n_{d,\cdot} + \alpha)}{B(n_{d,\cdot}^{(-\mathfrak{i})} + \alpha)}\ \prod_{k=1}^{T} \frac{B(n_{k,\cdot} + \beta)}{B(n_{k,\cdot}^{(-\mathfrak{i})} + \beta)} \end{aligned} \tag{3.11}$$

Expressing the Beta functions in terms of the Gamma functions ($\Gamma(x)$) and substituting the identity $\Gamma(x+1) = x\Gamma(x)$ in the above equation, all the terms inside the two products cancel each other except the last two terms, resulting into:

$$\begin{aligned} p(z_{\mathrm{i}} = k | z^{(-\mathrm{i})}, w, \alpha, \beta) &\propto \frac{n_{d,k}^{(-\mathrm{i})} + \alpha}{\sum_{k'=1}^{T}(n_{d,k'}^{(-\mathrm{i})} + \alpha)} \; \frac{n_{k,v}^{(-\mathrm{i})} + \beta}{\sum_{v'=1}^{V}(n_{k,v'}^{(-\mathrm{i})} + \beta)} \\ &\propto (n_{d,k}^{(-\mathrm{i})} + \alpha) \; \frac{n_{k,v}^{(-\mathrm{i})} + \beta}{\sum_{v'=1}^{V}(n_{k,v'}^{(-\mathrm{i})} + \beta)} \end{aligned} \tag{3.12}$$

Using Equation 3.12, we can sample the topic assignments for each token in each document conditioned on all other topic assignments sampled previously, beginning with a random initialisation. Sampling the topic assignments for the entire corpus will complete one iteration. After a considerably large number of iterations, a stationary state of the above Markov chain has been reached, and hence the samples begin to converge to what would be sampled from the true distribution. A summary of the complete procedure for estimating the LDA model parameters using Gibbs sampling algorithm is present in Appendix A.2.

3.3.2 Topic Inference on New Documents

Once an LDA model is trained, it can be used to infer the topic mixture in new unseen documents. Similar to model parameter estimation, the latent topic mixture θ_h of a new document h can be inferred from the topic assignments for words in h. Let $c_{h,k}$ be the count of words, in the new document h, which are assigned the topic k. Then following the analogy to training set documents (Equation 3.7 and 3.5), we can estimate the topic mixture of h as

$$\theta_{h,k} = \frac{c_{h,k} + \alpha}{\sum_{k'=1}^{T}(c_{h,k'} + \alpha)} \tag{3.13}$$

The count $c_{h,k}$ depends on the topic assignments for words in h and similar to the training procedure the topic assignments $p(z = k|w, \alpha, \beta)$ will be inferred by sampling from the joint distribution $p(z, w|\alpha, \beta)$. The Gibbs sampling equation for this inference will use the global word-topic assignment counts ($n_{k,v}$) estimated during training. Let us denote $c_{k,v}$ as the number of times a vocabulary item v is present in h and assigned topic k. The updated Gibbs sampling equation for inference on test/new document is given as:

$$p(z_{\mathrm{i}} = k | z^{(-\mathrm{i})}, h, \alpha, \beta) \;\propto\; (c_{h,k}^{(-\mathrm{i})} + \alpha) \; \frac{c_{k,v}^{(-\mathrm{i})} + n_{k,v} + \beta}{\sum_{v'=1}^{V}(c_{k,v'}^{(-\mathrm{i})} + n_{k,v'} + \beta)} \tag{3.14}$$

3.4 Skip-gram, CBOW models to learn Word Embeddings

Word embeddings have been a recent trend in distributional semantics with the work of Mikolov et al. [Mikolov et al., 2013a, Mikolov et al., 2013b] being one of the most influential. They proposed two models: (a) the Continuous Bag Of Words (CBOW) model which predicts the center word given the surrounding context words, and (b) the Skip-gram model, trained with an objective function to maximise the likelihood of predicting the context words given the center word, where context refers to a window of words in a document. These models can be seen as single hidden layer neural network models without a non-linearity. It has been shown that word vectors with similar properties and performance can be obtained also using matrix factorisation methods [Levy and Goldberg, 2014, Pennington et al., 2014].

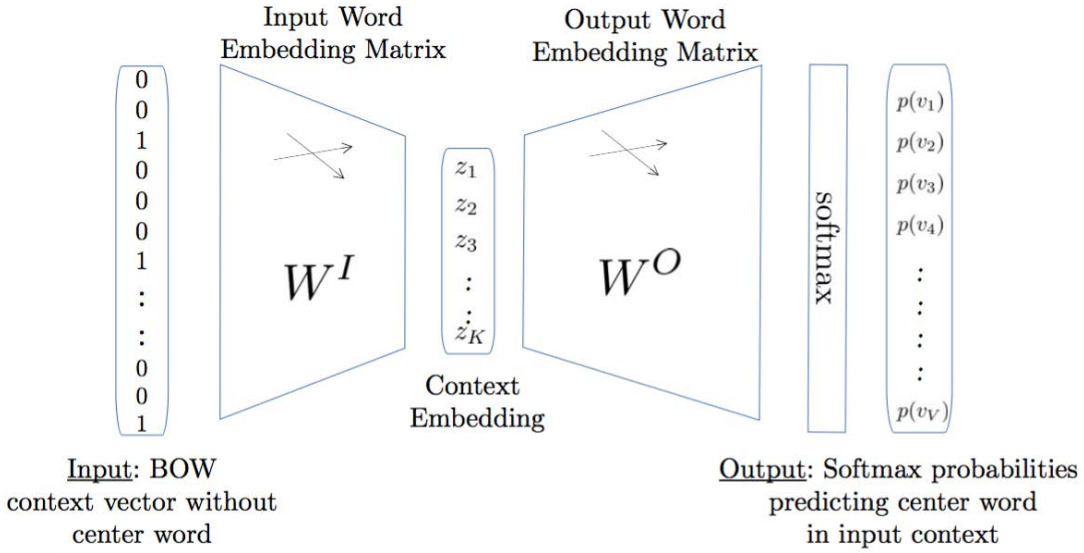
3.4.1 Model Descriptions

The CBOW and Skip-gram models have the same model architecture and they just differ in their functioning. We will describe these two models with the help of Figure 3.4, which gives an illustration of their architecture as well as their functioning.

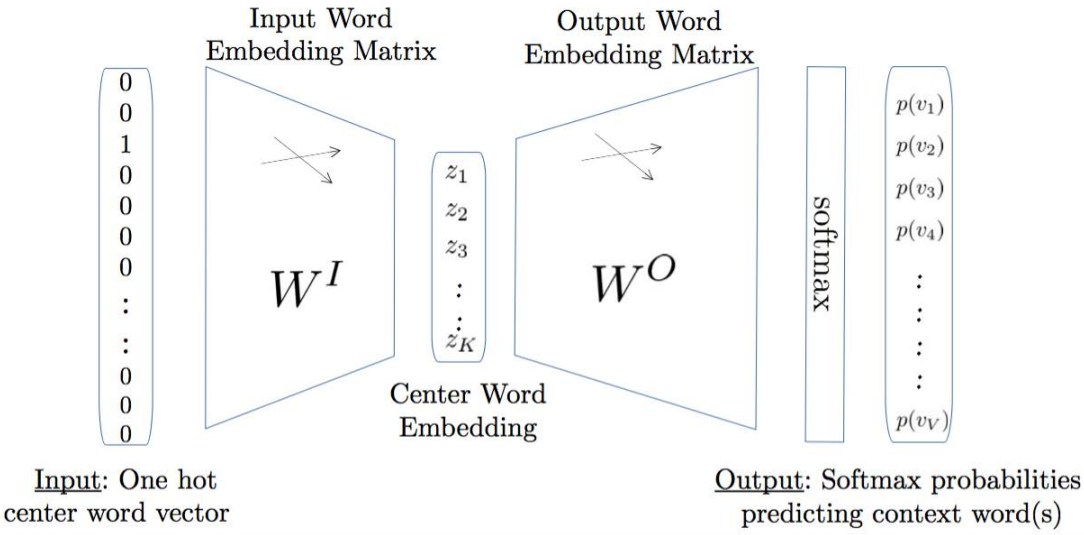
3.4.1.1 CBOW Model

As depicted in Figure 3.4 (a), the CBOW model takes as input a bag of word vector, of size V - the size of vocabulary, in which indexes corresponding to words present in the context are set to one and the remaining are set to zeros. This input vector is multiplied with the input embedding matrix W^I of size $[V \times K]$, where K is the embedding dimension, to obtain the context embedding z. As each row of the the input embedding matrix W^I corresponds to a word embedding, this is computationally equivalent to taking a sum of the embeddings of words in the context. Instead of a sum an average is used in practice. Denoting the bag of word input as $x_1, x_2, ..., x_V$:

$$\begin{aligned} z &= \frac{1}{|C|}(x_1\ x_2\ \cdots x_V)\ W^I \\ &= \frac{1}{|C|}\sum_{i=1}^{V} x_i W_i^I \\ &= \frac{1}{|C|}\sum_{i \in C} W_i^I \end{aligned} \tag{3.15}$$



(a) Continuous Bag Of Words Model



(b) Skip-gram Model

Figure 3.4: Architectures of CBOW and Skip-gram word embedding models

where C is the context (without the center word), $|C|$ denotes length (or number of words) of the context, W_i^I denotes input embedding for the i^{th} word in the vocabulary. The context embedding z is then multiplied to the output embedding matrix W^O of size $[K \times V]$ and followed by the softmax layer to predict the center word v_q in the context window. This prediction probability is calculated as:

$$\begin{aligned} p(v_q|C) &= \text{softmax}(z \cdot W_q^O) \\ &= \frac{exp(z \cdot W_q^O)}{\sum_{j=1}^{V} exp(z \cdot W_j^O)} \end{aligned} \tag{3.16}$$

where $(\cdot)$ denotes a vector matrix product.

3.4.1.2 Skip-gram Model

The Skip-gram model, depicted in Figure 3.4 (b), works in a reverse manner compared to the CBOW model. It takes as input a V dimensional one hot vector, with the one corresponding to the center word v_i. The multiplication of this one hot vector with the input embedding matrix W^I is equivalent to a simple lookup of the corresponding input word embedding W_i^I. This word embedding is then multiplied with the output embedding matrix W^O and followed by the softmax layer to predict the words surrounding the center word in the context window. In the actual implementation of the Skip-gram model one context word is predicted at a time instead of predicting all C context words simultaneously. Thus the probability of the q^{th} word from the vocabulary which is present in the current context window of the input center word v_i is calculated as:

$$\begin{aligned} p(v_q|v_i) &= \text{softmax}(W_i^I \cdot W_q^O) \\ &= \frac{W_i^I \cdot W_q^O}{\sum_{j=1}^{V} exp(W_i^I \cdot W_j^O)} \end{aligned} \tag{3.17}$$

3.4.2 Model Training

In both the CBOW and Skip-gram models, the input and output word embeddings matrices W^I, W^O are the model parameters which are to be trained and estimated from the training data. As with neural network models in general, the training of these model parameters is carried out using back propagation and stochastic gradient descent methods [LeCun et al., 1998]. Being an unsupervised learning method, the objective is to maximise the prediction log likelihood ($\mathfrak{L}$) over the training data. Denoting individual hidden units in the input and output matrix as ω, such that $\omega \in \{W_{jk}^I, W_{jk}^O\}_{k=1,2,3,\ldots,K\ ;\ j=1,2,3,\ldots,V}$, the parameter

update at the t-th iteration using stochastic gradient descent is performed as:

$$\begin{aligned} \omega_{t+1} &= \omega_t + \Delta\omega_t \\ &= \omega_t - \eta g_t \end{aligned} \tag{3.18}$$

where $\Delta\omega_t$ denotes the update in parameter ω, g_t denotes the the gradient of the parameters at the t-th iteration and η denotes the learning rate constant which controls how large of a step to take in the direction of the (negative) gradient. The gradient being calculated as $g_t = \frac{\partial \mathfrak{L}}{\partial \omega}$

Considering iterative training with one training sample at a time, and given Equation 3.15 and Equation 3.16, the loss function for the CBOW model is calculated as:

$$\begin{aligned} \mathfrak{L} &= -log\ p(v_{\mathrm{q}}\,|C) \\ &= -z \cdot W_i^O + log \sum_{j=1}^{V} exp(z \cdot W_j^O) \end{aligned} \tag{3.19}$$

Similarly, using Equation 3.17, the loss function for the Skip-gram model is:

$$\begin{aligned} \mathfrak{L} &= -log\ p(C|v_i) \\ &= -log \prod_{q \in C} \frac{W_i^I \cdot W_q^O}{\sum_{j=1}^{V} exp(W_i^I \cdot W_j^O)} \\ &= -\sum_{q \in C} W_i^I \cdot W_q^O + |C|\ log \sum_{j=1}^{V} exp(z \cdot W_j^O) \end{aligned} \tag{3.20}$$

The update equations are simple to derive further, as elaborated in [Rong, 2014].

3.4.2.1 Output Layer Optimization

In practise the vocabulary V can be very large. As a result, computation of softmax probabilities, each of which require a dot product with every word in the vocabulary, will be huge. This problem is also common to neural network based language models as discussed in [Morin and Bengio, 2005]. Additionally, the number of possible contexts can be as many as the total number of words in the entire text corpus. As a result evaluating Equation 3.19 for the entire corpus requires a tremendous amount of computations and hence training based on Equation 3.19 would be very slow. To address this problem Mikolov et.al [Mikolov et al., 2013a] proposed two alternate solutions, (a) hierarchical softmax and (b) negative sampling.

In the hierarchical softmax approach, a hierarchical tree is constructed at the output layer with vocabulary terms as the leaves. The tree is traversed by making decisions at each node, finally calculating the target word likelihood with the computational complexity reduced from $O(V)$ to $O(log(V))$ per training instance. It must be noted that this speedup also reflects in the back propagation. In the negative sampling approach a set of randomly sampled terms referred as negative samples are used instead of all the vocabulary terms in the denominator of the softmax function. For more details on these two approaches we refer the interested readers to [Rong, 2014, Goldberg and Levy, 2014].

CHAPTER 4

Retrieving OOV PNs with Topic Context

Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA) can obtain semantic and topic space representations of documents and words appearing in a text corpus. These models, described in Chapter 3, can thus provide context representations for our task of retrieving OOV proper names relevant to an LVCSR hypothesis. As the notion of context is very abstract we can exploit these representations in different ways. For a given OOV proper name we could imagine that it has a single global contextual image based on all its occurrences in the corpus, as also represented by the LDA and LSA models. At the same time, context of an OOV proper name can also come from the document containing this OOV proper name. In this chapter, we build on these ideas and propose two methodologies to retrieve context relevant OOV proper names.

LDA has been used to model proper names in [Senay et al., 2013]. A similar approach based on vector space representation similar to LSA has been tried in [Bigot et al., 2013]. However, these approaches estimate one LDA/LSA context model for each proper name, which restricts them to only frequent proper names, which have a significant amount of associated documents to learn individual LDA/LSA models. In our approach, we train a global topic model over all the diachronic text documents. As opposed to the usual practice of discarding less frequent terms in topic modelling, we need to retain the less frequent proper names both in the training and the test set.

We first introduce our retrieval methodologies using the topic space representations from LDA and then extend these to semantic vectors from the LSA model. We perform a detailed analysis and comparison of the performance of the two methodologies for the LSA and LDA model. We explore some re-ranking methods to further improve the performance of the LDA model and also study variants of the LDA model in which words and OOV proper names are modelled separately, but with a correlation between their corresponding topic spaces.

4.1 Proposed Retrieval Methodologies

To retrieve OOV proper names relevant to an LVCSR hypothesis, **our first methodology exploits the closeness of the context representation of the LVCSR hypothesis and that of the OOV proper name**. Models like LDA and LSA can be trained on the diachronic text corpus collected from the internet and they can learn the topic space representations of all the OOV proper names found in the diachronic text corpus. These models can also be used to infer the topic space representation of the LVCSR text hypothesis. Being a common representation space we can now measure the closeness of different OOV proper names to the LVCSR hypothesis, and hence use it to score the relevance of each OOV proper name. This methodology is illustrated in Figure 4.1.

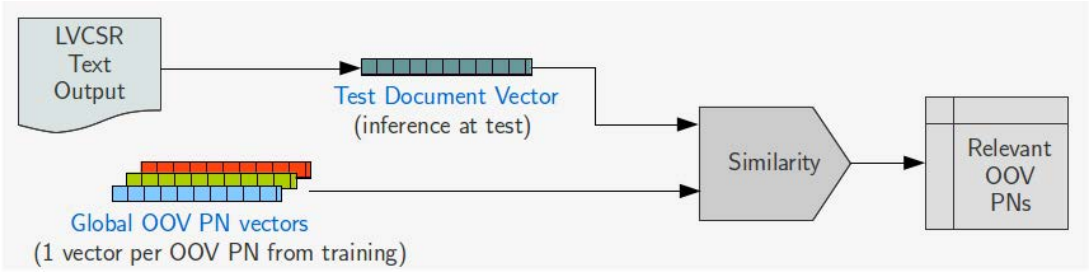


Figure 4.1: Retrieval of OOV PNs based on Closeness in the Context Space

We can also exploit the fact that a document in which an OOV proper name appears, can also represent the context of that OOV proper name. Following this idea, **our second methodology relies on document specific representations of OOV proper names**, instead of their global representations. Similar to our first methodology, models like LDA and LSA can be trained on the diachronic text corpus collected from the internet and a representation for each of its text document can be obtained. These document representations can now be used as context representations of the contained OOV proper names. This methodology is illustrated in Figure 4.2. We expect that the document specific representations will not only induce a document specific context but will also assist (a) OOV proper names which have only few training instances causing non-reliable global representations and (b) OOV proper names which have too many variations in context leading to averaged and sub-optimal global representations.

A more detailed description of the proposed retrieval methodologies is presented using the LDA topic model. The methodologies are then extended to the representations from the LSA model.

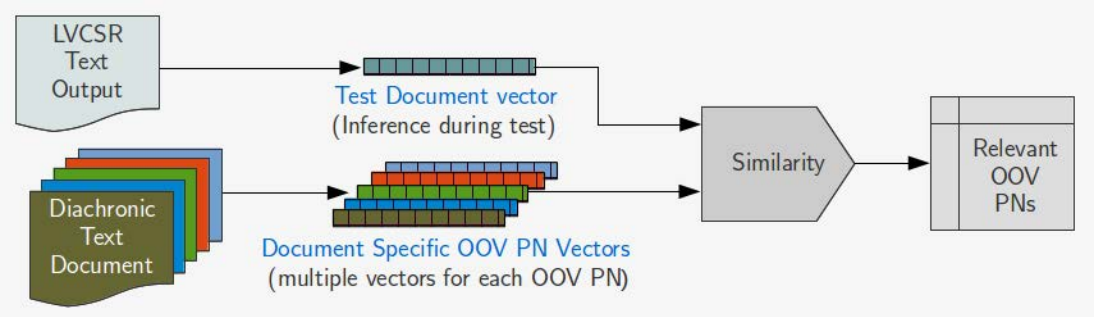


Figure 4.2: Retrieval of OOV PNs based on Document Specific Representations

4.1.1 OOV PN retrieval using LDA Topic Representations

Consider an LDA topic model trained on the diachronic text corpus containing D documents. The topic vocabulary size V, the number of topics T and Dirichlet priors α, β are chosen before training. During training, topic model parameters θ and ϕ are estimated using the procedure discussed in Algorithm 1 and Algorithm 2 in Appendix A.2. As discussed previously, $\theta = [\theta_{dt}]_{D\times T}$ is the mixture of topics in each document d in the diachronic text corpus and $\phi = [\phi_{vt}]_{V\times T}$ is the distribution of words in each topic, both sharing a topic space composed of T topics.

Let us denote the LVCSR word hypothesis by h and OOV proper names by $\tilde{v}_i$. As discussed in Section 3.3.2, Gibbs sampling can be used to infer the topic assignments for each of the words in the LVCSR hypothesis h and using these topic assignments the topic mixture $\{p(k|h)\}_{k=1,2,....,T}$ in h is calculated as:

$$p(k|h) = \frac{c_{h,k} + \alpha}{\sum_{k=1}^{T} c_{h,k} + \alpha} \tag{4.1}$$

where $c_{h,k}$ is the count of words, in the LVCSR hypothesis h, which are assigned the topic k.

4.1.1.1 LDA Method I: Closeness of LVCSR hypothesis and OOV PNs in LDA Topic Space

If the topic space representation of the LVCSR hypothesis is known, then it can be compared with the topic space representation of the OOV proper names. The probability of an OOV proper name given a particular topic, i.e. $p(\tilde{v}_i|k)$, can be obtained directly from ϕ. Finally, given the topic space probabilities of the LVCSR hypothesis h and that of an OOV proper name $\tilde{v}_i$, the likelihood of the

OOV PN $\tilde{v}_i$ can be calculated as:

$$p(\tilde{v}_i|h) = \sum_{k=1}^{T} p(\tilde{v}_i|k)\ p(k|h) \tag{4.2}$$

To perform retrieval of OOV proper names we calculate $p(\tilde{v}_i|h)$ for each OOV proper name $\tilde{v}_i$ and then use it as a score to rank OOV proper names relevant to LVCSR hypothesis h.

4.1.1.2 LDA Method II: Document Specific LDA Representations of OOV PNs

During training, the diachronic text corpus documents are indexed with each of the contained OOV proper names. The topic mixture for each of the documents in the diachronic corpus is obtained and stored as a prototype context vector for each of the OOV proper names in that document. OOV proper names occurring in more than one diachronic document will have multiple prototype context vectors or in other words the OOV proper names are characterised by multiple document specific context vectors. Multiple OOV proper names in a diachronic corpus document will share a common topic context vector.

To find the relevant OOV proper names, the T dimensional topic mixture vector of the LVCSR hypothesis of the audio document (h) is compared with the q context vectors (C_q^i) for each of the OOV proper name $\tilde{v}_i$ to calculate a score, using cosine similarity[1], as follows:

$$\begin{aligned} s_i &= \max_q \{\text{Cosine\_Similarity}(h, C_q^i)\} \\ &= \max_q \left\{ \frac{h.C_q^i}{\|h\| \left\|C_q^i\right\|} \right\} \\ &= \max_q \left\{ \frac{\sum_{k=1}^{T} h_k\ C_{qk}^i}{\sqrt{\sum_{k=1}^{T}(h_k)^2}\ \sqrt{\sum_{k=1}^{T}(C_{qk}^i)^2}} \right\} \end{aligned} \tag{4.3}$$

where s_i is the score to rank and retrieve OOV proper names relevant to h. h_k and C_{qk}^i denote the k^{th} topic component of h and C_q^i. It should be noted that this approach can be directly applied to vector representations of documents from any other model.

[1]Since the document representations are probability distributions, divergence based measures like Kullback-Leibler divergence, Jason-Shanon divergence and Hellinger distance can be used [David M. Blei, 2007, Niraula et al., 2013, Celikyilmaz et al., 2010, Krstovski et al., 2013]. However, from our initial experiments we found that cosine similarity gives the best overall performance for our task.

4.1.1.3 LDA Method III: Avoiding Topic Inference on LVCSR hypothesis

LDA Method I and Method II discussed above require to follow the inference procedure to obtain the topic mixture of the LVCSR hypothesis. Gibbs sampling based inference discussed in Section 3.3.2, as well as methods based on variational inference [Blei et al., 2003], would require the complete (or a significantly long) LVCSR hypothesis to infer its topic mixture reliably. As as alternative, we propose another method for retrieval of OOV proper names using LDA topic models. This method relies on associations between in-vocabulary words in the LVCSR hypothesis (h) and the OOV proper names to be retrieved. It does not require any inference of word-topic assignments and simply performs a lookup in the word-topic distributions ϕ. While this simplification skips the hierarchical generative process of LDA, it tries to exploit both association and separation of in-vocabulary words and OOV proper names in the topic space.

Denoting the words in LVCSR hypothesis h by $\{w_j\}_{j=1}^{N_{vh}}$, with N_{vh} being the number of words in h, the retrieval score for an OOV proper name $\tilde{v}_i$ can be calculated as:

$$\begin{aligned} p(\tilde{v}_i|h) &= p(\tilde{v}_i|\{w_j\}_{j=1}^{N_{vh}}) \\ &\approx \prod_{j=1}^{N_{vh}} p(\tilde{v}_i|w_j) \\ &\approx \prod_{j=1}^{N_{vh}} \sum_{k=1}^{T} p(\tilde{v}_i|k)\, p(k|w_j) \end{aligned} \tag{4.4}$$

where, both $p(\tilde{v}_i|k)$ and $p(k|w_j)$ can be obtained using word-topic distribution captured in ϕ.

Note that using a sum instead of the product, in Equation 4.4, will lead to a linear summation of the words in h. Such a linearity is exploited later with word embeddings (in Section 5.1), as well implicitly incorporated for LSA. When used with LDA it induces a bias towards frequent words leading to a lower performance of retrieval of relevant OOV proper names.

From Equation 4.4, we can also note that all of the elements in this scoring technique can be pre-computed just after the training phase. Thus this retrieval method can also work with online LVCSR decoding.

4.1.2 Extension of the Proposed Retrieval Methodologies to LSA

The proposed methodologies to retrieve relevant OOV proper names presented for the topic space representations from the LDA model, can be extended to the LSA

model. As mentioned before in Chapter 3, the LSA model was a pre-cursor to the LDA model. Similar to the LDA model it can learn a so called semantic vector representation for each OOV proper name ($\tilde{v}_i$) and it can also project the LVCSR hypothesis into this semantic space to obtain its semantic vector representation (h). Similarly during training it also generates the semantic vector representation of each document in the diachronic text corpus, which could be used as context vectors (C^i) for each OOV proper name ($\tilde{v}_i$).

Give the K dimensional semantic space representations based on LSA, the scoring function s_i for the OOV proper name retrieval methods are given as:

$$\begin{aligned} \text{LSA Method I:} \quad s_i &= \text{Cosine\_Similarity}(h, \tilde{v}_i) \\ &= \frac{h.\tilde{v}_i}{\|h\| \, \|\tilde{v}_i\|} \\ &= \frac{\sum_{k=1}^{K} h_k \, \tilde{v}_{ik}}{\sqrt{\sum_{k=1}^{K}(h_k)^2} \, \sqrt{\sum_{k=1}^{K}(\tilde{v}_{ik})^2}} \end{aligned} \tag{4.5}$$

$$\begin{aligned} \text{LSA Method II:} \quad s_i &= \max_q \{\text{Cosine\_Similarity}(h, C_q^i)\} \\ &= \max_q \left\{ \frac{h.C_q^i}{\|h\| \, \left\|C_q^i\right\|} \right\} \\ &= \max_q \left\{ \frac{\sum_{k=1}^{K} h_k \, C_{qk}^i}{\sqrt{\sum_{k=1}^{K}(h_k)^2} \, \sqrt{\sum_{k=1}^{K}(C_{qk}^i)^2}} \right\} \end{aligned} \tag{4.6}$$

where (C_q^i) are context vectors for each of the OOV proper name $\tilde{v}_i$. h_k, $\tilde{v}_{ik}$ and C_{qk}^i denote the k^{th} vector component of h, $\tilde{v}_i$ and C_q^i.

4.2 Evaluation of the Proposed Retrieval Methods

In this section we present an evaluation of our proposed methodologies, using LDA and LSA representations to retrieve relevant OOV proper names. The corpus setup and the retrieval evaluation measures have been presented in Section 2.4. We will first introduce the baseline methods for these experiments. Then we will present a discussion on selection of hyper-parameters for learning the LDA topic models. These will be followed by a comparison of the retrieval results achieved with the best model configurations.

4.2.1 Baseline Methods

In this section we present models and methods, for comparison with our proposed context based retrieval methods.

4.2.1.1 Pointwise Mutual Information (PMI)

Pointwise Mutual Information (PMI) is used as a measure of association in statistics and information theory. In computational linguistics, PMI has been used for finding collocations and associations between words [Church and Hanks, 1990]. We use it to measure the associations between OOV proper names and the in-vocabulary words. Denoting v_x and v_y as any two terms co-occurring in a document in the diachronic text corpus, the PMI is calculated as[2]:

$$pmi(v_x, v_y) = log \frac{p(v_x, v_y)}{p(v_x)p(v_y)} \tag{4.7}$$

where $p(v_x, v_y)$ denotes the probability of co-occurrence of the terms v_x and v_y in a document, $p(v_x)$ and $p(v_y)$ denote the probabilities of occurrence of terms v_x and v_y respectively, throughout the corpus. Given an LVCSR hypothesis h containing words $\{w_1, w_2, w_3, ...\}$, the score for retrieval of each OOV PN $\tilde{v}_i$ is calculated as:

$$s(\tilde{v}_i) = \sum_{j=1}^{|h|} log \frac{p(\tilde{v}_i, w_j)}{p(\tilde{v}_i)\ p(w_j)} \tag{4.8}$$

The PMI based method does not explicitly model any semantic or topic information. Due to its similarity with Method I, it is treated as the baseline equivalent for Method I for LDA and LSA, as well as for LDA Method III.

4.2.1.2 Random Projections (RP)

It is classical to represent text documents as vector of Term Frequency-Inverse Document Frequency (TF-IDF) values of the words in the vocabulary. Bingham and Mannila [Bingham and Mannila, 2001] showed that Random Projections (RP) can efficiently reduce the dimensionality of term frequency vectors while still preserving their original similarities and distances. With random projection, the V-dimensional TF-IDF vectors of D diachronic text corpus documents, denoted as $X_{D\times V}$), are projected to a K-dimensional ($K << N$) subspace through the

[2]Other normalised variants of PMI have been proposed [Bouma, 2009, Turney and Pantel, 2010, Role and Nadif, 2011, Damani, 2013], however in our initial experiments the improvements with these variants were not significant.

origin as: $X^{RP}_{K\times D} = R_{K\times D}X_{D\times V}$, where $R_{K\times D}$ is a random projection matrix with random unit vectors. In our experiments R is chosen as in the original work [Bingham and Mannila, 2001] with the elements of R being:

$$r_{ij} = \sqrt{3} \cdot \begin{cases} +1 & \text{with probability } \frac{1}{6} \\ 0 & \text{with probability } \frac{2}{3} \\ -1 & \text{with probability } \frac{1}{6} \end{cases} \tag{4.9}$$

The random projection of the LVCSR hypothesis can be obtained by multiplication with the random projection matrix R. During test the K dimensional vector representations of the diachronic text corpus documents and that of the LVCSR hypothesis are available and can be used to retrieve relevant OOV proper names using the Method II described in Section 4.1.1.2. This random projection based method does not explicitly model any semantic or topic information. It can be treated as the baseline for Method II.

4.2.2 Selection of LDA Model Hyper-parameters

The role of hyper-parameters is mostly known beforehand in a well defined model like LDA. But an exploration of hyper-parameter values enables to obtain the best model performances. The LDA model has three hyper-parameters (a) α the Dirichlet prior for document-topic distributions, (b) β the Dirichlet prior for topic-word distributions, and (c) T the number of topics which is also the size of the word and document topic vectors. In general bigger topic size is better for larger amounts of text data and with higher topic variability. The priors α and β are like smoothing hyper-parameters and they also control the nature of the topic distributions[3]. There are works in literature [Griffiths and Steyvers, 2004, Wallach et al., 2009] which discuss about selection of the LDA hyper-parameters and they are generally based on the log probability achieved by the model on a held out dataset. Following the common practice, we explore only symmetric Dirichlet priors with values less than 1, for our task. In this case the priors act more as smoothing and regularisation hyper-parameters. We will select the final hyper-parameters based on the maximum MAP for OOV proper names achieved on our validation set.

4.2.2.1 Method I and LDA Hyper-parameters

Figure 4.3 shows a chart depicting the variations in the maximum MAP values obtained on the validation set, using Method I, for a range of values for LDA

[3]roughly stating, the peakiness and flatness

hyper-parameters α, β and T. With the hyper-parameter values shown, the maximum MAP varies between 0.229 and 0.370. Beyond these set of values we observed degradation, or the improvement is statistically insignificant. Each of the LDA models were trained with 1500 iterations of Gibbs Sampling over the *L'Express* diachronic text corpus.

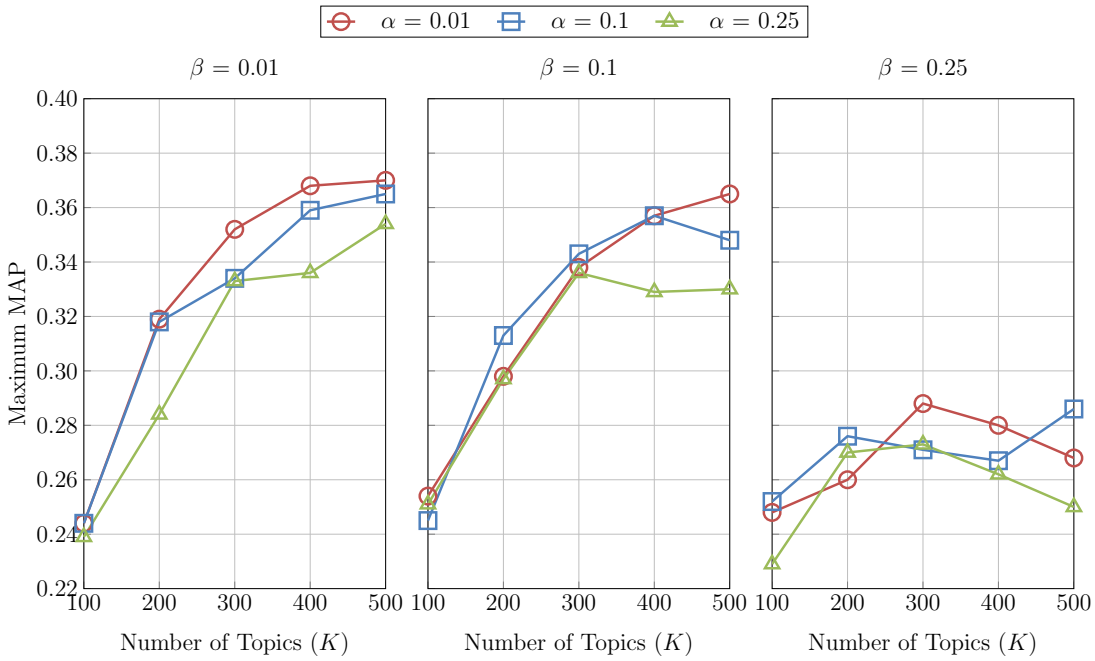


Figure 4.3: Variation in maximum MAP of retrieval of OOV PNs using Method I, with different topic sizes K and Dirichlet priors α, β for LDA. Evaluated on the validation set.

We can observe an increase in maximum MAP performance until 500 topics, beyond which the improvement is not significant with the *L'Express* diachronic text corpus. Based on the maximum MAP obtained on the validation set 400 LDA topics seem to be the best for Method I.

The priors in the LDA model gave better performance when they were set to smaller values, with $\alpha = 0.01$ and $\beta = 0.01$ being the values for the best performing LDA model with Method I.

4.2.2.2 Method II and LDA Hyper-parameters

Figure 4.4 shows a chart depicting the variations in the maximum MAP values obtained on the validation set, using Method II, for a range of values for LDA

hyper-parameters α, β and T. With the hyper-parameter values shown, the maximum MAP varies between 0.244 and 0.394. There is an increase in maximum MAP performance until 500 topics and beyond this the improvement is not significant. Similar to Method I, the maximum MAP obtained with 400 LDA topics seem to be the best for Method II. However, priors $\alpha = 0.1$ and $\beta = 0.01$ gave the best performing LDA model with Method II. Since Method II relies on document topic distributions, a slightly higher value of document prior α seems more suitable for Method II. Overall the MAP values for the validation set are similar to those from Method I, except around the best configuration of Method II, which performs better than the best model of Method I.

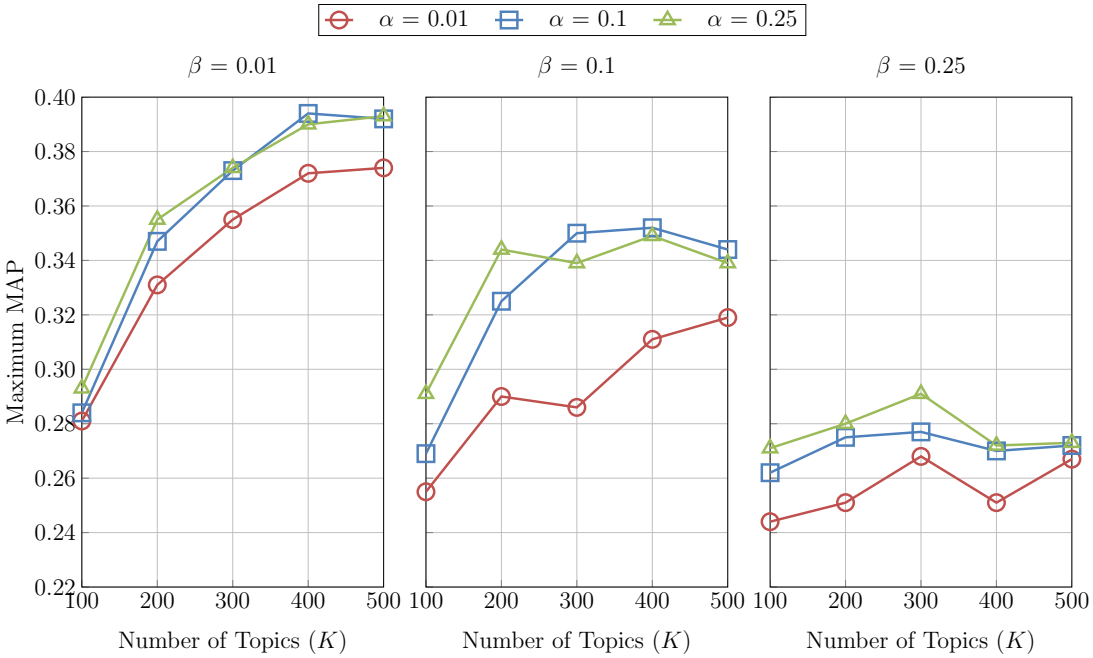


Figure 4.4: Variation in maximum MAP of retrieval of OOV PNs, with different topic sizes K and Dirichlet priors α, β for LDA Method II. Evaluated on the validation set.

4.2.2.3 Method III and LDA Hyper-parameters

Figure 4.5 shows a chart depicting the variations in the maximum MAP values obtained on the validation set, using Method III, for a range of values for LDA hyper-parameters α, β and T. With the hyper-parameter values shown, the maximum MAP varies between 0.172 and 0.313, which is quite low compared to Methods I and II. However unlike Method I and Method II, Method

III gives instant retrieval results without the delay in topic inference on LVCSR hypothesis. Similar to Method I, the maximum MAP is obtained with 400 LDA topics. Similarly, even the prior values $\alpha = 0.01$ and $\beta = 0.01$ give the best MAP performance for Method III, as the case with Method I.

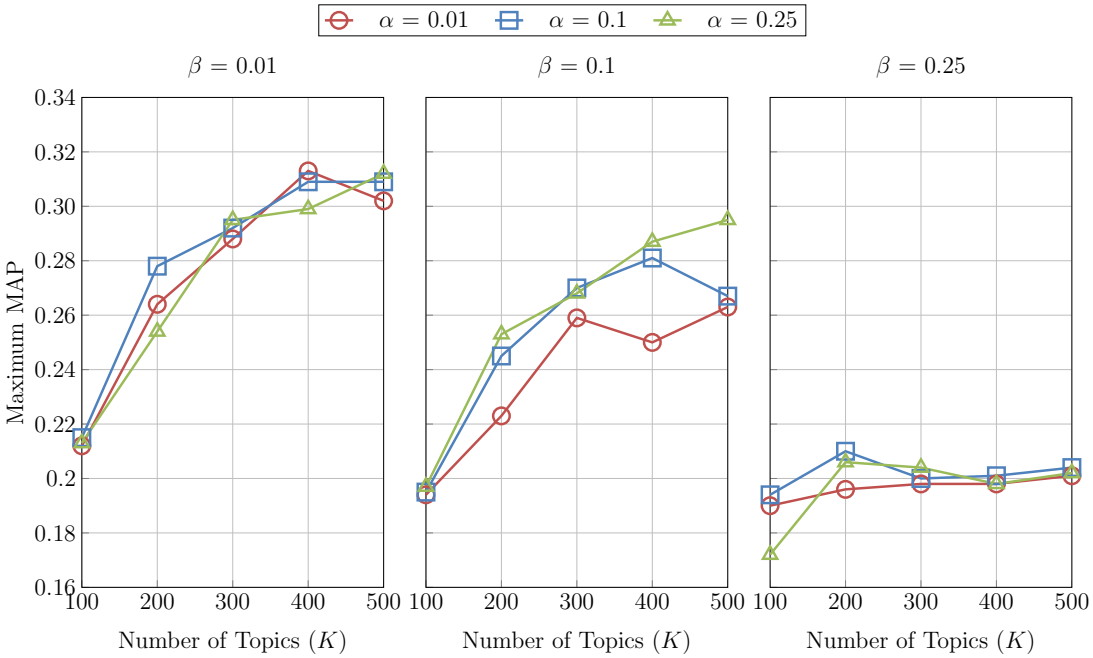


Figure 4.5: Variation in maximum MAP of retrieval of OOV PNs, with different topic sizes K and Dirichlet priors α, β for LDA Method III. Evaluated on the validation set.

4.2.3 Retrieval results achieved with the best model configurations

Figure 4.6 shows the recall and MAP performance obtained with the different models and methods. The figure shows the recall and MAP performance obtained on the reference transcription of the *Euronews* audio test set (on the left) as well as automatic transcriptions obtained from the ANTS LVCSR system (in the middle) and KATS LVCSR system (on the right), which were presented in Section 2.4.2. In comparison to the LDA model, Figure 4.6 shows the performance of the baseline PMI method, 400 dimensional Random Projections, the LSA model with 400 dimensional semantic space representations. Further it also includes the recall performance that would be achieved by simply selecting OOV proper names

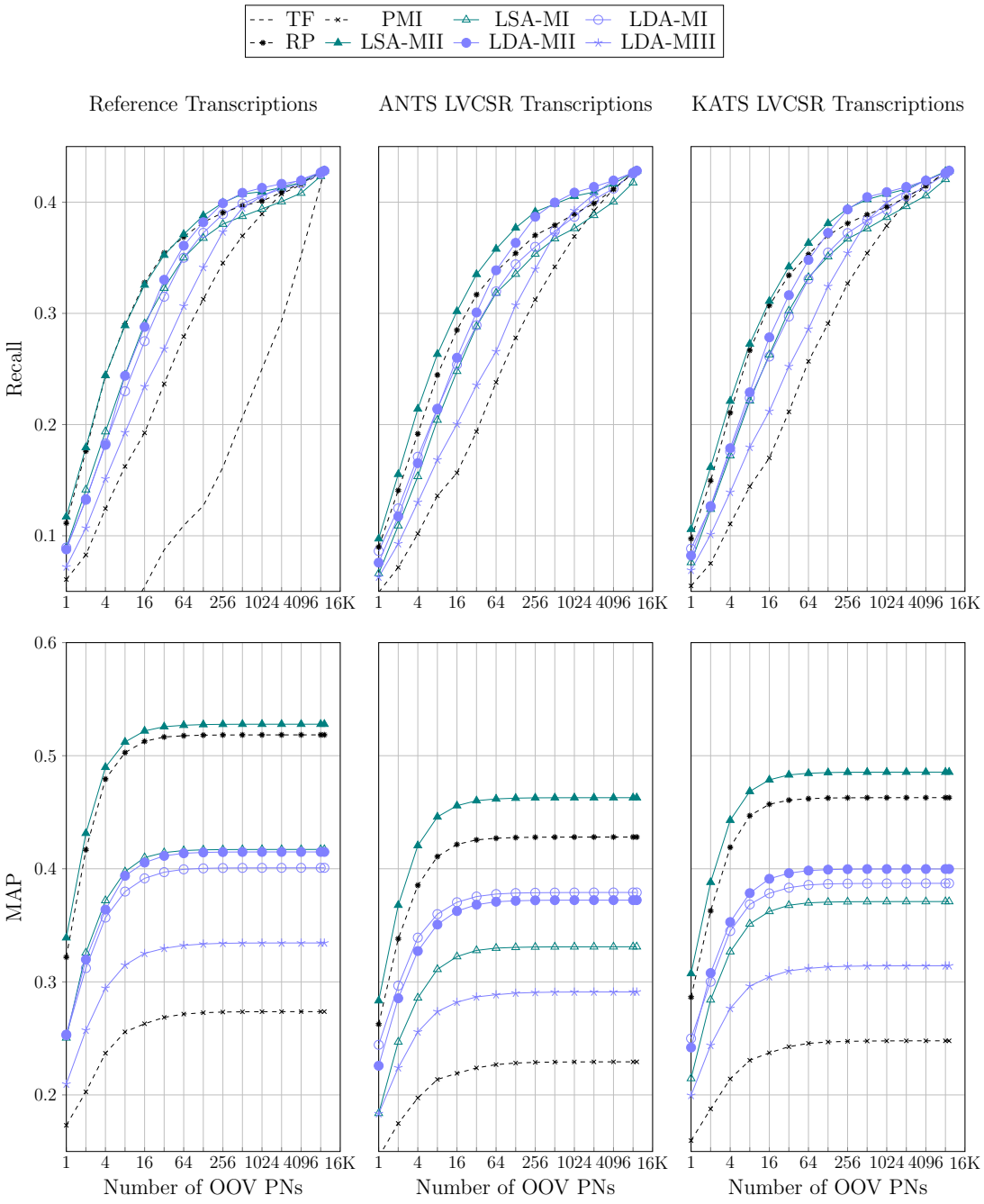


Figure 4.6: Recall and MAP performance of OOV PN retrieval, using LSA and LDA representation, evaluated on the *Euronews* audio test set. (TF, stands for term frequency, depicts MAP performance for a simple selection of top-k most frequent OOV PNs.)

based on their frequency of occurrence in the diachronic text corpus (denoted as TF for term frequency).

The number of possible OOV proper names that can be recovered with the six months *L'Express* corpus being used as the diachronic text corpus limits the recall performance to 0.42. The number of OOV proper names and the recall limit can be increased by using diachronic text from additional sources and/or additional period, as it will be discussed in Section 4.6.3.2.

Some of the recall curves appear closer to each other and for higher values on the X-axis, denoting the top-N retrieved OOV proper names, the recall curves merge with each other. However, the difference in performance is clearly identified by the MAP curves. MAP takes into account the ranks assigned to the target OOV proper names and thus indicates that the model which gave better ranks (closer to 1) to the target OOV proper names are more better. Eventually a system with better MAP will perform better when the number of possible OOV proper names are large, as discussed in Section 4.6.3.2. Moreover this is also important for acoustic search based audio indexing systems which would rely on context based models to automatically form a keyword list for searching, because a longer keyword list would add to false alarms and confusions.

The MAP@128 (maximum MAP) for the different models and methods presented in this chapter are presented in Table 4.1. Comparing the performances in Table 4.1 and Figure 4.6, we can make the following observations:

- Document specific representation of OOV proper names, specifically using Random projection (RP) and LSA (LSA-MII), achieves the best retrieval results. While the document specific representation of OOV proper names in Method II performs better than the global representation of OOV proper names, it must be noted that this improvement comes at extra computation cost, because Method II is equivalent to comparing the LVCSR hypothesis with each document in the diachronic text corpus. This could be a problem when the diachronic text corpus is extended further to improve coverage of target OOV proper names.

- Simple word association based methods, whether based on pointwise mutual information (PMI) or on LDA (LDA-MIII), give the lowest recall and MAP. However, it must be noted that these methods are computationally equivalent to performing a lookup from the word-OOV proper name association matrix and they retrieve the relevant OOV proper name list instantly and without requiring the whole LVCSR hypothesis to be available. This makes them suitable for use with online LVCSR decoding.

- LDA based retrieval methods (especially LDA-MI and LDA-MII) are more robust to LVCSR errors. As it can be seen for each of the LDA methods, the difference between the MAP values on reference transcriptions and LVCSR transcriptions is quite small compared to that for other models and methods. For instance LSA-MII and RP, which achieve the best MAP, show a relative drop of 12.1% and 17.3% respectively between reference and ANTS transcriptions, whereas LDA-MII and LDA-MI give a drop of 10.1% and 5.25% respectively.

- While the document specific representation method of LSA (LSA-MII) outperforms that of LDA (LDA-MII), it must be noted that the topic space representation of OOV proper names learned by LDA is more robust than the semantic space representation learned by LSA. This can be observed from the MAP of Method I (LDA-MI versus LSA-MI) for the LVCSR transcriptions.

Table 4.1: Comparison of MAP@128 for PMI, RP, LSA and LDA models. (The best model is highlighted in bold. * denotes statistically insignificant difference compared to the best model.)

	Reference Transcription	ANTS Transcription	KATS Transcription
PMI	0.273	0.229	0.247
RP	0.518*	0.428	0.462
LSA-MI	0.417	0.331	0.371
LSA-MII	**0.527**	**0.462**	**0.485**
LDA-MI	0.400	0.379	0.387
LDA-MII	0.414	0.372	0.399
LDA-MIII	0.334	0.291	0.314

4.3 Frequent versus Rare OOV PNs

Some OOV proper names in the diachronic corpus may belong to popular and bursty news events making them relatively frequent OOV proper names. On the other hand, many other OOV proper names are not so frequent or rare in the diachronic text corpus. This behaviour, similar to the Zipf's law on frequency of

occurrence of words [Powers, 1998], is depicted by the chart in Figure 4.7. This chart shows the distribution of frequency of OOV proper names in the *L'Express* diachronic text corpus. The Y-axis shows bins corresponding to frequency of OOV proper names in the diachronic text corpus and X-axis shows the count of OOV proper names in a particular frequency bin.

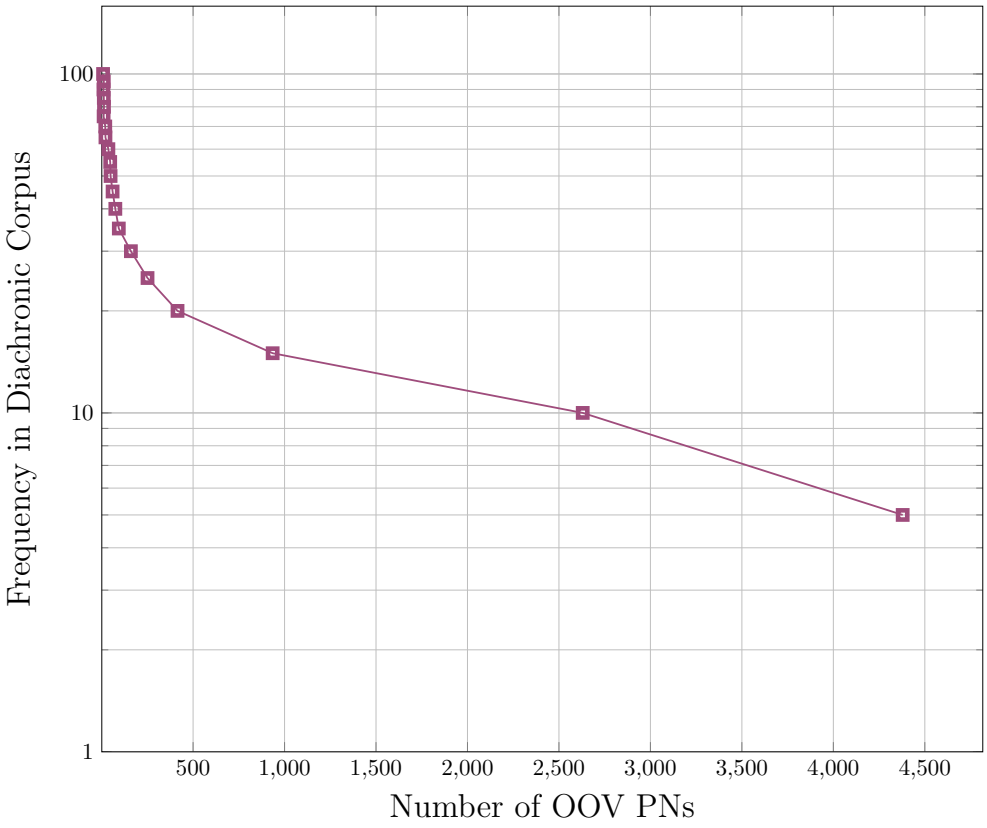


Figure 4.7: Distribution of frequency of OOV PNs in the *L'Express* diachronic text corpus.)

As the less frequent (or rare) OOV proper names do not have many training instances and/or documents, we believe that they do not have good co-occurrence statistics and hence have non-reliable topic/semantic space representations. So we hypothesise that they do not achieve good retrieval ranks. In our test set of *Euronews* videos about 24% (479 out of the possible 2010) OOV proper names appear in 10 or fewer documents in the *L'Express* train corpus. This motivates us to specifically study the behaviour of retrieval of rare OOV proper names.

In this section, we will analyse the performance of LDA and LSA models along our hypothesis - **'bias against rare OOV proper names'**. To test our hypoth-

esis will use (a) a rank-frequency distribution plot, which shows the distribution of ranks assigned to OOV proper names versus their frequency of occurrence in the diachronic text corpus, and (b) MAP for rare OOV proper names, which is simply the MAP calculated only for OOV proper names appearing 20 or fewer times in the diachronic text corpus.

4.3.1 Effects in LDA

LDA Method I and Method III rely directly on the OOV proper name topic probabilities $p(\tilde{v}_i|t)$ (see Equation (4.2) and Equation (4.4)). For rare OOV proper names these probabilities would be low and hence they would be achieving lower retrieval ranks with Method I and Method III. To validate this hypothesis we plotted the rank-frequency distribution for retrieval ranks obtained with LDA Method I, II and III. This plot is shown in Figure 4.8. As evident from the plots, Method I and III give better ranks to frequent OOV proper names and lower ranks to rare OOV proper names.

Table 4.2: Maximum MAP for rare and frequent OOV proper names, obtained using the three retrieval methods for LDA Topic Model. (Best Performance in each category is highlighted in bold. * denotes statistically insignificant difference compared to the best performance in that category.)

Method	Type of OOV PNs	Reference	ANTS	KATS
LDA-MI	all	0.400	**0.379**	0.388
LDA-MII	all	**0.414**	0.372*	**0.399**
LDA-MIII	all	0.334	0.291	0.314
LDA-MI	rare	0.069	0.060	0.065
LDA-MII	rare	**0.215**	**0.173**	**0.208**
LDA-MIII	rare	0.032	0.020	0.026
LDA-MI	frequent	**0.609**	**0.579**	**0.591**
LDA-MII	frequent	0.515	0.473	0.493
LDA-MIII	frequent	0.524	0.460	0.495

Table 4.2 presents a more quantitative evaluation as compared to Figure 4.8. It lists the maximum MAP for rare and frequent OOV proper names separately, as achieved with the three retrieval methods using LDA topic model. Using the results from Table 4.2 and Figure 4.8, we can add following observations to those made from Figure 4.6:

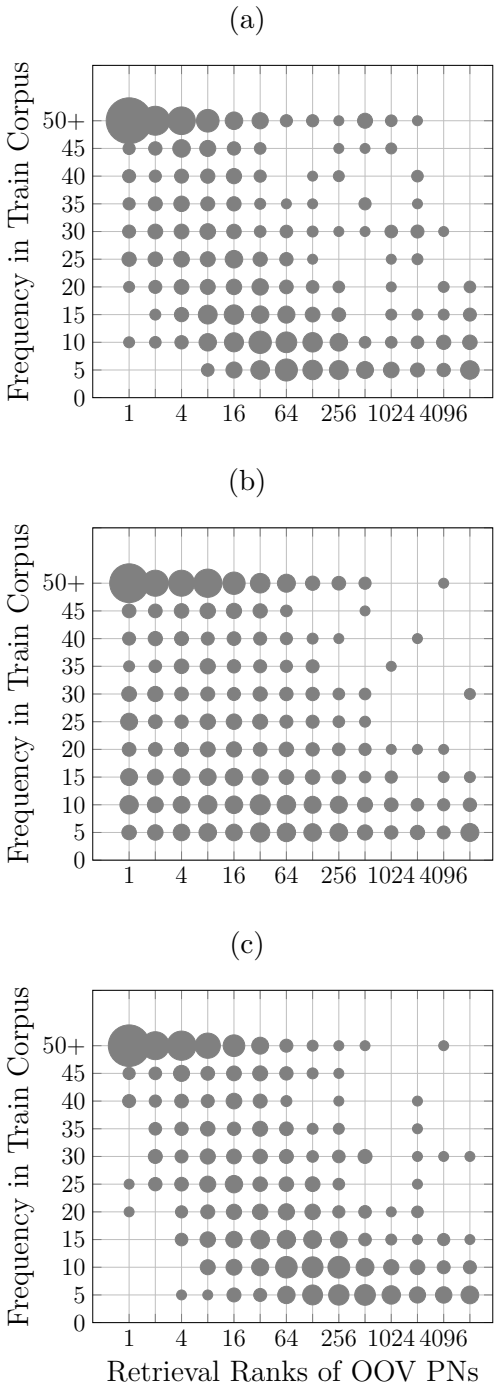


Figure 4.8: Rank-Frequency distribution for retrieval of OOV PNs for (a) LDA-MI, (b) LDA-MII, and (c) LDA-MIII.

- LDA-MII gives the best overall MAP performance among LDA methods as it significantly improves the MAP for rare OOV proper names as compared to LDA-MI and LDA-MII. This improvement is due to document specific topic representations for OOV proper names. However, it also leads to some loss in performance for the frequent OOV proper names.
- As we had hypothesised, LDA-MI and LDA-MIII which rely on global OOV proper name topic representations are clearly biased against rare OOV proper names.

4.3.2 Effects in LSA

We observed earlier, in Table 4.1, that Method II with LSA (LSA-MII) clearly outperformed the Method II with LDA (LDA-MII) in terms of MAP (although not in terms of robustness to LVCSR errors). However, Method I with LSA (LSA-MI) is almost as good as Method I with LDA (LDA-MI). In order to investigate further we plot the rank-frequency distribution for retrieval using LSA Method I and LSA Method II. This plot is shown in Figure 4.9. It can be observed that LSA Method I gives poor performance for the frequent OOV proper names whereas the Method II performs good for both frequent as well as the rare OOV proper names.

Table 4.3: Maximum MAP for rare and frequent OOV proper names, obtained using the two retrieval methods for LSA. (Best Performance in each category is highlighted in bold. * denotes statistically insignificant difference compared to the best performance in that category.)

Method	Type of OOV PNs	Reference	ANTS	KATS
LSA-MI	all	0.417	0.331	0.370
LSA-MII	all	**0.527**	**0.463**	**0.485**
LSA-MI	rare	**0.370**	**0.316**	**0.340**
LSA-MII	rare	0.309	0.262	0.281
LSA-MI	frequent	0.413	0.311	0.356
LSA-MII	frequent	**0.634**	**0.562**	**0.586**

Table 4.3 presents a quantitative evaluation, listing the maximum MAP for rare and frequent OOV proper names separately. Using the results from Table 4.3 and Figure 4.9, we can add the following few more observations to the ones previously stated:

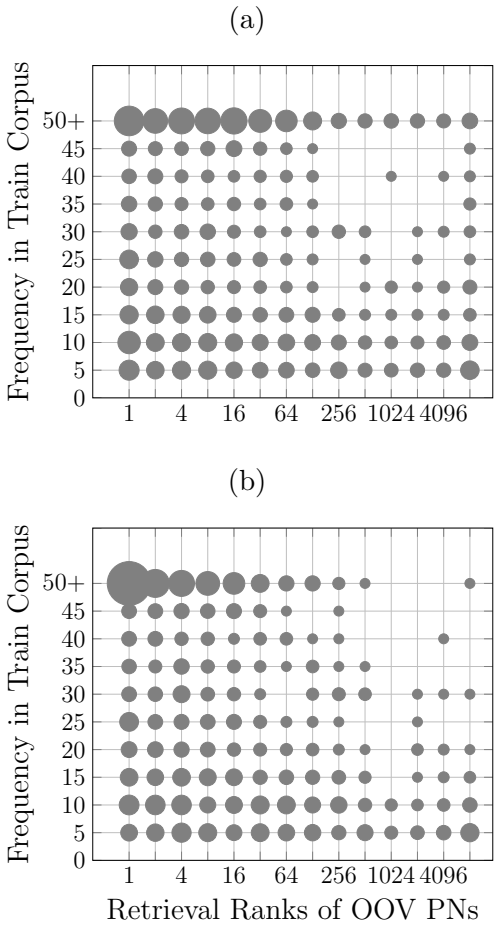


Figure 4.9: Rank-Frequency distribution for retrieval of OOV PNs for (a) LSA-MI, and (b) LSA-MII.

- Document specific representation again gives the best overall MAP performance. However, the improvement given by LSA-MII for the rare OOV proper names is less than with LSA-MI. But the improvements obtained with any of the LSA methods for rare OOV proper names is better than those obtained with the LDA methods.
- Document specific representations can improve performance of frequent OOV proper names too, as opposed to the observation for LDA where LDA-MII caused loss in performance for the frequent OOV proper names.

4.4 OOV PN Re-ranking with Lexical Context Model

A problem with ranking proper names using LDA topic models is that if topic k is prominent for a test document h (i.e. $p(k|h)$ is high) then all the proper names which have high $p(\tilde{v}_x|k)$ take higher ranks. For instance, the diachronic news corpus from the period of the 2014 Football World Cup leads to a topic of sports which is dominated by football. As a result, the topic models tend to give higher scores to football proper names whenever a document handles any sports topic. Increasing the number and granularity of topics is one possible solution to this problem, but its is not a feasible solution when the diachronic corpus is not large enough. Thus topics alone may not be discriminant enough for ranking OOV proper names.

To address this problem we proposed a lexical context model in [Sheikh et al., 2015b] to re-rank OOV proper names retrieved by the LDA topic model. The proper name lexical context model is structured such that each word in a document is generated directly by a proper name in that document. During training the model learns which words are generated by each of the proper names, using the co-occurrences of proper names and words within and across documents. During test, the lexical context model is used to improve the scores of those proper names which are more likely to have generated these words. Continuing our previous example, when the document is about a sport other than football the lexical context model will help to improve the scores of proper names specific to this sport.

This lexical context model showed some improvements in MAP of retrieval of OOV proper names in our work in [Sheikh et al., 2015b]. However, the diachronic corpus used there was much smaller (compared to the *L'Express* diachronic text corpus used in this dissertation). Hence we would like to study how much improvement the lexical context model provides with the relatively larger *L'Express* diachronic text corpus.

4.4.1 Lexical Context Model

Figure 4.10 shows the graphical representation of our proposed proper name lexical context model. In its structure, it shares similarity with the smoothed LDA model [Blei et al., 2003]. A description of the different variables in this models is as follows:

w is the word observed in a document

$\tilde{w}$ is the unobserved proper name which generates w

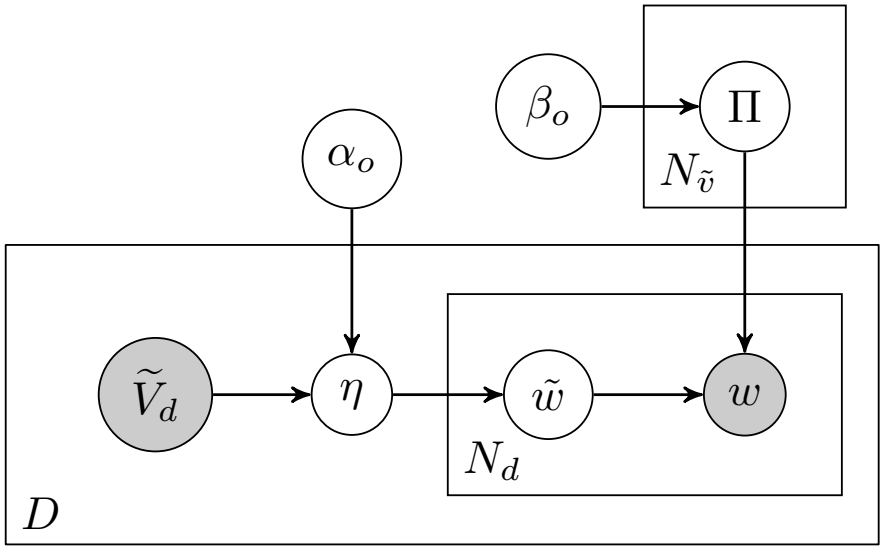


Figure 4.10: Proper name Lexical Context Model

N_d is the number of words in document d

$N_{\tilde{v}}$ is total number of proper names

$\Pi_{\tilde{v}}$ is the distribution of words for the proper name $\tilde{v}$

η_d is the distribution of proper name (word counts) in document d

$\widetilde{V}_d$ is the (known) set of proper names in a document d

α_o, β_o are Dirichlet priors to η, Π respectively

For the $\mathfrak{i}^{th}$ term in a diachronic document d, a proper name $\tilde{w}_\mathfrak{i}$ is sampled from the document specific proper name distribution η_d, i.e. $\tilde{w}_\mathfrak{i} \sim Multi(\eta_d)$. Then a corresponding word $w_\mathfrak{i}$ is sampled from the proper name specific word distribution $\Pi_{\tilde{w}_\mathfrak{i}}$, i.e. $w_\mathfrak{i} \sim Multi(\Pi_{\tilde{w}_\mathfrak{i}})$. $\widetilde{V}_d$ is used as a prior knowledge in training.

We use Gibbs sampling to estimate the proper name assignments to each word; the sampling equation being:

$$p(\tilde{w}_\mathfrak{i} = \tilde{v} | w_\mathfrak{i} = v, \tilde{w}^{(-\mathfrak{i})}, w^{(-\mathfrak{i})}, \widetilde{V}_d) \propto \frac{n^{(-\mathfrak{i})}_{d,\tilde{v}} + \alpha_o}{\sum_{\tilde{v}'=1}^{N_{\tilde{v}}} n^{(-\mathfrak{i})}_{d,\tilde{v}'} + \alpha_o} \frac{n^{(-\mathfrak{i})}_{\tilde{v},v} + \beta_o}{\sum_{v'=1}^{N_v} n^{(-\mathfrak{i})}_{\tilde{v},v'} + \beta_o} \quad (4.10)$$

where $\tilde{w}_{\mathfrak{i}} = \tilde{v}$ implies that the $\mathfrak{i}^{th}$ term in a document is assigned proper name $\tilde{v}$. The superscript $^{(-\mathfrak{i})}$ denotes that the $\mathfrak{i}^{th}$ term itself is left out of the counts and calculation. $n_{d,\tilde{v}}$ is the number of words in d assigned to proper name $\tilde{v}$. $n_{\tilde{v},v}$ is the number of times word v is assigned to proper name $\tilde{v}$. N_v is the total number of non proper name words.

4.4.2 Re-Ranking with Lexical Context

Lexical context is used only to re-rank OOV proper names. During test, the topic model is first used to choose top-N topic relevant OOV proper names. Then the lexical context model is used to re-rank OOV PNs in the top-N list. We use Gibbs sampling to infer the best OOV proper name assignments to each word in h, using a modified equation:

$$p(\tilde{w}_{\mathfrak{i}} = \tilde{v} | w_{\mathfrak{i}} = v, \tilde{w}^{(-\mathfrak{i})}, w^{(-\mathfrak{i})}, \widetilde{V}_d^T) \;\propto\; p_T(\tilde{v}|h)\, \frac{c_{\tilde{v},v}^{(-\mathfrak{i})} + n_{\tilde{v},v} + \beta_o}{\sum_{v'=1}^{N_v} c_{\tilde{v},v'}^{(-\mathfrak{i})} + n_{\tilde{v},v'} + \beta_o} \quad (4.11)$$

where $\tilde{w}_{\mathfrak{i}} = \tilde{v}$ implies that the $\mathfrak{i}^{th}$ term in LVCSR hypothesis h is assigned an OOV proper name $\tilde{v}$, from the top-N OOV proper names $\widetilde{V}_d^T$. $c_{\tilde{v},v}$ is the number of times term $w_{\mathfrak{i}}$ in h is the word v and assigned to a top-N OOV proper name $\tilde{v}$. $n_{\tilde{v},v}$ is the count saved from training. $p_T(\tilde{v}|h)$ is the score given to OOV proper name $\tilde{v}$ by the topic model.

The top-N OOV proper names are then re-ranked using:

$$P_N(\tilde{v}|h) \approx p_T(\tilde{v}|h) + s_h^L\, \frac{\sum_{v'=1}^{N_v} c_{\tilde{v},v'} + \alpha_o}{N_{vh} + \alpha_o N} \quad (4.12)$$

where, N_{vh} is the number of words in LVCSR hypothesis h, and N is same as the (N in) top-N. s_h^L is a scaling factor to combine topic and lexical model scores of the top-N OOV proper names. This scaling factor has to be tuned using the validation set.

Table 4.4 shows the relative improvements in maximum MAP obtained by re-ranking the retrieval results, obtained from LDA Method I, using the proposed lexical context model. From the results we can see that the lexical context re-ranking helps, but only when the number of topics are small (w.r.t. the given diachronic text corpus). As the number of LDA topics are increased, it does not give any improvements. The scaling factor s_h^L was chosen such that it does not cause degradation in the MAP.

Table 4.4: Relative improvement in maximum MAP after applying lexical context re-ranking to OOV proper name retrieval results from LDA Method I.)

Number of Topics	Reference Transcription	ANTS Transcription	KATS Transcription
100	16.3%	8.6%	13.5%
200	7.3%	3.9%	6.2%
300	1.1%	0.3%	0.6%
400	≈0%	≈0%	≈0%

4.5 Retrieving OOV PNs with Entity Topic Models

Newman et. al. [Newman et al., 2006] proposed Entity-Topic models as an extension of the LDA topic model, to explicitly address the interactions between entities, i.e. person, organisation, locations[4], and topics. In their work, Entity-Topic models were shown to perform better than LDA for entity prediction tasks. This motivated us to try these extensions of LDA in our task, by treating OOV proper names as entities.

4.5.1 Entity Topic Models

Figure 4.11 shows graphical representations of entity-topic models. We can see that the entity topic models share structural similarity with LDA, except that **in entity-topic models there is a separate hierarchy for the generation of words and entities**. $\tilde{z}$ denotes the latent entity topic variable which generates an entity $\tilde{w}$, based on the entity topic distribution $\tilde{\phi}$, whereas z denotes the latent word topic variable which generates word w based on word topic distribution ϕ.

SwitchLDA has a switch variable x which controls the generation of words and entities. In *Conditionally Independent* LDA (CI-LDA) and SwitchLDA, the document specific topic distribution θ_d generates both word and entity topics. Whereas in the *Correspondence LDA* models (CorrLDA1 and CorrLDA2), the document specific topic distribution θ_d generate the word topics and then word topics are used to generate entity topics and entities. CorrLDA2 has an additional hierarchy ($z \rightarrow x \rightarrow \tilde{z}$) which allows different number of word and entity topics (T and $\tilde{T}$). A more detailed description of the variables and the generative/sampling process for these models is available in [Newman et al., 2006].

[4]which can include non proper name words

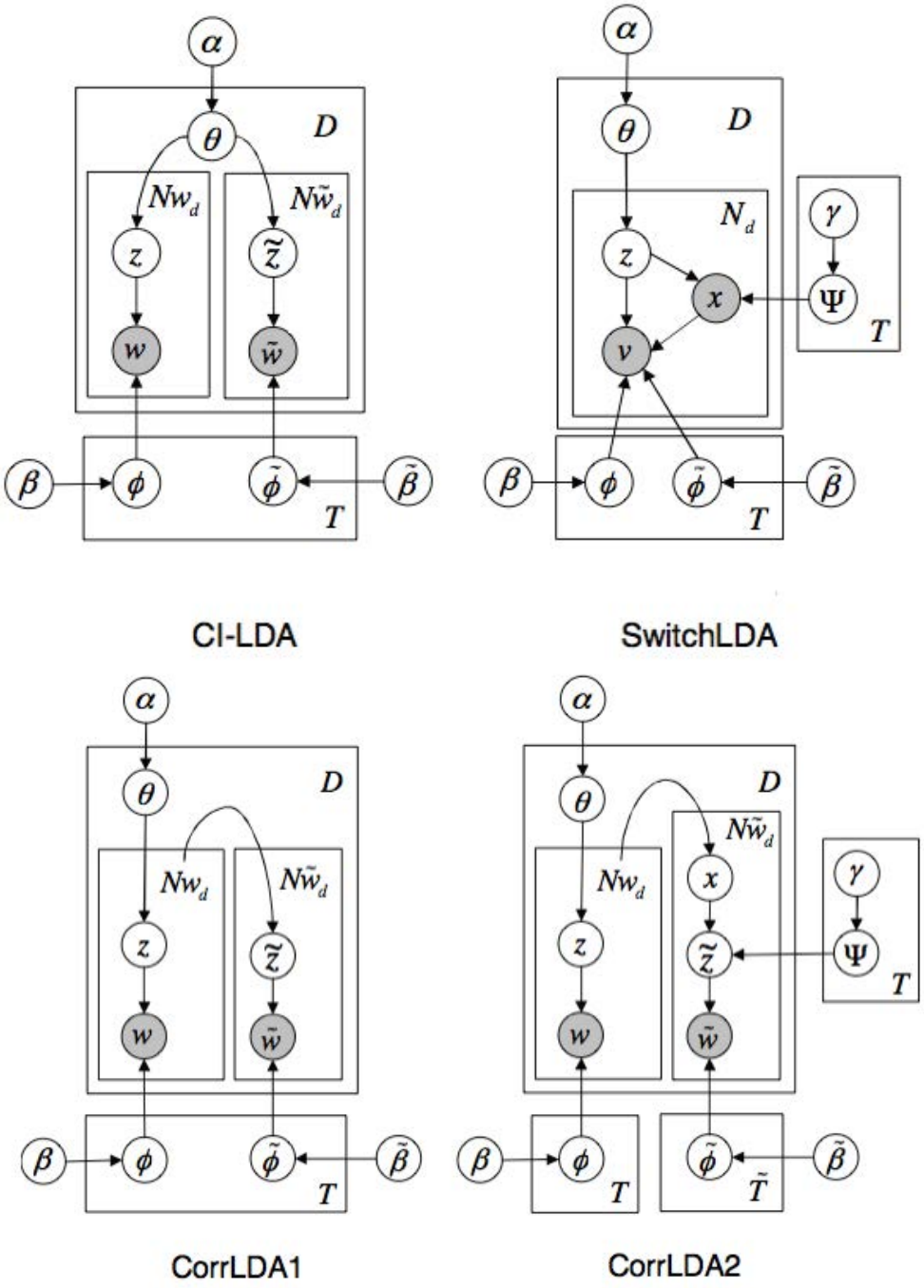


Figure 4.11: Graphical representation of Entity-Topic models [Newman et al., 2006].

In addition to these models, we tried further **variations of the entity-topic models by inter-changing the hierarchy of word topics and entity topics**. In the CorrLDA1 model, word topic z_i is first sampled from the document topic distribution θ, i.e. $z_i \sim Multi(\theta)$, and entity topic $\tilde{z_j}$ is then sampled uniformly from the word topics, $\tilde{z_j} \sim Unif(z_1, z_2...z_{N_{w_d}})$. Instead of this, entity topic $\tilde{z_j}$ can be first sampled from the document topic distribution θ, $\tilde{z_j} \sim Multi(\theta)$, and the word topic can then be sampled uniformly from entity topic $z_i \sim Unif(\tilde{z}_1, \tilde{z}_2...\tilde{z}_{N_{\tilde{w}_d}})$. We refer to this variation of CorrLDA1 as *CorrLDA1-Flipped* (CorrLDA1-F). Figure 4.12 depicts the graphical model for the CorrLDA1-F model. The motivation for trying CorrLDA1-F model is that since it learns entity centric topics, it may perform better in retrieving OOV proper names using the document topic.

Similar to CorrLDA1, even the CorrLDA2 model can be flipped. This has been proposed as *Entity Centred Topic Model* (ECTM) [Hu et al., 2013]. We tried this model in our work in [Sheikh et al., 2015a], where the diachronic text corpus was smaller. However, with a large diachronic text corpus like *L'Express*, training the ECTM model takes weeks due to complexity of the Gibbs sampling of the hierarchy of entity-word topics.

Thus we study the performance of 6 different topic models for OOV PN retrieval: classic LDA, CI-LDA, SwitchLDA, CorrLDA1 and CorrLDA2, and the CorrLDA1-F model discussed above.

4.5.2 Setup for OOV PN Retrieval

To use these models for our task of OOV proper name retrieval we divide the topic model vocabulary into a set of N_v words & proper names in the LVCSR vocabulary and a set of $N_{\tilde{v}}$ OOV proper names[5]. ϕ_v, the topic distribution to IV words, and $\phi_{\tilde{v}}$, the topic distribution to OOV proper names, can be estimated using Gibbs sampling as in case of LDA. The Gibbs sampling equation for these models are also analogous to LDA and are available in [Newman et al., 2006].

During test the latent topic mixture of the LVCSR hypothesis h, i.e. $p(t|h)$, is inferred by sampling the topic assignments for words in h, using the word-topic assignments accumulated during training. This procedure is the same as that used for LDA, as discussed in Section 3.3.2. Then the likelihood of an OOV proper

[5]The topic model vocabulary can be divided in two more ways, (a) words and PNs, as in the original work [Newman et al., 2006]; (b) forming three categories i.e. IV PNs, OOV proper names and other words. We tried these variations, however our approach discussed above (to separate OOV proper names from IV words) gives relatively better or similar results on our dataset.

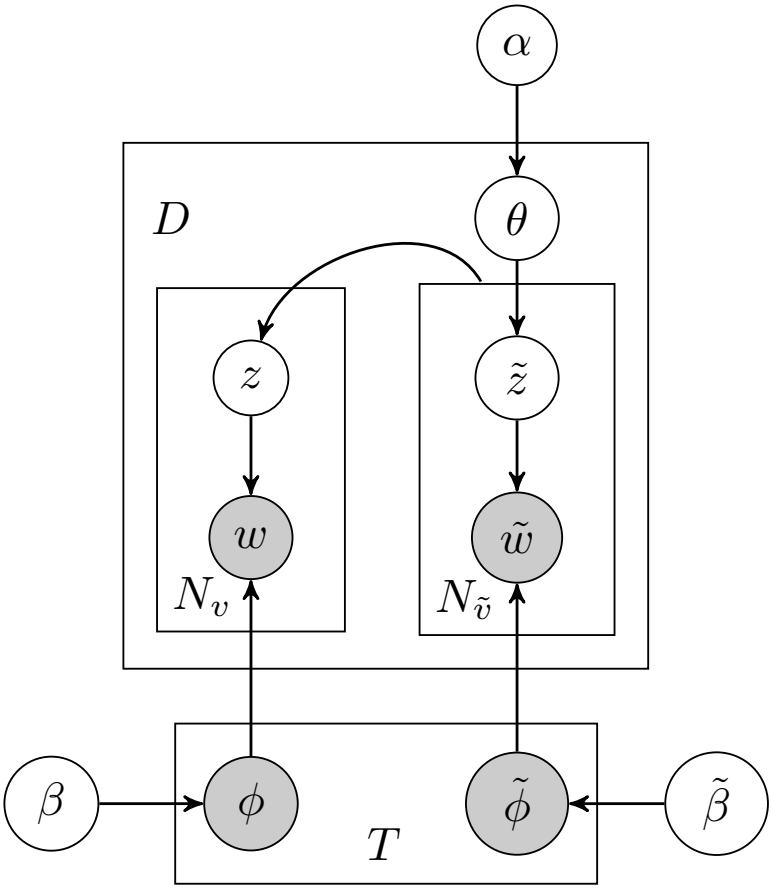


Figure 4.12: Plate Diagram for the CorrLDA1-F Entity Topic Model

name $\tilde{v}_i$ in the diachronic corpus is calculated using Equation (4.2) for Method I and using Equation (4.3) for Method II. The only exception is the entity centered model CorrLDA1-F. In CorrLDA1-F, the document topic representations (θ) are dependent on OOV proper names and OOV proper names are not observed in LVCSR hypothesis. Therefore, as an alternative during test, we infer the word topic representation for each of the diachronic text corpus document using the same procedure as for the LVCSR hypothesis. These (independently) inferred document topic vectors are now treated as document specific OOV proper name representations in Method II.

It must be noted that Method III can also be tried with the entity-topic models discussed above. However, as it was discussed earlier, Method III is an alternative for computational efficiency. We will thus exclude the evaluation of Method III for the different entity-topic models.

4.5.3 Performance of Entity Topic Models

Our work in [Sheikh et al., 2015a] compared the performance of LDA and entity-topic models using a smaller diachronic text corpus and test set. The results from these experiments did not show significant differences between the performance of LDA and entity-topic models. One of our observation was that the number of OOV proper name instances in the diachronic text corpus used in our previous work was quite small. In this dissertation we would like to present the results obtained from our experiments with the entity-topic models trained on the *L'Express* diachronic text corpus which has about 10 times more training data and OOV proper name instances. We do not perform an explicit hyper-parameter search for the entity-topic models and present the results of Method I and Method II obtained with 400 topics. The α, β hyper-parameters are set to 0.01, which are the values corresponding to best LDA Method I configuration.

Table 4.5: Comparison of LDA and Entity Topic models in terms of maximum MAP obtained with Method I and Method II on the *Euronews* audio test set. (Best Topic model performance is highlighted in bold. * denotes statistically insignificant difference compared to the best configuration.)

	Method I			Method II		
	Reference	ANTS	KATS	Reference	ANTS	KATS
LDA	0.400	0.379	0.387	0.390	0.335	0.357
CI-LDA	0.398	0.371	0.385	**0.417**	**0.362**	0.379*
SwitchLDA	0.400	0.370	0.383	0.402	0.354*	**0.383**
CorrLDA1	0.390	0.366	0.375	0.395	0.348	0.368
CorrLDA1-F	**0.421**	**0.394**	**0.402**	0.372	0.332	0.341
CorrLDA2	0.376	0.351	0.357	0.401	0.357*	0.369

The maximum MAP obtained on the reference transcription of the *Euronews* audio test set as well as automatic transcriptions obtained from the ANTS LVCSR system and KATS LVCSR system (presented in Section 2.4.2) are shown in Table 4.5. Similar to our results in [Sheikh et al., 2015a], we can observe that most entity models perform only as good as LDA, as opposed to the improved entity prediction results obtained in [Newman et al., 2006]. However, our proposed variation CorrLDA1-F significantly outperforms LDA and other entity-topic models for Method I, while it did not perform that well in [Sheikh et al., 2015a]. This is because in [Sheikh et al., 2015a] the total count of our entities (i.e. OOV proper names) was quite small and as the topics are centred around entities the model representations were not optimal. In our current experiments there is a relatively

larger amount of entity-instances to learn better entity centered topic representations. Contrary to Method I, CorrLDA1-F does not perform well with Method II. As discussed in Section 4.5, when testing with Method II the generation hierarchy of CorrLDA1-F is not followed. Document topic representations for both train and test are inferred independently from the word topics, without the availability of OOV proper names and their topics. This could be the reason for the reduced performance of CorrLDA1-F with Method II. For Method II different models including CI-LDA, SwitchLDA and CorrLDA2 seem to perform better than LDA.

4.6 On the Selection of the Diachronic Text Corpus

Until now we have presented different methods for exploiting topic context to retrieve OOV proper names. The topic context of the OOV proper names are modelled and derived from the diachronic text corpus and therefore it is important to study the selection of documents for the diachronic corpus. In this section we try to investigate some characteristics of the diachronic corpus which can affect the performance of retrieval of OOV proper names.

4.6.1 New Diachronic Text Corpora

Table 4.6 presents a new set of new diachronic text corpora which will be used as training sets in our study in this section. The datasets are collected from two sources: (a) the French newspaper *L'Express* (http://www.lexpress.fr/), and (b) the French newspaper *Le Figaro* (http://www.lefigaro.fr/). Both *L'Express* and *Le Figaro* contain news articles. The LX in LX+FIG is the *L'Express* corpus corresponding to Jan - Jun 2014 as presented in Table 2.1. The details on corpora setup in our experiment are presented in Section 4.6.3.1.

Similar to the processing for corpora in Table 2.1, TreeTagger[6] [Schmid, 1994a] is used to automatically tag proper names in the text. The words and proper names which occur in the lexicon of our LVCSR system are tagged as IV, and the remaining proper names are tagged as OOV. As before we use the term 'target OOV proper name coverage' to refer to the percentage of OOV proper names in *Euronews* videos which can be recovered with a given diachronic corpus. The target OOV proper name coverage for each of the diachronic text corpus is as follows: 40% for FIG, 52% for LX+FIG combined and 54% for LX-18m, as compared to 42% for LX only. We can observe that LX-18m captures more new

[6]http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

proper names (28.2K) as compared to those by LX+FIG (18.4K) but the OOV proper name coverage of the two differs only by 2% absolute.

Table 4.6: More Diachronic News Datasets

	Le Figaro (FIG)	*L'Express* + *Le Figaro* (LX+FIG)	*L'Express* (LX-18m)
Type of Documents	Text	Text	Text
Time Period	Jan - Jun 2014	Jan - Jun 2014	Jul 2013 - Dec 2014
Number of Documents[1]	59K	104K	142K
Vocabulary Size[2]	140K	180K	270K
Corpus Size (word count)	18M	42M	70M
Number of PN unigrams[2]	51K	80K	104K
Total PN count	1.3M	2.7M	4.2M
Number of OOV unigrams[3]	11.9K	24.4K	37.1K
Documents with OOV[3]	36.4K	73K	109K
Total OOV count[3]	142K	320K	509K
Number of OOV PN unigrams[3]	8.8K	18.4K	28.2K
Documents with OOV PN[3]	30K	61.3K	93.5K
Total OOV PN count[3]	103K	243K	388K
Target OOV PN coverage	40%	52%	54%

[1]K denotes *Thousand* and M denotes *Million*
[2]unigrams occurring less than two times are ignored
[3]unigrams occurring in less than three documents ignored, documents with more than 20 and less than 500 terms
Note: OOV, OOV PN statistics are after term-document filtering

4.6.2 Configurations of the Diachronic Corpus

The topic context of OOV proper names, which enables retrieval of the relevant OOV proper names, is learned from a diachronic text corpus. We would like to study the effect of selection of documents for the training diachronic corpus. In particular we study the following configurations of the diachronic corpus.

(A) Documents containing OOV proper names[7] and from the same time period as the test set, e.g. *L'Express* documents containing OOV proper names and corresponding to the same 6 months of the *Euronews* video test set.

(B) Documents coming from two different originating sources, e.g. *L'Express* and *Le Figaro*.

(C) Documents from a time period extending beyond the timeline of the test set (e.g. *L'Express* documents from 18 months for the *Euronews* video test set).

(D) Documents with OOV proper names that are collected from one source and then for the less frequent OOV proper names in this collection new documents are additionally collected from another source. As discussed in Section 4.3, retrieval of less frequent OOV PNs has a poor performance because there is not enough data to learn their topic distribution. This problem of reduced representation of less frequent OOV proper names motivates us to study this configuration.

4.6.3 Experimental Analysis

It must be noted that our **objective in these experiments is not to achieve the best (MAP) retrieval performance but instead to study the coverage (recall) behaviour of diachronic text corpora**. Our work in [Sheikh et al., 2016a] discussed these experiments and following this work we will use LDA Method I (from Section 4.1) for retrieving relevant OOV proper names.

4.6.3.1 Experiment Setup

The datasets presented in Table 2.1 and Table 4.6 will be used for our experiments. The different configurations discussed in Section 4.6.2 will be studied with the *L'Express* and *Le Figaro* datasets, where they will be used as a diachronic corpus to train the topic models. Audio news extracted from the *Euronews* video dataset will be the test set. To train the context models, the diachronic text corpus vocabulary is lemmatized and filtered by removing proper names occurring only once, non proper name words occurring less than 4 times, and using a stoplist of common and non-content French words. Moreover, a POS based filter is employed to choose only words tagged as proper name, noun, adjective, verb or acronym. The retrieval results reported are obtained with an LDA topic model

[7]including documents not containing OOV PN did not give significant improvement in the retrieval performance

with 300 topics trained on each of the diachronic text corpora. We ignore the fact that each corpus will perform optimally for a certain number of topics.

Note that LX, FIG and LX+FIG correspond to configurations A and B of Section 4.6.2. LX-18m corresponds to configuration C. Corresponding to configuration D, we form a corpus LX+rFIG which contains documents of LX (*L'Express*, Jan 2014 - Jun 2014) supplemented with documents from FIG (*Le Figaro*, Jan 2014 - Jun 2014) which contain OOV proper names occurring less than 10 times in LX. The target OOV proper name coverage for LX+rFIG was found to be 49%.

4.6.3.2 Retrieval Performance with Different Diachronic Corpora

Figure 4.13 presents the effects of diachronic corpus configurations A, B, C and D discussed in Section 4.6.2. Figure 4.13 shows a graph of recall and MAP of retrieval of OOV proper names. As the focus is on comparing different diachronic corpora, only the performance on reference and ANTS LVCSR transcriptions are shown. The X-axes represent the number (N) of top-N retrieved OOV proper names. The Y-axes represent recall (top) and MAP (bottom) of the target OOV proper names.

Table 4.7 compares the MAP@128 (maximim MAP) obtained for different diachronic corpora. By analysing the performance in Table 4.7 and Figure 4.13, we can draw the following observation:

- Smaller corpora achieve good MAP but give low coverage of the target OOV proper names: the MAP for LX and FIG is the highest but they do not have the best recall rates. They have a small number of OOV proper names to choose from and make smaller retrieval errors but they give only 40% coverage of the target OOV proper names.

- Expanding the time period of a diachronic corpus (as in LX-18m) gives better target OOV PN coverage, but not the best recall rates. Similar is the case with including data from additional source (as in LX+FIG). A low MAP is obtained in these cases because both the number of possible OOV proper names and the number of target OOV proper names have increased but the recall rate is still poor than (or similar to) that of LX. Such corpora can possibly be exploited by training a larger number of topics or by employing better retrieval methods [Sheikh et al., 2015c, Sheikh et al., 2016b].

- Adding new documents corresponding to less frequent OOV proper names is not effective: performance of LX+rFIG is similar to that of LX+FIG. We

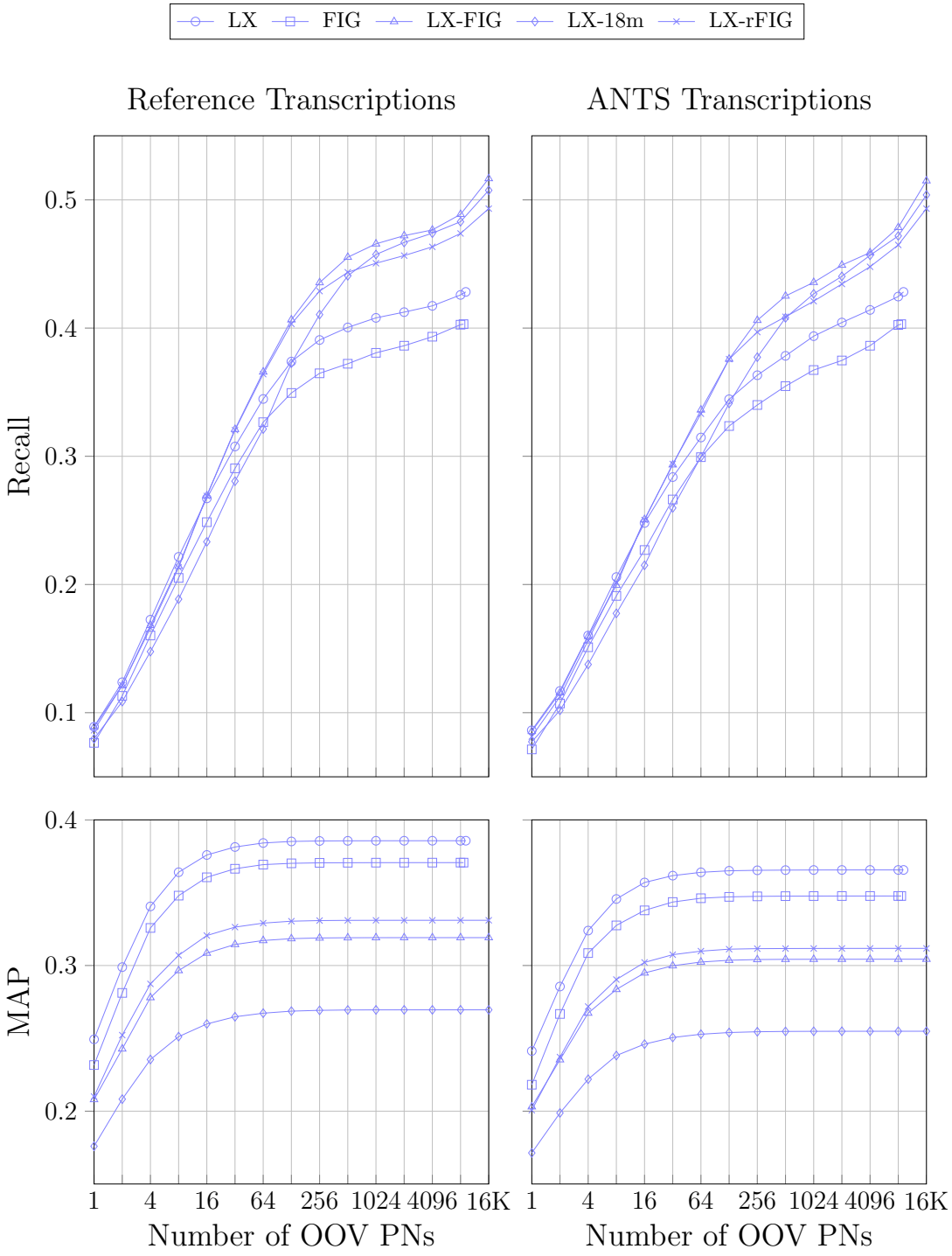


Figure 4.13: Recall and MAP for OOV proper name retrieval on *Euronews* news video test set with different diachronic text corpora for training the LDA topic model. LDA Method I is used for OOV PN retrieval.

found that adding documents containing less frequent OOV proper names in LX+rFIG leads to inclusion of more than 60% of FIG documents. The additional data not only increases data for learning topic representation of less frequent OOV proper names but also comes with additional less frequent OOV proper names and more instances of frequent OOV PNs. Further analysis of the ranks of the less frequent OOV proper names obtained with LX, LX+FIG and LX+rFIG showed that the ranks with LX+rFIG are better with respect to LX but similar to that with LX+FIG.

- Using diachronic text documents of the same time period and from multiple sources (LX+FIG) gives a good balance of recall, MAP and target OOV PN coverage.

Table 4.7: Comparison of MAP@128 for different diachronic corpora

	Reference Transcription	ANTS Transcription
FIG	0.370	0.347
LX	0.385	0.365
LX+FIG	0.319	0.304
LX+rFIG	0.331	0.311
LX-18m	0.269	0.254

4.7 Conclusion

In this chapter, we presented two methods for retrieval of OOV proper names. The first method measures the closeness of OOV proper names and LVCSR hypothesis in the context space, and it relies on the global representation of each OOV proper name in the context space. On the other hand, the second method relies on multiple document specific representations of each OOV proper name. These two methods were extended to different type of semantic/topic representations including LSA, LDA topic model and entity-topic models.

Our analysis and observations from the comparison of the different models and methods enable us to conclude the following:

- Semantic and topic context models (LSA and LDA) outperform simple word co-occurrence based models (PMI and also Random Projections) in

our task of retrieval of relevant OOV proper names, both in terms of MAP and robustness to LVCSR errors.

- Previous works [Blei et al., 2003, Griffiths et al., 2007] and our earlier experiments on smaller datasets [Sheikh et al., 2016b] showed that LDA based topic models perform better than or as good as LSA. However, experiments with a larger diachronic corpus revealed that LSA outperforms LDA in terms of retrieval MAP. But at the same time, we found that the LDA model performance showed more robustness to LVCSR errors as compared to that for LSA.

- On contrary to LDA, retrieval with LSA model showed that the method based on global representation of OOV proper names can possibly address both frequent and rare OOV proper names.

- Among the proposed retrieval methods, the document specific representations showed the largest improvements in MAP. This method can improve the retrieval of rare OOV proper names with LDA. However, it comes at extra computation cost, which would increase linearly with the size of the diachronic text corpus.

- LDA gives the flexibility to model OOV proper names and their topics separately with entity-topic models. As compared to existing entity-topic models our proposed CorrLDA1-F model gave significant improvements over the basic LDA model. In CorrLDA1-F model the topics are centered around OOV proper names and this gives improvements (when more data and OOV proper name instances are available).

- Inferring the topic distribution of an LVCSR hypothesis with an LDA model requires multiple iterations over entire hypothesis. As an alternative we proposed a retrieval method for LDA which is based on association of words and OOV proper names in the topic space. This method achieves its goal of being a computationally efficient method and gives improvements over an equivalent method based on pointwise mutual information. However it attains a lower performance compared to the other methods based on semantic/topic space representations.

- Exploration of hyper-parameters in the LDA model showed that its MAP performance is very sensitive to the choice of hyper-parameters. Also suggesting that hyper-parameter selection is crucial for performing OOV proper name retrieval using the topic space representations from LDA topic models.

- From our experiments on selection of diachronic text corpus, we can conclude that
  - text from a longer time span can give increased coverage of OOV proper names
  - but a corpus with text from different sources leads to better retrieval performance than relying on text from a single source, even if it corresponds to a longer time span
  - less frequent OOV proper names need improvement in retrieval methods and not just additional training data.

These findings motivate us to continue to explore models with better global representations of OOV proper names and at the same time having robustness to LVCSR errors.

CHAPTER 5

Retrieving OOV PNs using Word Embeddings

Developments in Neural Network based language models [Mikolov et al., 2013c] led to a renewed interest in the field of distributional semantics, more specifically in learning word embeddings. Computational efficiency was one big factor which popularised word embeddings from the CBOW and Skip-gram models. The word embeddings capture syntactic as well as semantic properties of the words [Mikolov et al., 2013b]. As a result they outperformed several other word vector representations on different tasks [Baroni et al., 2014]. This motivates us to study word embeddings for our task of context based OOV proper name retrieval.

In this chapter, **we extend our proposed OOV proper name retrieval methods to utilise word embedding representations**. For this, we first present the important linearity property of word embeddings and discuss how it can be used to form document context embeddings in our task of retrieval of OOV proper names. This is followed by an analysis of model performance and choice of best configuration. Previous works have shown that the word embeddings from the CBOW model are more syntactic whereas the word embeddings from the Skip-gram model are more semantic [Mikolov et al., 2013a] (although the differences in performance in some of the tasks was not quite large). Moreover both the CBOW and Skip-gram models have hyper-parameters to choose. So we will carry out an analysis of performance of both models, under different hyper-parameter settings.

Finally we will compare the retrieval results from the best model configurations to those obtained from the experiments on LSA and LDA models. This will be followed by a conclusion and further directions to our search for robust high performance representations for OOV proper name retrieval.

5.1 Enabling Retrieval Methods for Word Embedding Space

In Section 4.1 we proposed two main methods for OOV proper name retrieval. These methods can be extended to representations from any vector space model. Method I is based on comparison of LVCSR hypothesis and OOV proper names in context space, and Method II is based on document specific OOV proper name representations. However, unlike LDA and LSA neither CBOW nor Skip-gram models the document representation required by the two methods. To address this problem we exploit the linearity property of the word embeddings.

As evident from Equation 3.15 and Figure 3.4 (a), the bag of words representation of the context and the absence of non-linearity at the hidden layer causes the context embedding to be a sum of the word embeddings. This leads to the linearity property in which **word embeddings can be simply added and subtracted to obtain embeddings representing semantically and/or syntactically relevant words**. In effect the embeddings resulting after the linear operations are closer to, and not exactly equal to, the embeddings of the relevant words.

To illustrate the linearity property, Table 5.1 shows sample contexts built using words from French news articles. Alongside are their nearest neighbour terms, which were obtained by calculating cosine similarity between the context embedding and embeddings of all terms in a Skip-gram model. From the examples in Table 5.1 (a) we can see that context embeddings can be obtained by performing an average over the constituent word embeddings, and that the resulting context embeddings are closer to relevant words in the embedding space. We also present Table 5.1 (b), which shows individual words taken from the different context examples and their top five nearest neighbours. It shows that individual word embeddings also carry rich semantic information, but it is too generic to use directly in our task. We thus rely on the linearity property, and perform an average of the word embeddings, to obtain a document context representation.

Thus in our task, a CBOW/Skip-gram model is trained on the diachronic text corpus to learn embeddings for in-vocabulary words and OOV proper names. Given these word embeddings and their linearity property, we obtain a representation for a diachronic corpus text document or LVCSR hypothesis by taking the average of the embeddings of all words. We will refer to this representation as *AverageVec* representation. Given the AverageVec representations for the LVCSR hypothesis and those for the diachronic text corpus documents our proposed OOV proper name retrieval methods, Method I and Method II, can be applied with cosine similarity as the scoring measure, as in case of context vector representations obtained from the LSA model.

Table 5.1: Illustration of linearity property of word embeddings. (a) Sample contexts, built with words from French news articles, and the corresponding nearest neighbour terms obtained by calculating cosine similarity between the context embedding and embeddings of all the terms in the model.

Context of five words	Top five nearest neighbour terms
ski coma CHU hospital accident	*Schumacher* *Méribel* *neurochirurgie* *crânien* *cérébral*
avion recherche trouver vol perdre	*Boeing* *MH370* *Malaysia* *débris* *airlines*
jeux olympique hiver athlète mondiaux	*Sotchi* *ski* *JO-2014* *biathlète* *paralympiques*

(b) Nearest neighbours of individual words, obtained by calculating the cosine similarity between its embedding and embeddings of all other terms.

	ski	*recherche*	*hiver*	*accident*
Top five nearest neighbours	*Alpin* *biathlon* *bosse* *biathlète* *freestyle*	*Bluefin-21* *débris* *Perth* *scientifique* *chercheur*	*week* *JO* *fashion* *neige* *collection*	*déraillement* *crash* *drame* *mortel* *Schumacher*

For Method I, the K dimensional vector representation of the LVCSR hypothesis h is compared with the embedding $\tilde{v}_i$ of an OOV proper name to calculate a score as:

$$\begin{aligned} \text{Method I:} \quad s_i &= \text{Cosine\_Similarity}(h, \tilde{v}_i) \\ &= \frac{h.\tilde{v}_i}{\|h\| \, \|\tilde{v}_i\|} \\ &= \frac{\sum_{k=1}^{K} h_k \, \tilde{v}_{ik}}{\sqrt{\sum_{k=1}^{K} (h_k)^2} \, \sqrt{\sum_{k=1}^{K} (\tilde{v}_{ik})^2}} \end{aligned} \tag{5.1}$$

Similarly for Method II, the K dimensional representation of the LVCSR hypothesis of the audio document h is compared with the context vectors (C_q^i) for each of the OOV proper name $\tilde{v}_i$ to calculate a score as follows:

$$\begin{aligned} \text{Method II:} \quad s_i &= \max_q \{\text{Cosine\_Similarity}(h, C_q^i)\} \\ &= \max_q \left\{ \frac{h.C_q^i}{\|h\| \, \|C_q^i\|} \right\} \\ &= \max_q \left\{ \frac{\sum_{k=1}^{K} h_k \, C_{qk}^i}{\sqrt{\sum_{k=1}^{K} (h_k)^2} \, \sqrt{\sum_{k=1}^{K} (C_{qk}^i)^2}} \right\} \end{aligned} \tag{5.2}$$

To perform retrieval of OOV proper names we calculate s_i for each OOV proper name $\tilde{v}_i$ and then use it as a score to rank OOV proper names relevant to h.

5.2 Experiments and Results

In this section we present an evaluation of our proposed approaches to use word embeddings for the retrieval of OOV proper names. One of the objective, when performing retrieval of OOV proper names, is to learn relevant semantic embeddings. Syntactic embeddings could also be useful in predicting proper names, however they rely more on local word context and it has been commonly observed that LVCSR hypotheses are erroneous in the region of OOV words. Based on the conclusions from earlier works, the Skip-gram model would be favourable to obtain word embeddings which capture semantic properties and thus perform better for semantic tasks. As our task will rely also on the document context embeddings, we would like to analyse the performance of both CBOW and Skip-gram models.

We will first present discussions on the selection of hyper-parameters for learning the embeddings. These will be followed by a comparison of the retrieval results achieved with the best model configurations. The experiment corpus setup and the retrieval evaluation measures have been presented in Section 2.4.

5.2.1 Selection of Model Hyper-parameters

The hyper-parameters for CBOW and Skip-gram models are the dimensionality of the word embeddings and the context window length. Previously, it has been shown that the context window size is a crucial parameter for distributional methods relying on context windows [Bullinaria and Levy, 2007]. It is also very crucial for our task as it will help composition of the document context embeddings. Furthermore, the number of training epochs is also important, as in any neural network model with an iterative learning procedure. Generally more training epochs show better results, which is also true for the CBOW and Skip-gram models[1]. However these improvements stop after some training epochs and further epochs can lead to over fitting.

The negative sampling approach for training CBOW and Skip-gram models models [Mikolov et al., 2013a] comes with an additional hyper-parameter, namely number of negative samples. This approach gave lesser validation set performance in our initial experiments. So we will be using the approach with hierarchical softmax, which was also proposed in [Mikolov et al., 2013a].

5.2.1.1 Method I and Model Hyper-parameters

Figure 5.1 and Figure 5.2 show charts depicting the variation in maximum MAP values obtained for Method I with CBOW and Skip-gram model embeddings respectively, on the validation set. Each figure shows MAP for a range of values for context window size, word embedding size and number of training epochs.

We tried until a window size of 40, limited by the length of the smallest documents in our datasets. Beyond an embedding size of 500, and the respective number of epochs shown, the improvements were not statistically significant. With the different hyper-parameters, the maximum MAP for the CBOW model varies between 0.151 and 0.392. And for skip-gram the MAP values vary between 0.275 and 0.494.

In general, bigger word embeddings are better for larger amounts of text data

[1]The original work of Mikolov et. al obtained slightly better performance when training for more epochs [Mikolov et al., 2013a].

with several topics and events. We can observe an increase in maximum MAP performance until an embedding size of 500, beyond which the improvement is not significant with the *L'Express* diachronic text corpus. As in the case of the LDA model, word embeddings of 400 dimensions seem to be optimal for Method I, both for CBOW and Skip-gram models.

As we had predicted the context window size proves to be quite crucial for our task. Larger context windows enable a better composition of the document context embeddings and gives a better MAP performance for both the CBOW and Skip-gram models. Comparing the number of training epochs for the two models, the performance of the CBOW model stops improving after 5 epochs while that of the Skip-gram model continues to improve until 25 epochs. We believe that the CBOW model with larger context window starts overfitting faster. This particular problem must not be happening with the Skip-gram model because, instead of predicting the complete context window, it predicts one word from the context window at a time, as discussed in Section 3.4.1.2.

Based on the MAP performances on the validation set, the CBOW and Skip-gram models trained with a context window size of 40, with 5 and 25 epochs respectively, are chosen as the best performing word embedding model configurations for Method I. Models with better performance but statistically insignificant improvement are ignored in favour of choosing models of an embedding dimension of 400, to keep similarity to the LDA and LSA models chosen before.

5.2.1.2 Method II and Model Hyper-parameters

Figure 5.3 and Figure 5.4 show charts depicting the variation in maximum MAP values on the validation set achieved by Method II with the CBOW and Skip-gram model embeddings respectively. The MAP values are obtained for a range of values of context window size, word embedding size and number of training epochs.

As with Method I, we experimented window size up till 40, limited by the length of the smallest documents in our datasets. Similarly, we varied the embedding size from 100 till 500, and the number of epochs from 1 to 10 for the CBOW model and from 1 to 25 for the Skip-gram model. As evident from Figure 5.3 and Figure 5.4, for Method II, the maximum MAP for the CBOW model varies between 0.361 and 0.474, and for Skip-gram it varies between 0.451 and 0.485.

The next important observation is that for Method II with Skip-gram model representations (see Figure 5.4) the variations in MAP performance are quite limited compared to that of CBOW (see Figure 5.3), as well as of Method I with both CBOW and Skip-gram word embeddings (see Figure 5.1 and Figure

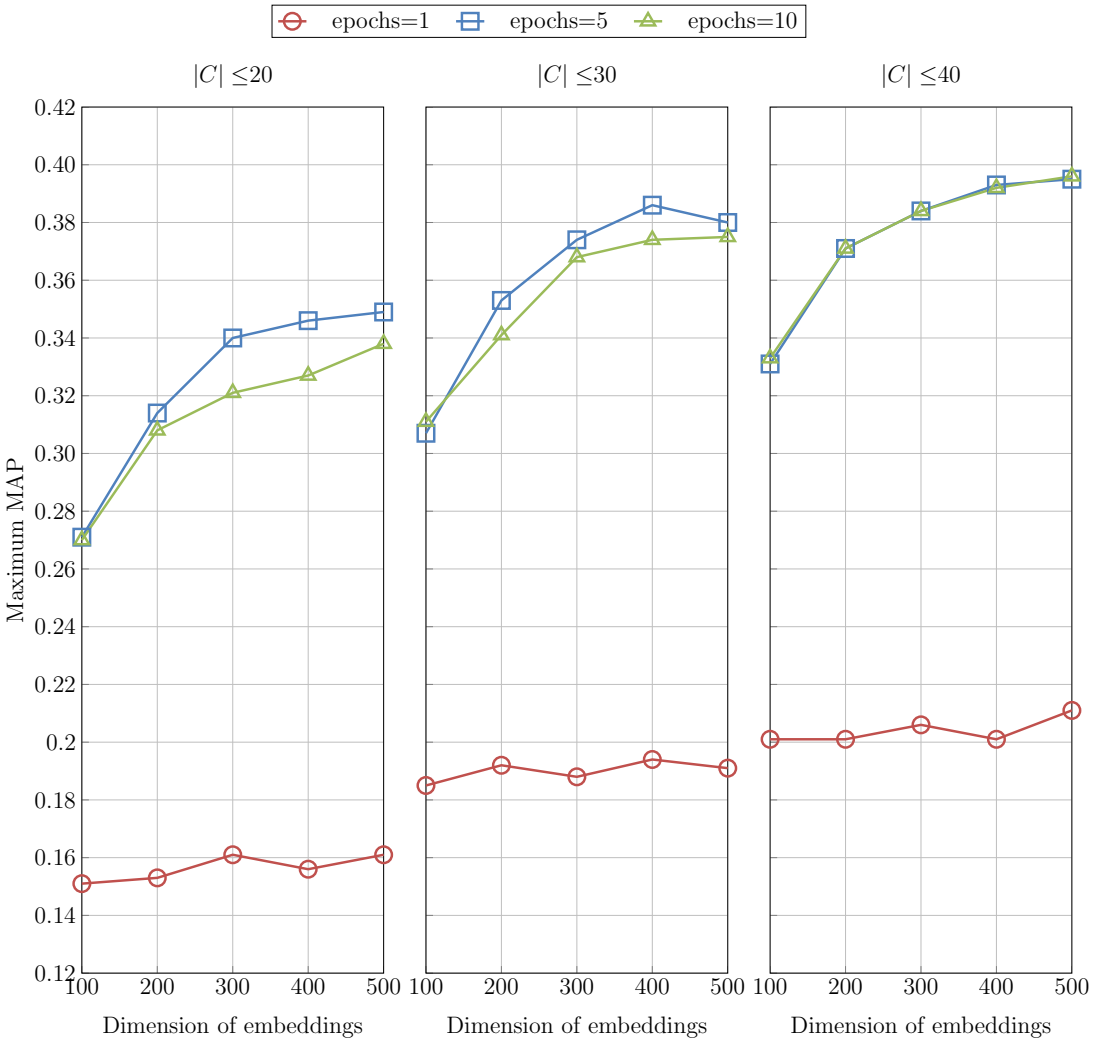


Figure 5.1: Variation in maximum MAP of retrieval of OOV PNs using CBOW Method I, with different embedding sizes K, (maximum) context length $|C|$ and number of training epochs. Evaluated on the validation set.

Figure 5.2: Variation in maximum MAP of retrieval of OOV PNs using Skip-gram Method I, with different embedding sizes K, (maximum) context length $|C|$ and number of training epochs. Evaluated on the validation set.

5.2). This suggests that the Skip-gram model word embeddings must be well separated in the embedding space, and this separation is less affected by the model hyper-parameters. More importantly the model is able to compose the document context reliably with a sum composition, at least for the non-video/LVCSR news transcriptions in the validation set.

The word embeddings of 400 dimension do not give the best MAP performance with Method II, especially for the Skip-gram model. But its MAP performance is still quite close[2] to that of the model with the best MAP. For sake of consistency we choose both CBOW and Skip-gram models with 400 dimensional word embeddings. Similar to Method I, larger context window size gave better performance for both the models. Comparing the number of training epochs, the MAP performance of the CBOW model stops improving after 10 epochs, while that of the Skip-gram model continues to improve until 25 epochs.

Based on the MAP performances of Method II on the validation set, CBOW and Skip-gram models trained with a context window size of 40, and until 10 and 25 epochs respectively, are chosen as the best performing word embedding model configurations.

5.2.2 Retrieval results achieved with the best model configurations

Figure 5.5 shows the recall and MAP performance obtained with CBOW and Skip-gram models. It shows the recall and MAP performance obtained on the reference transcription of the *Euronews* audio test set (on the left) as well as automatic transcriptions obtained from the ANTS LVCSR system (in the middle) and KATS LVCSR system (on the right). Figure 5.5 also shows the performance of the LSA and LDA model with 400 dimensional semantic and topic space representations. The MAP@128 (maximum MAP) for LSA, LDA, CBOW and Skip-gram models is also presented in Table 5.2. Comparing the recall and MAP performances of the different models, we can make the following observations:

- All word embedding based methods, except for Method I with CBOW (CBOW-MI), give better MAP performance than the previous best MAP obtained with Method I on LSA semantic space (LSA-MI). The poor performance of CBOW-MI could be connected to the problem of faster overfitting in CBOW, which we attributed to the larger context windows as discussed in Section 5.2.1.1.
- The Method II MAP performance for both CBOW and Skip-gram models seems to be relatively robust to LVCSR errors, as compared to that for the

[2]statistically significant difference but with a low p-value

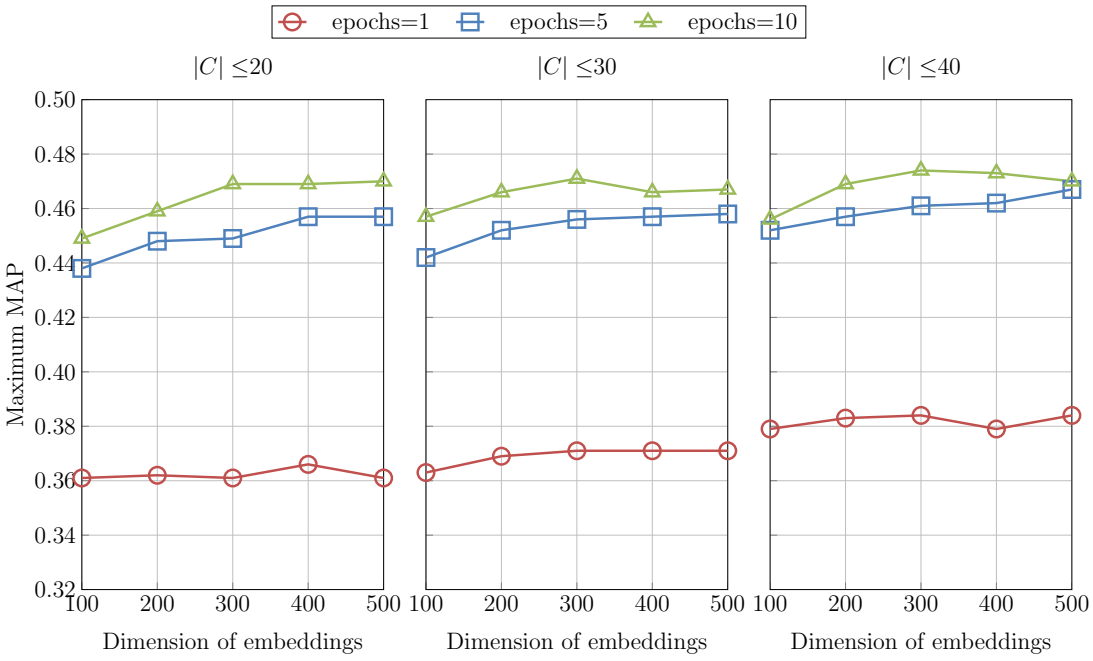


Figure 5.3: Variation in maximum MAP of retrieval of OOV PNs using CBOW Method II, with different embedding sizes K, (maximum) context length $|C|$ and number of training epochs. Evaluated on the validation set.

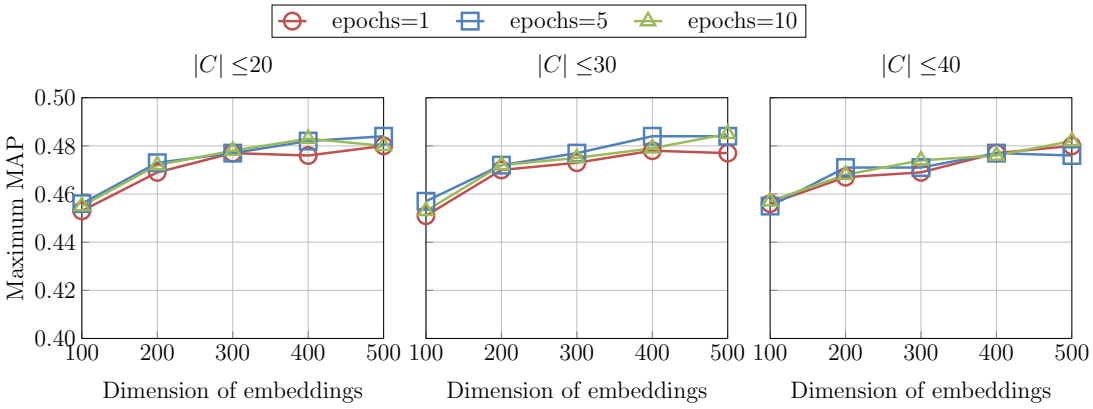


Figure 5.4: Variation in maximum MAP of retrieval of OOV PNs using Skip-gram Method II, with different embedding sizes K, (maximum) context length $|C|$ and number of training epochs. Evaluated on the validation set.

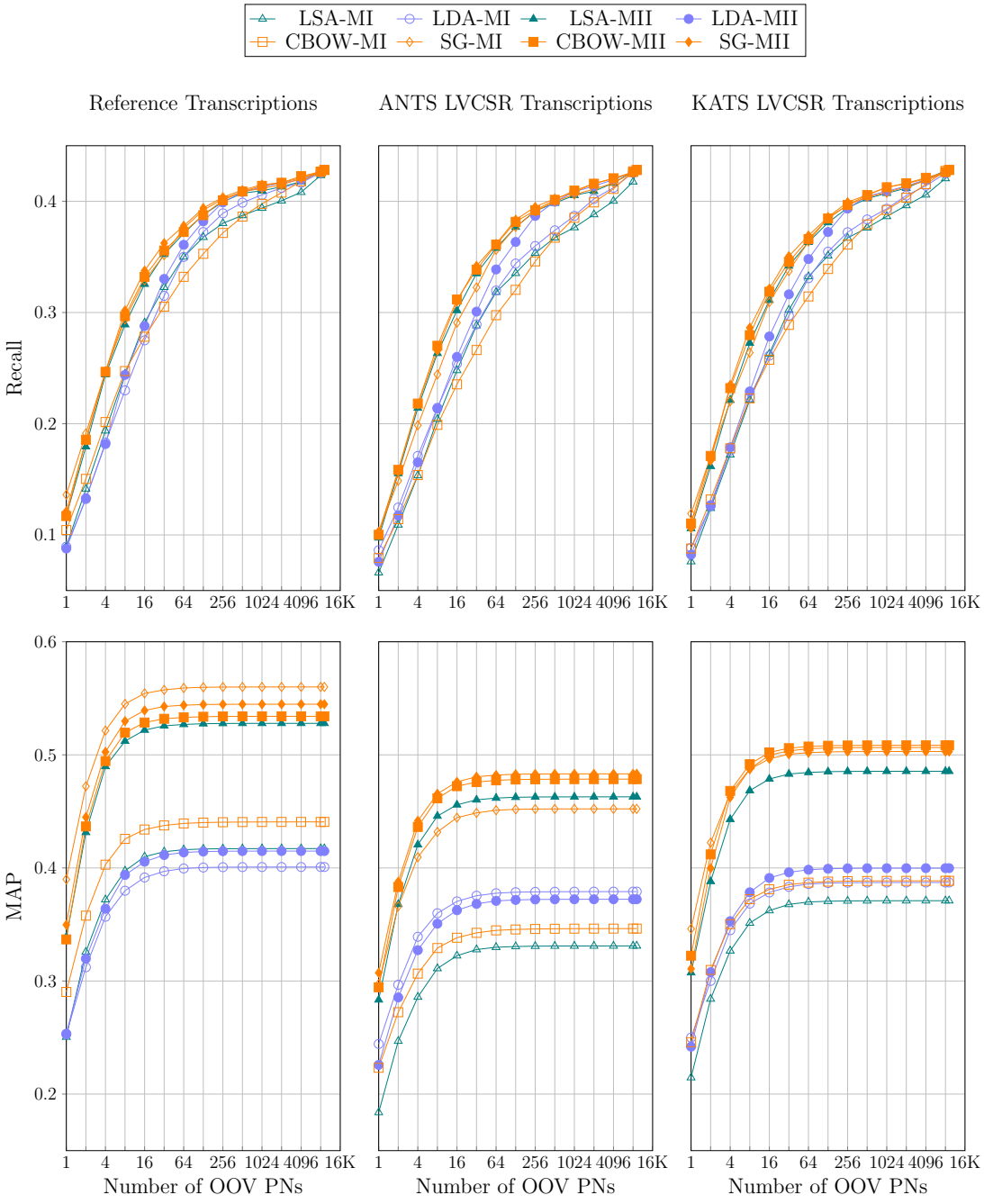


Figure 5.5: Recall and MAP performance of OOV PN retrieval, using CBOW and Skip-gram word embeddings, on *Euronews* audio test set.

Method I performance for these models. The drop between the MAP values for reference and ANTS transcriptions is 10.4% and 11.2% relative, for the CBOW (CBOW-MII) and the Skip-gram (SG-MII) models respectively. For LDA Method II (LDA-MII) this drop is 10.1% relative.

- The Method I MAP performance for both CBOW and Skip-gram models is highly affected by LVCSR errors. The difference between the MAP values for reference and ANTS transcriptions is 21.3% and 19.2% relative, for the CBOW (CBOW-MI) and the Skip-gram (SG-MI) models respectively. For LDA Method I (LDA-MI) this drop is only 5.25% and that for the LSA model (LSA-MI) it is 20.6% relative.

- The recall curves for the highest MAP performance method SG-MI appears to be overlapping with that of some other methods/models. However, it must be noted that SG-MI achieves much more OOV proper names in the top 3 ranks, suggesting that it is a much better retrieval method. As mentioned earlier, this improvement is captured by the MAP curves.

Table 5.2: Comparison of MAP@128 for LSA, LDA, CBOW and Skip-gram models. (The best model is highlighted in bold. * denotes statistically insignificant difference compared to the best model.)

	Reference Transcription	ANTS Transcription	KATS Transcription
LSA-MI	0.417	0.331	0.371
LSA-MII	0.527	0.462	0.485
LDA-MI	0.400	0.379	0.387
LDA-MII	0.414	0.372	0.399
CBOW-MI	0.534	0.346	0.388
CBOW-MII	0.534	0.478*	**0.508**
SG-MI	**0.560**	0.452	0.506*
SG-MII	0.544	**0.483**	0.502*

5.2.3 Retrieval of Rare and Frequent OOV PNs

Following the discussion in Section 4.3, we would like to analyse the performances of the CBOW and Skip-gram model word embeddings combined with Method I

and Method II for retrieval of rare OOV proper names. Table 5.3 and Table 5.4 present a quantitative evaluation of retrieval of rare versus frequent OOV proper names. Table 5.3 lists the maximum MAP for rare and frequent OOV proper names, as achieved with the two retrieval methods using word embeddings from the CBOW model. Similarly, Table 5.4 lists the maximum MAP achieved for rare and frequent OOV proper names using word embeddings from the Skip-gram model. The corresponding plots depicting rank-frequency distribution for the retrieval are shown in Figure B.1 and Figure B.2 in Appendix B.1.

From the results in Table 5.3 and Table 5.4, we can add the following observations to those made from Figure 5.5:

- In case of CBOW, Method II improves the retrieval of rare as well as frequent OOV proper names and hence the overall MAP improves. Method II helps to alleviate the overfitting problem that affected the performance of the CBOW model with Method I.
- On contrary to the CBOW model, in case of the Skip-gram model the Method I gives a better retrieval of rare as well as frequent OOV proper names. Indicating that it is possible to address both frequent and rare OOV proper names with a global representation of OOV proper names. A similar observation was also made for LSA model with Method I.

Table 5.3: Maximum MAP, for rare and frequent OOV proper names, using the two retrieval methods and word embeddings from the CBOW model. (Best Topic model performance is highlighted in bold.)

Method	Type of OOV PNs	Reference	ANTS	KATS
CBOW-MI	all	0.440	0.346	0.388
CBOW-MII	all	**0.534**	**0.478**	**0.508**
CBOW-MI	rare	0.257	0.183	0.212
CBOW-MII	rare	**0.327**	**0.274**	**0.305**
CBOW-MI	frequent	0.539	0.431	0.478
CBOW-MII	frequent	**0.629**	**0.572**	**0.600**

Table 5.4: Maximum MAP, for rare and frequent OOV proper names, using the two retrieval methods and word embeddings from the Skip-gram model. (Best Topic model performance is highlighted in bold. * denotes statistically insignificant difference compared to the best configuration.)

Method	Type of OOV PNs	Reference	ANTS	KATS
SG-I	all	**0.560**	0.452	**0.506**
SG-II	all	0.544	**0.483**	0.502*
SG-I	rare	**0.382**	**0.289**	**0.337**
SG-II	rare	0.326	0.273	0.294
SG-I	frequent	**0.649**	**0.583**	**0.598**
SG-II	frequent	0.633	0.524	0.576

5.3 Conclusion

In this chapter, we extended our proposed methods for retrieval of OOV proper names to word embeddings learned from the popular CBOW and Skip-gram models. The first method compared OOV proper names and LVCSR hypothesis in the embedding space. The second method relied on multiple document specific representations of OOV proper names in the word embedding space. As the CBOW and Skip-gram models do not provide document representations, we exploited the linearity property of the word embeddings to obtain document representations by performing an average of the constituent word embeddings.

Comparison of the proposed retrieval methods for CBOW, Skip-gram, LSA and LDA models, enables us to draw the following conclusions:

- In terms of MAP of retrieval of target OOV proper names, word embeddings from both CBOW and Skip-gram models outperformed LDA based topic space representations as well as those from LSA, which gave the previous best MAP. However, their MAP is not robust to LVCSR errors.

- When using the OOV proper name representations from the CBOW model, the MAP performance is quite low. This is perhaps due to the overfitting problem of the CBOW model, when it is trained with larger context windows. Proposed retrieval method of using document specific OOV proper name representations helped to improve the MAP performance of the CBOW model.

- The MAP performance of OOV proper name representations from the Skip-gram model re-confirms that it is possible to address both frequent and rare OOV proper names with a global representation of OOV proper names, as it was also observed for LSA-MI. However, the big differences in MAP performance of reference and LVCSR transcriptions are discouraging. This continues to motivate us to learn representations with robustness to the LVCSR errors.

CHAPTER 6

Discriminative Context Representations using Neural Networks

The OOV proper name retrieval methods based on LDA topic space representations showed robustness to LVCSR errors. On the other hand, the methods based on LSA and word embeddings, obtained from CBOW and Skip-gram models, were less robust to LVCSR errors, while they outperformed the LDA based methods. The topic space representations from the LDA model are multinomial distributions learned in an unsupervised manner following a Bayesian parameter estimation setup. Similarly, the CBOW and Skip-gram models learn word vectors with an objective to maximise the average log probability of predicting the center word given the surrounding context words and vice versa. Arguing that these context representations learned in an unsupervised manner are not the most optimal for our task of retrieving OOV proper names, we explore discriminative context representations. The CBOW and Skip-gram models are unsupervised methods but they use a pseudo supervision when predicting their outputs. We exploit this mechanism to **train neural network models which predict OOV proper names**. The training objective will be to maximise the retrieval of relevant OOV proper names, thus learning a discriminative context representation at the hidden layer.

Our methodology is related to the recent approaches for text classification with neural networks. In this context, fully connected feed forward networks [Iyyer et al., 2015, Nam et al., 2014], Convolutional Neural Networks (CNN) [Kim, 2014, Johnson and Zhang, 2015, Wang et al., 2015] and also Recurrent/Recursive Neural Networks (RNN) [Socher et al., 2013, Hermann and Blunsom, 2013, Dong et al., 2014, Tai et al., 2015, Dai and Le, 2015] have been applied. On the one hand, the approaches based on CNN and RNN capture rich compositional information, and have outperformed the state-of-the-art results in text classification; on the other hand they are computationally intensive and require careful hyperparameter selection and/or regularisation [Zhang and Wallace, 2015, Dai and Le, 2015]. As compared to text, LVCSR transcriptions of audio documents are

firstly prone to noise in word sequences due to word errors and secondly have no direct information about the position of OOVs. Hence for our task **we rely on a document level bag-of-words architectures because they are suitable to process LVCSR transcriptions**. Moreover, in contrast to the tasks considered in most state-of-the-art text classification works, our task has a large number of output classes i.e. OOV proper names and the distribution of documents per OOV proper name is very skewed.

In this Chapter, we will present proposed discriminative context models and discuss methods for training these models to achieve improved retrieval performance as well as robustness to LVCSR errors. Apart from the comparison of its OOV proper name retrieval performance to the other models, we will study the evaluation of these models in terms of recovery of the target OOV proper names, as also discussed in [Sheikh et al., 2016c]. Additionally, we also evaluate the effectiveness of the proposed NBOW2 model on standard sentiment and topic classification tasks, as discussed in [Sheikh et al., 2016d].

6.1 Neural Bag-of-Words Model

The first model that we propose takes the in-vocabulary words in the document as input and predicts OOV proper names at the output. This model can also be seen as the AverageVec setup of Section 5.1, with the word vectors being trained to maximise the retrieval of relevant OOV proper names. Interestingly this turns out to be similar to the Neural Bag-of-Words model (NBOW)[1], proposed in [Iyyer et al., 2015].

The Neural Bag-of-Words (NBOW) model [Kalchbrenner et al., 2014, Iyyer et al., 2015] is a fully connected neural network model which takes an input text X containing a set of words w and generates probability estimates for the L output labels. Figure 6.1 shows a schematic representation of the NBOW model. The NBOW model has two hidden layers, one corresponding to the input (W^I) and the other for the output (W^O). The first hidden layer W^I is a $[V \times K]$ matrix containing K dimensional vectors corresponding to each of the words in the chosen input vocabulary of size V. With a sparse BOW input vector, with words present in the input set to 1 and others set to 0, the vector-matrix product at first hidden layer translates into a sum of the vectors corresponding to input

[1]Our initial work in [Sheikh et al., 2015c] reported this model with the name Document Continuous Bag-Of-Words (D-CBOW) due to its resemblance to the CBOW model. However, later we switched the name from D-CBOW to NBOW because the name NBOW was used in the text classification work [Iyyer et al., 2015], which appeared around the same time.

words. In practice the average of the word vectors is used instead, as follows:

$$z = \frac{1}{|X|} \sum_{w \in X} v_w \tag{6.1}$$

The average vector z is then fed into the output layer to estimate probabilities for the output labels as:

$$\hat{y} = \text{softmax}(zW^O + b) \tag{6.2}$$

where W^O is the $[K \times L]$ matrix corresponding to the output layer with b as a bias vector and $\text{softmax}(l) = \exp(l)/\sum_{j=1}^{L} \exp(l_j)$.

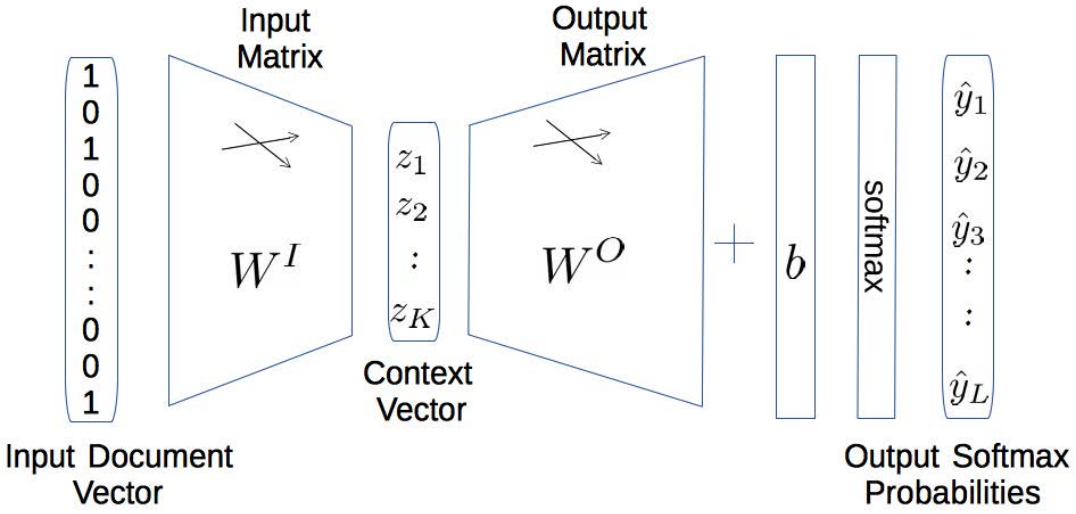


Figure 6.1: Neural Bag-of-Words (NBOW) Model.

In our task to retrieve relevant OOV proper names, the input (word embedding) matrix W^I has vectors corresponding to the in-vocabulary words & proper names ($W^I \equiv \{v_1, v_2, v_3 ... v_V\}$). The output matrix W^O has vectors corresponding to OOV proper names. The input is a sparse BOW vector with 1's representing the in-vocabulary words and proper names present in a training/test document. The average vector $z \equiv \{z_1, z_2 ... z_K\}$ represents the context vector for the document. A vector-matrix product between the average/context vector and the output/OOV proper name matrix (W^O) is equivalent to comparison of the input document and the OOV proper names in the context space.

For the retrieval of relevant OOV proper names, the words from the LVCSR hypothesis are given at the input and the softmax probabilities at the output are

used as scores to rank the OOV proper names. During training of the model, the words in a document from the diachronic text corpus are given at the input and each co-occurring OOV proper name in the document is set at the output in turns. The NBOW model is trained to minimise the categorical cross-entropy loss [Goldberg, 2015]. The categorical cross-entropy error function is commonly used for single label classification and some documents can have more than one OOV proper name. In this case the training document is replicated for each OOV proper name. It has been shown that using cross-entropy error leads to better classification and faster convergence than the pairwise error function which tries to minimise the ranking loss in multi label classification [Nam et al., 2014].

Similar to the CBOW model, the input and output word embeddings W^I, W^O are the unknown parameters in the NBOW model which are to be learned from the training data. The training for these model parameters is carried out using back propagation and gradient descent based learning methods [LeCun et al., 1998]. The training objective is to minimise the cross entropy error between true labels y and predicted labels $\hat{y}$, calculated for a batch of M training samples as:

$$\mathfrak{L} = -\frac{1}{M}\sum_{m=1}^{M}\left(\sum_{j=1}^{L} y_{mj}\ log(\hat{y}_{mj})\right)\ : y_{mj} \in \{0,1\} \tag{6.3}$$

During training, we consider one OOV proper name per document at a time and only one output is set to one and others to zero. This function then becomes equivalent to the negative log likelihood for prediction of OOV proper names:

$$\mathfrak{L} = -\frac{1}{M}\sum_{m=1}^{M} log(\widehat{y}_m) \tag{6.4}$$

With this loss function the equations for updating W^I, W^O are similar to those for the CBOW model, as elaborated in [Rong, 2014]. Other specific details and techniques adopted for training the NBOW model are discussed in Section 6.3.

6.2 Neural Bag-of-Weighted-Words (NBOW2) Model

The NBOW model learns discriminative word and context vector representations specialised for the retrieval task. However we feel that it fails to *explicitly* model the information that certain words are more important than others for retrieval of an OOV proper name. Therefore we propose the Neural Bag-of-Weighted-Words (NBOW2) model, with the motivation of enabling the NBOW model to learn and use word importance weights which can attribute an OOV proper name. This

idea was inspired by the works on learning to pay attention in a sequence of input, which became popular with the neural network machine translation architecture proposed in [Bahdanau et al., 2014] and was later applied in speech [Chan et al., 2015], image [Xu et al., 2015] and protein sequence analysis [Sønderby et al., 2015]. [Ling et al., 2015] also proposed the use of different word weights in a bag-of-word neural network model. However, they use word position based weights to improve vectors learned by the CBOW model. As it will be discussed, our NBOW2 model learns to assign task specific word importance weights based on distances in the word embedding space.

To learn these word importance weights, we introduce a weighted sum composition of the input word sequence X as follows.

$$z = \frac{1}{|X|} \sum_{w \in X} \alpha_w \, v_w \tag{6.5}$$

where α_w are scalar word importance weights for each word $w \in X$. The weights α_w are obtained by integrating a new K dimensional vector a into the model, and using the following operation:

$$\alpha_w = f(v_w \cdot a) \tag{6.6}$$

where $(\cdot)$ represents the dot product and f scales the importance weights to $[0, 1]$. The word importance weight α_w is a function of the distance of that word w from a in the context space. This ensures that the calculation of α_w takes into account the contextual word similarities and it is not biased by the frequency of occurrence of words in the training corpus. We believe that the vector a, which is itself learned and updated along with the word vectors, will act as a reference for separation and composition of the word vectors into a context vector. Regarding the function f, common activation functions can be used such as sigmoid, softmax (as in [Sheikh et al., 2015c]) and even hyperbolic tangent. In our experiments we found that the sigmoid function $f(x) = (1 + e^{-x})^{-1}$ is a better choice in terms of convergence speed and accuracy. We present a more elaborate discussion on the choice of f in Section 6.4.

Figure 6.2 shows a schematic representation of the NBOW2 model. The inputs, input embedding matrix, outputs and the output matrix are similar to that of the NBOW model. However, the procedure to obtain the document context vector has changed. After the lookup of the word vectors for input text, a dot product is performed between each input word vector and the vector a. The scalar values from the dot product are then passed through the function f. The resulting scalar word importance weights are multiplied with the input word vectors and a weighted sum composition representing the document context vector is obtained.

Similar to the NBOW model, the NBOW2 model is trained to minimise the categorical cross-entropy loss, given by Equation 6.4. However, as the forward propagation obtains the context vector z using a weighted sum combination of the input words with Equation 6.5 and Equation 6.6, the backward propagation and the update equations will involve the vector a. The parameter update equations for the NBOW2 model are similar to those for the CBOW model [Rong, 2014].

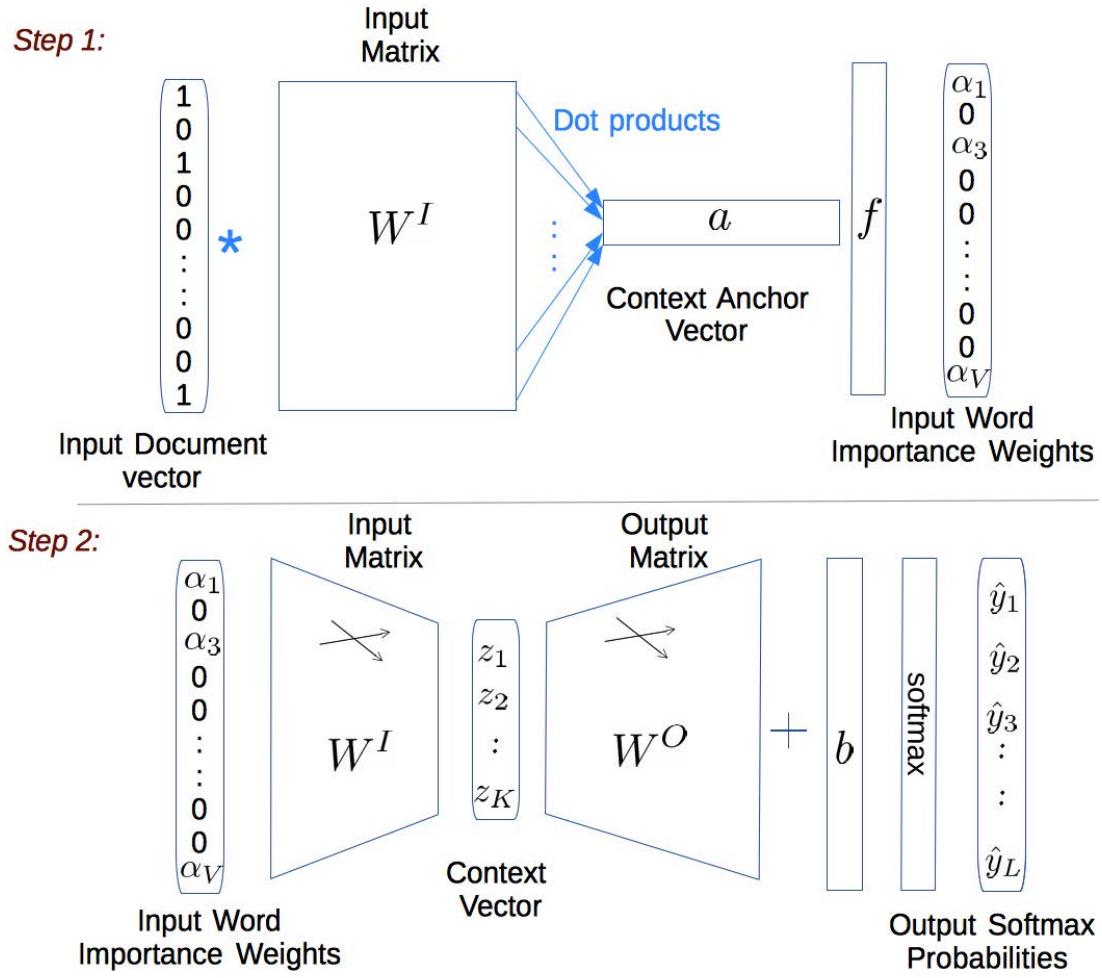


Figure 6.2: Neural Bag-of-Weighted-Words (NBOW2) Model.

6.2.1 Combination of the NBOW and NBOW2 Models

We further propose the NBOW2+ model in which the NBOW and NBOW2 document context vectors are concatenated together. The motivation behind

this combination is to enhance the averaged context representation, where all words are equally important, with the specificity of more important words from weighted average context and vice versa. NBOW2+ has two input matrices (W_1^I, W_2^I) and hence maintains two K dimensional word vectors v_w^1 and v_w^2 for each input word w. It has one K dimensional anchor vector a similar to NBOW2 and one matrix W^O and one bias vector b in the output layer. The document context vector z is obtained as the concatenation of two context vectors z_1 and z_2 as follows:

$$\begin{aligned} z_1 &= \frac{1}{|X|} \sum_{w \in X} v_w^1 \\ z_2 &= \frac{1}{|X|} \sum_{w \in X} \alpha_w \; v_w^2 \\ z &= [z_1 \; z_2] \end{aligned} \tag{6.7}$$

As the document context vector is concatenation of the two K dimensional context vectors, the output layer parameters (W^O, b) have a dimension of $2K$. The training procedure for the NBOW2+ model is the same as for the NBOW and NBOW2 models.

6.3 Training the NBOW group of models

In this section we discuss in general the choices made for training the NBOW, NBOW2 and NBOW2+ models. It includes some crucial hyper-parameters which can affect the retrieval performance significantly. A more model specific discussion and comparison is made in Section 6.4.2.

6.3.1 Initialisation

It is well known that good initialisation and pre-training of hidden layer weights are crucial for training deep neural networks [Larochelle et al., 2009, Goldberg, 2015]. While the NBOW model is not deep, we examined if initialisation is crucial and if it affects the performance of the NBOW model in our task. We will present the results for the NBOW model with input word vectors (W^I) initialised (a) randomly and (b) with Skip-gram word vectors pre-trained on the diachronic text corpus. The vectors corresponding to output OOV proper names (W^O) are randomly initialised. Initialising W^O with Skip-gram word vectors did not give any significant performance improvements. In our results and discussions the prefixes 'RAND-' and 'Sg-' will be used to denote models with random and Skip-gram initialisation of word vectors, respectively.

6.3.2 Full Training v/s Two Phase Training

We explore two methods of training the NBOW, NBOW2 and NBOW2+ models: (a) *full training* and (b) *two phase training*. In full training all the network parameters including the input matrix, output matrix, output bias vector of NBOW model (c.f. Section 6.1), and additionally the anchor vector for NBOW2 and NBOW2+ models (c.f. Section 6.2 and 6.2.1), are trained and updated using back-propagation.

The two phase training method has a first training phase in which only the output parameters (W^O, b), and the vector a for the NBOW2 and NBOW2+ models, are updated by keeping the input word vectors fixed to pre-trained Skip-gram word vectors. In the second training phase all the model parameters including the word vectors are updated. The motivation behind the two phase training is again a better initialisation and convergence. The first training phase is supposed to take the randomly initialised output parameters to a better state for simultaneously training all the network parameters. In our results and discussions the suffixes '-1p' and '-2p' will be used to denote models trained in one and two training phases, respectively.

6.3.3 Learning Rate and Stopping Criteria

All the NBOW models are trained using a stochastic gradient descent algorithm with ADADELTA [Zeiler, 2012]. ADADELTA provides an adaptive per-dimension learning rate for gradient descent and is robust to noisy gradient information. We tested two values of the ADADELTA decay constant (ρ), 0.99 and 0.95, and used ρ =0.99 in all our experiments as it gives a lower validation error rate and a better retrieval performance.

To control the training of all the NBOW models an early stopping criterion [Bengio, 2012] based on the validation set error is used. Early stopping is used in full training as well as both the first and the second training phases of two phase training[2].

6.3.4 Dropout at Input

Dropout is a technique, adopted for training deep neural networks, in which the output of randomly selected units in the network is set to zero [Srivastava et al.,

[2]Using a fixed number of epochs in the first phase of the two phase training did not give a better performance.

2014]. The dropout technique has been shown to significantly reduce overfitting and give major improvements over other regularisation methods in deep neural networks. While the NBOW model and the proposed NBOW2 and NBOW2+ models are not deep architectures we are interested to analyse if the dropout mechanism helps us to avoid overfitting and add robustness to the document level BOW input. We specifically applied dropout at the input layer. The dropout at the input layer is equivalent to dropping words in the input. The motivation behind this word dropout is (a) to synthesise variations of document context using training set documents and (b) to simulate deletion errors in LVCSR hypothesis. Based on the experiments on the validation set we chose a word dropout probability of 0.9 (from among 0, 0.25, 0.5, 0.75 and 0.9)[3]. We found that word dropout has been recently tried and gave improvements in text classification tasks [Dai and Le, 2015, Iyyer et al., 2015].

6.4 Experiment Results and Discussion

In this section we first present a comparison of recall and MAP performance of the NBOW, NBOW2 and NBOW2+ models with the methods discussed in previous chapters. As the LSA, LDA, CBOW and Skip-gram models gave the best performance with 400 dimensional word and context vectors, we set the embedding size of the NBOW group of models to 400. The recall and MAP performance comparison will be followed by a detailed discussion on the learning and performance improvements of the NBOW, NBOW2 and NBOW2+ models.

6.4.1 OOV Proper Name Retrieval Performance

Figure 6.3 shows the recall and MAP performance of retrieval of OOV proper names for different methods. As before, the graphs shown are for the reference transcriptions (left), LVCSR transcriptions from ANTS (middle) and the LVCSR transcriptions from KATS (right) for the *Euronews* test set audio. The X-axis represents the number of OOV proper names selected from the diachronic text corpus i.e. the 'N' in the top-N retrieved results. The Y-axis represents recall (top) and MAP (bottom) of the target OOV proper names. For each of the methods, the models giving best performance on validation set are chosen. The MAP@128 (maximum MAP) values are also presented in Table 6.1.

The recall and MAP retrieval performance for NBOW, NBOW2 and NBOW2+

[3]Word dropout probability p does not necessarily translate to leaving out p% of the input words in our implementation.

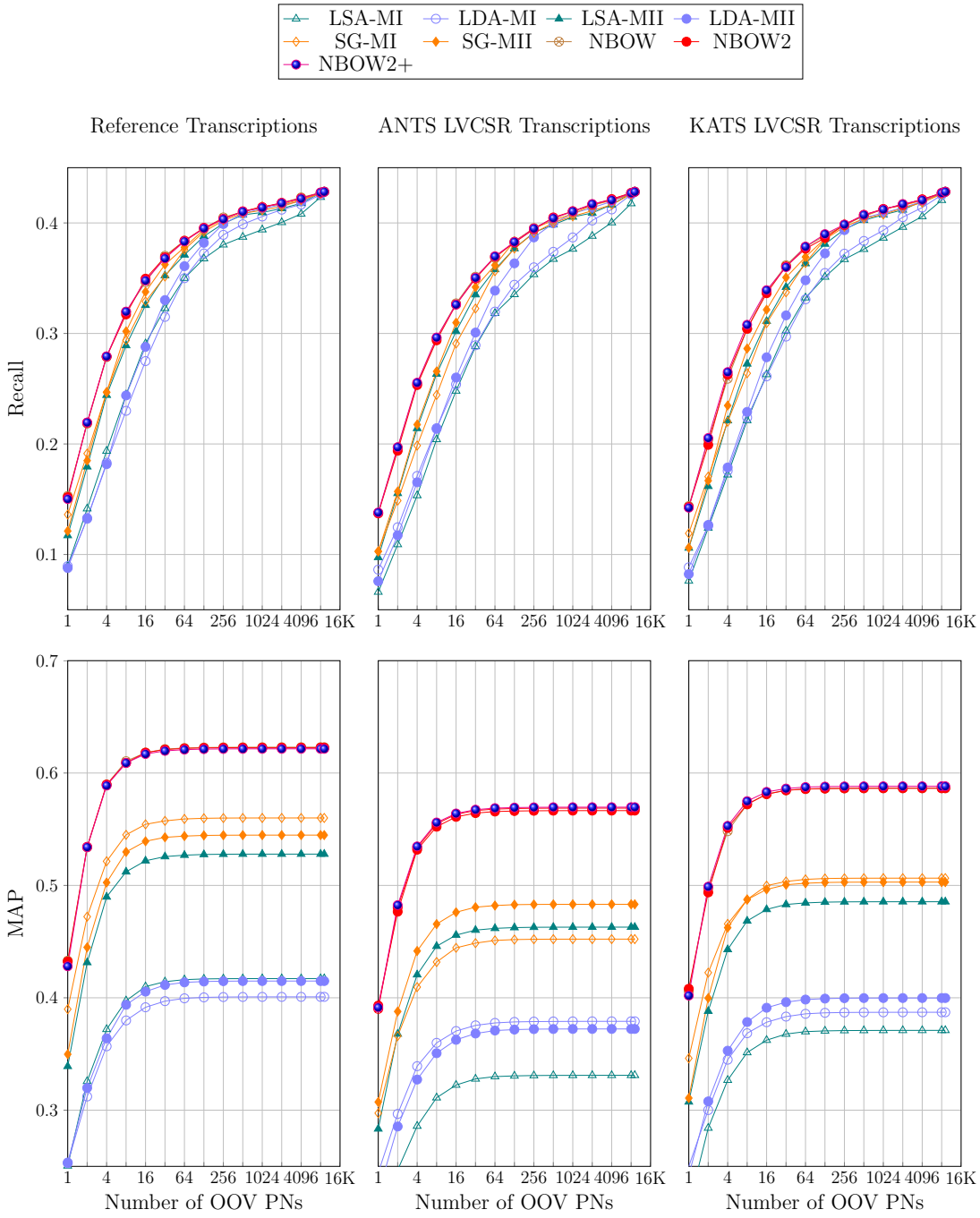


Figure 6.3: Recall and MAP performance of NBOW, NBOW2 and NBOW2+ models for OOV proper name retrieval on *Euronews* audio test set. The NBOW, NBOW2 and NBOW2+ models were initialised with Skip-gram word vectors and trained in two phases (c.f. Section 6.3.2 and Table 6.2). The NBOW and NBOW2 graphs are overlapped by that of NBOW2+.

Table 6.1: Comparison of MAP@128 for LSA, LDA, Skip-gram and NBOW group of models. (The best model is highlighted in bold. * denotes statistically insignificant difference compared to the best model.)

	Reference Transcription	ANTS Transcription	KATS Transcription
LSA-MI	0.417	0.331	0.371
LSA-MII	0.527	0.462	0.485
LDA-MI	0.400	0.379	0.387
LDA-MII	0.414	0.372	0.399
SG-MI	0.560	0.452	0.506
SG-MII	0.544	0.483	0.502
NBOW	0.622*	0.568*	0.586*
NBOW2	0.622*	0.566*	0.586*
NBOW2+	**0.621**	**0.569**	**0.588**

models is similar and their graphs are overlapping. We will discuss in detail in Section 6.4.2 about the difference in performance of the NBOW2 and NBOW2+ models as compared to the NBOW model. Overall the three models clearly outperform the methods based on LSA, LDA and Skip-gram in terms of recall and MAP, both for reference and LVCSR transcriptions. In terms of robustness to LVCSR errors, the % reduction in the MAP values of ANTS transcriptions (compared to that of reference transcriptions) with NBOW, NBOW2 and NBOW2+ model is 8.6%, 9.0% and 8.3% (relative) respectively. Where this reduction for Method I of Skip-Gram (SG-MI), LSA (LSA-MI) and LDA (LDA-MI) is 19.2%, 20.6% and 5.25% (relative) respectively.

6.4.2 Scrutinising the Training of NBOW models

In this section, we first analyse how the choice of training conditions, namely (a) word dropout and (b) two phase training, affect the performance of the NBOW model. We present Table 6.2 for this discussion. Then with the help of Figure 6.4 and Table 6.3 we compare the training convergence and retrieval performance of the NBOW, NBOW2 and NBOW2+ models. Alongside we will also compare performance of random versus Skip-gram embedding initialisation of the NBOW model, as well as for the NBOW2 and NBOW2+ models.

6.4.2.1 Robustness with Word Dropout

The effect of applying word dropout can be observed from Table 6.2. It is clear that word dropout improves the MAP; higher dropout rate giving higher MAP. We can observe that the NBOW model initialised with Skip-gram word vectors (Sg-1p) takes a smaller number of training epochs and gives better MAP performance than the NBOW model with random initialisation (Rand-1p). However, applying word dropout gives larger relative improvements in Rand-1p as compared to Sg-1p.

For instance the MAP value for reference transcriptions i.e. MAP-TR improves by 15% for Rand-1p and by 6.75% for Sg-1p for word dropout of 0.9 as compared to no word dropout. Secondly, we can observe that the improvement in MAP with word dropout is relatively larger for LVCSR transcriptions. For instance if we compare the MAP value for reference and ANTS LVCSR transcriptions i.e. MAP-TR and MAP-TA the improvements are 15% v/s 25% for Rand-1p, 6.75% v/s 11.8% for Sg-1p and 3.3% v/s 8.2% for Sg-2p. These **improvements validate our speculation that word dropout would enhance performance by simulating variations in document context as well as deletion errors in the LVCSR hypothesis**.

6.4.2.2 Two phase training and the improvement with NBOW2+ model

In Section 6.3.2 we proposed to train the NBOW models in two phases. The MAP results in Table 6.2 show that the best retrieval performance is obtained with this two phase training method. However, it takes a larger number of training epochs compared to training the NBOW model in one phase (Sg-1p). With the help of Figure 6.4, we show that this problem can be addressed by the NBOW2+ model.

Figure 6.4 shows a graph of validation set errors of the NBOW, NBOW2 and NBOW2+ models, as training progresses. It can be observed that all three models (NBOW, NBOW2 and NBOW2+) converge to a similar point but at different convergence rates. While both NBOW and NBOW2 models take a larger number of training epochs, **the NBOW2+ model gives a faster convergence without compromise in error rate**. This can be seen in Table 6.3, which compares the MAP achieved by the NBOW, NBOW2 and NBOW2+ models.

As a counter experiment we examined if the ADADELTA decay constant (ρ) can speed up the two phase training convergence. We observed from our experiments that the ADADELTA decay constant (ρ) of 0.95 takes fewer training epochs as compared to a decay constant (ρ) of 0.99, but at the cost of reduced MAP performance. For instance with word dropout of 0.9, the 400 dimensional

Table 6.2: Maximum MAP for retrieval of OOV proper names, obtained by the NBOW model (400 dimension word vectors) trained with an early stopping criterion. Suffixes V, TR, TA and TK denote the performance on the validation set, the reference transcription test set and the ANTS LVCSR and KATS LVCSR transcriptions of the test set respectively. Rand and Sg denote random and Skip-gram word vector initialisation. 1p and 2p denote one and two phase training. The best configuration is highlighted in bold. * denotes statistically insignificant difference compared to the best configuration.

		word dropout probability (p)				
		0.0	0.25	0.5	0.75	0.9
Rand-1p	epochs	175	217	249	320	276
	MAP-V	0.458	0.482	0.502	0.537	0.530
	MAP-TR	0.500	0.522	0.549	0.578	0.576
	MAP-TA	0.419	0.435	0.464	0.505	0.526
	MAP-TK	0.457	0.473	0.500	0.533	0.542
Sg-1p	epochs	112	147	152	149	155
	MAP-V	0.511	0.522	0.535	0.541	0.543
	MAP-TR	0.563	0.569	0.576	0.587	0.601
	MAP-TA	0.491	0.483	0.502	0.531	0.549
	MAP-TK	0.523	0.522	0.532	0.551	0.566
Sg-2p	epochs	481	482	398	417	410
	MAP-V	0.551	0.553	0.562	0.574	**0.585**
	MAP-TR	0.602	0.598	0.605	0.615*	**0.622**
	MAP-TA	0.525	0.519	0.533	0.561*	**0.568**
	MAP-TK	0.555	0.552	0.561	0.578*	**0.586**

Table 6.3: Maximum MAP for retrieval of OOV proper names, obtained by the NBOW, NBOW2 and NBOW2+ models with 400 dimension word vectors trained with word dropout probability $p = 0.9$ and an early stopping criterion. Suffixes V, TR, TA and TK denote the performance on the validation set, reference transcription in test set and the ANTS LVCSR and KATS LVCSR transcriptions of the test set respectively. Rand and Sg denote random and Skip-gram word vector initialisation. 1p and 2p denote one and two phase training. The best configuration is highlighted in bold. * denotes statistically insignificant difference compared to the best configuration.

		NBOW	NBOW2	NBOW2+
R-1p	epochs	276	123	210
	MAP-V	0.530	0.474	0.519
	MAP-TR	0.576	0.507	0.574
	MAP-TA	0.526	0.402	0.526
	MAP-TK	0.542	0.440	0.546
Sg-1p	epochs	155	166	161
	MAP-V	0.543	0.541	0.547
	MAP-TR	0.601	0.599	0.601
	MAP-TA	0.549	0.549	0.545
	MAP-TK	0.566	0.566	0.566
Sg-2p	epochs	410	648	273
	MAP-V	0.585*	0.587*	**0.593**
	MAP-TR	**0.622**	**0.622**	0.621*
	MAP-TA	0.568*	0.566*	**0.569**
	MAP-TK	0.586*	0.586*	**0.588**

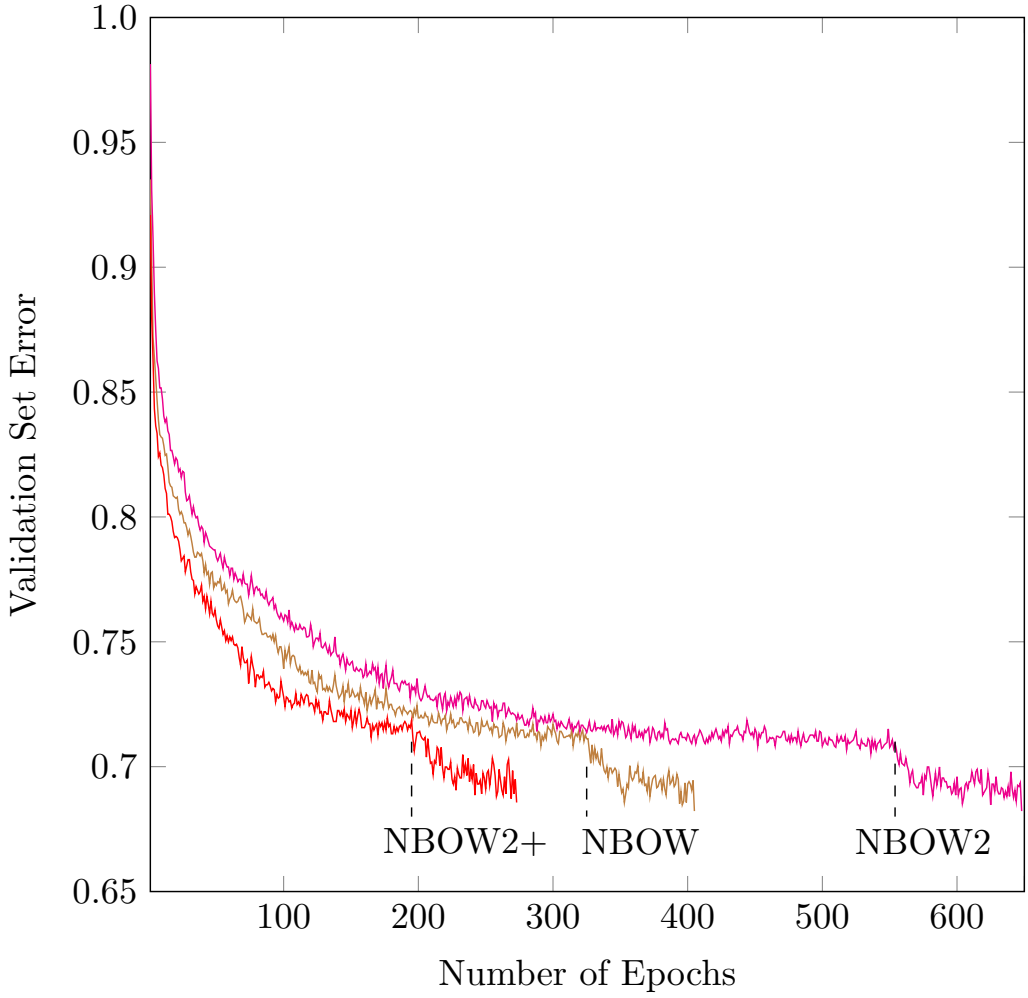


Figure 6.4: Validation set errors during the two phase training of NBOW, NBOW2 and NBOW2+ models. (400 dimension word vectors and 0.9 word dropout probability. − − − markers indicate end of first and begin of second training phase)

NBOW model takes 351 epochs and achieves a maximum MAP of 0.5 as compared to 410 epochs and 0.568 MAP obtained with $\rho = 0.99$.

From these experiments we can conclude that (a) two phase training leads to better retrieval performance with the NBOW and NBOW2 models but it requires a longer training and (b) the NBOW2+ model, which combines the average and weighted average contexts of NBOW and NBOW2 models, can significantly reduce this training time without compromise in the MAP performance.

6.4.3 Word Importance weights of the NBOW2 model

We present Figure 6.5 to discuss about (a) the scalar word importance weights α_w learned by the NBOW2 model and (b) the choice of the function f for the NBOW2 model (see Equation (6.6)). Figure 6.5 shows a graph of the importance weights of words in a document from the test set. The top graph of Figure 6.5 shows the weights assigned by the NBOW2 model with f as sigmoid activation and the bottom graph shows the weights assigned with f as softmax activation.

Firstly it is clear from these graphs that the NBOW2 model learns and assigns different degrees of importance for different words. For example this test document is about the accident of Formula one driver Michael Schumacher and it has a missing OOV proper name *'Kehm'* (*Sabine Kehm* is the spokesperson for Michael Schumacher). If we analyse the list of words as per the top graph the top four important words are *michael*, *formule*, *critique* and *hospitaliser* and the four least important words are *rester*, *tenir*, *monde* and *présent*[4]. From this example, it is evident that the NBOW2 model assigns higher weights to words which are important for retrieval of the OOV proper name. The same holds true for the NBOW2 model with softmax f.

The second observation is that the NBOW2 model with f as **softmax tends to assign higher weights to fewer words** and a weight close to zero most of the other words. While this feature could help in tasks like selection of keywords in written texts, it leads to a relatively bad OOV proper name retrieval performance [Sheikh et al., 2015c]. We hypothesise that this happens because the NBOW2 model with softmax f ignores (or gives low importance value to) too many words from the input which affects its discriminative ability, especially when (a) the LVCSR hypothesis has many word errors and (b) the document contains OOV proper names from different contexts, for instance both sports and politics.

6.4.4 Document Specific Representation of OOV PNs

The NBOW, NBOW2, NBOW2+ models perform an average composition or/and weighted sum composition of the words in the input document. This composition, denoted by z in the Equations 6.1, 6.5 and 6.7, represents the document context vector. Thus we can use our proposed methodology based on document specific representations, as discussed in Section 5.1, to retrieve OOV proper names. When applying this method (Method II) to NBOW, NBOW2 and NBOW2+ models these models are trained as discussed previously. Then the document

[4]English translations: *michael, formula, critical, hospitalise, remain, stay, world, present*

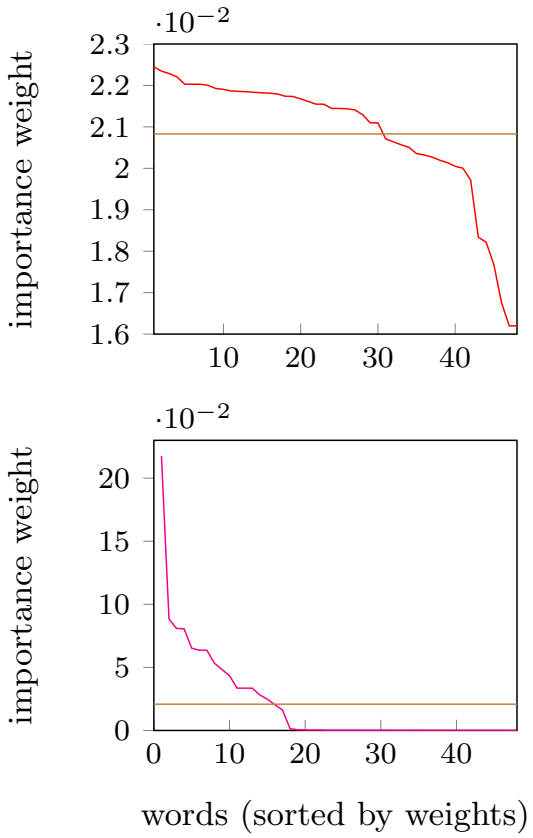


Figure 6.5: Distribution of word importance weights assigned by the NBOW2 model in a sample document with 48 words. Two variations of the NBOW2 model are shown: (top) f as sigmoid and (bottom) f as softmax. Horizontal lines denote the all equal weights ($1/48 = 0.0208$) in the simple average by the NBOW model.

context vector z is obtained for each of the diachronic corpus documents and the documents in the test set.

Table 6.4 presents the maximum MAP achieved with the NBOW, NBOW2 and NBOW2+ models using Method II. The first observation is that the MAP performance of these models is lower as compared to that obtained by their equivalent Method I (see Table 6.3). It must be noted that although the NBOW, NBOW2 and NBOW2+ models are trained discriminatively, they heavily rely on the OOV proper name representations in the output layer of these models. So their performance is not as good as their performance with a forward pass (Method I, Table 6.3). However the NBOW, NBOW2 and NBOW2+ models perform better when comparing the MAP for LVCSR transcriptions of the LSA (LSA-MII) and Skip-gram (SG-MII) models, which gave the best MAP with Method II before. As highlighted in Table 6.4, the NBOW2 model gives best MAP for both ANTS and KATS LVCSR transcriptions.

Table 6.4: Comparison of maximum MAP obtained using document level representations. The best MAP is highlighted in bold. * denotes statistically insignificant difference compared to the best configuration.

	Reference	ANTS	KATS
LSA-MII	0.527	0.463	0.485
SG-MII	**0.544**	0.483	0.502*
NBOW-MII	0.524	0.492*	0.505*
NBOW2-MII	0.527	**0.498**	**0.510**
NBOW2+-MII	0.525	0.487	0.504*

6.5 Recognition of OOV PNs

The list of relevant OOV proper names retrieved by the context model is to be used for recognition/recovery of the missed proper names. In our previous works we evaluated the effectiveness of the list of relevant OOV proper names obtained from the context models by performing a keyword search based recovery. In [Sheikh et al., 2016b] we performed a phonetic search for the top-N relevant OOV proper names in the 1-best LVCSR hypothesis. In [Sheikh et al., 2016c] we performed a *Finite State Transducer* (FST) based keyword search in the LVCSR lattice. Keyword search based recovery enables a faster evaluation but it results

in many false alarms. In this dissertation, we perform a second pass speech recognition to recognise the OOV proper names by updating the LVCSR system with the list of relevant OOV proper names retrieved by the context model.

6.5.1 Updating LVCSR for Recognition of OOV PNs

Updating the LVCSR system for new words requires updating the pronunciation lexicon and the language model (LM) n-gram probabilities. To update the pronunciation lexicon automatic *Grapheme-to-Phoneme* (G2P) converters can be used. We trained the Sequitur G2P converter [Bisani and Ney, 2008] on our original pronunciation lexicon and used it to generate up to 3 pronunciations of each new OOV proper name. Estimating LM probabilities for new words is a non-trivial and open problem. Most of the proposed methods rely on similarity between in-vocabulary and OOV words [Orosanu and Jouvet, 2015, Qin, 2013, Lecorvé et al., 2011] or use word classes in the LM [Allauzen and Gauvain, 2005b, Pražák et al., 2007, Naptali et al., 2012]. In our second pass speech recognition experiments we added OOV proper names as new unigrams without changing the existing unigram probabilities and leaving out the higher order n-grams of OOV proper names. The unigram probabilities are adjusted by taking a part of the $<unk>$ probability and assigning it to an OOV proper name as:

$$p_{oov-pn-unigram} = p_{<unk>} \times \frac{\delta}{\#\text{ OOV PNs}} \tag{6.8}$$

where δ is the fraction of $<unk>$ probability assigned to all the OOV proper names to be added. This approach to add OOV proper names is similar to a class LM with a class in unigrams. (A detailed comparison to other methods is not in the scope of this dissertation.)

6.5.2 Recognition Experiment Setup

Since the second pass speech recognition experiments are to be performed for different OOV proper name lists, we formed a smaller test set for these experiments. From the 3000 *Euronews* videos (see Table 2.1), we formed a subset of videos appearing in 4 randomly selected weeks. This test subset comprises 467 videos, of which 318 videos have one or more proper name missed in the first pass speech recognition as they were OOV. It must be noted that there are 149 videos in this test set with no known OOV proper name; as would be the case in a real setup where it is not known beforehand if the video has OOV(s) or not. The 318 videos contain a total of 1023 OOV proper name (non unique) terms, of which

up to 483 can be recovered with the *L'Express* diachronic corpus. The number of words and proper names to be recognised are 97935 and 5838, respectively.

We perform the second pass speech recognition with our ANTS system since it can easily perform a document specific LM update at runtime[5]. Our baselines will be ANTS one pass speech recognition without knowledge of OOV proper names, denoted as *No-OOV*, and ANTS one pass speech recognition which includes all 9.3K new OOV proper names from *L'Express*, denoted as *LX-All*. We compare these to second pass speech recognition with the ANTS system updated with the top-128 ($\sim$1%) document specific relevant OOV proper names retrieved by the LDA and the NBOW2+ models. These will be denoted as *LDA-128* and *NBOW2+-128*. We chose the point 128 for our analysis because after this point both the recall as well as the MAP curves are flat, and before this point there are big differences in the recall of the different retrieval methods. Moreover, Skip-gram-128 performance is not shown but we found that it is similar to that of LDA. Similarly, we expect that NBOW-128 and NBOW2-128 would perform similar to NBOW2+-128.

The Recall@128 and MAP@128, i.e. the recall and MAP with the top-128 retrieved OOV proper names, for the LDA-128 and the NBOW2+-128 setup are shown in Table 6.5. Since we are using only a subset of the original test set, the Recall@128 and MAP@128 values are different compared to those in Figure 6.3, but NBOW2+ gives a better performance than LDA as observed in Figure 6.3.

Table 6.5: OOV proper name retrieval performance on the test sub-set after the first pass using ANTS LVCSR. (These retrieval results will be used in the second pass recognition.)

	LDA-128	NBOW2+-128
Recall@128	0.37	0.41
MAP@128	0.41	0.62

We used another subset of *Euronews* videos (not part of the test subset) to tune the δ parameter in Equation 6.8. After different trials we chose a value of 0.001 for δ, which gave an optimal performance for each of the methods. A higher value of δ will improve the OOV proper name recognition but also lead to increased false alarms.

We also present the PNER and WER results from an *oracle* setup. In the oracle setup we perform only one pass of ANTS speech recognition using an

[5]The KATS system is based on Kaldi which requires a lengthy ($\sim$6hours) compilation of the LM (HCLG) FST.

updated pronunciation lexicon and LM (using Equation (6.8)). They are specific to each video from the test sub-set and include the OOV proper names which actually appear in the video. For comparison we add only those OOV proper names which can be obtained using the L'Express diachronic text corpus.

6.5.3 Recognition Results

Table 6.6 shows the *Proper Name Error Rate* (PNER) after the second pass speech recognition. PNER is obtained by first aligning the reference and hypothesised word level transcriptions and then calculating substitution, deletion, insertion errors, and thus the error rate only on the proper name terms. Similarly, OOVPNER is the error rate calculated only for OOV proper names.

It can be observed from Table 6.6 that adding all OOV proper names from the diachronic corpus (LX-All) leads to an increased PNER and OOVPNER. The increased error rate is mainly due to insertion and substitution errors, and it can possibly be reduced with better LM update techniques. The LDA and NBOW2+ context models enable selection of relevant OOV proper names and hence recognition of new proper names and a reduction of PNER. While LDA and NBOW2+ models show similar PNER performance, we can see that NBOW2+ gives a lower OOVPNER. The NBOW2+ model leads to more correctly recognised OOV proper names. The performance of NBOW2+ is close to our Oracle setup. After analysing errors in the Oracle setup we hypothesise that automatic G2P pronunciations of OOV proper names is another source of recognition errors. Adding the new proper names into the vocabulary and LM did not have a negative impact on the WER. Instead the WER showed minor improvements of 0.7% and 0.8% absolute for LDA-128 and NBOW2+-128, compared to the No-OOV case having a WER of 41.7%. Improvements in WER were due to recognition of OOV proper names and reduction in insertion and deletion errors.

Table 6.6: Second pass proper name recognition results. PNER denotes Proper Name Error Rate. OOVPNER denotes OOV Proper Name Error Rate. (In LDA-128 and NBOW2+-128, top-128 document specific OOV PNs retrieved by LDA and NBOW2+ models are added to lexicon and LM.)

	No-OOV	LX-All	LDA-128	NBOW2+-128	Oracle
OOV PNs added	0	9.3K	128	128	oracle
% OOVPNER	100.0	117.8	63.9	63.6	63.1
% PNER	61.6	67.8	57.0	56.8	56.7
% WER	52.7	52.8	52.0	51.9	51.8

6.6 Performance of NBOW2 on Text Classification Tasks

In this section we will evaluate the NBOW2 model on standard text classification tasks and compare its performance with that of the NBOW model and other state of the art results. This evaluation is done to showcase the improved performance that our proposed NBOW2 model can achieve and to further support our argument that the NBOW2 model can learn task specific word importance weights. We will first present a brief description of the two text classification tasks that we will use for evaluation of the NBOW2 model. This will be followed by a discussion on the word importance weights learned by the NBOW2 model and a comparison of its classification performance. It must be noted that as oppose to OOV proper name retrieval, these are single label classification tasks and their performance will be evaluated in terms of percentage accuracy of classification of the input text document into the target class.

6.6.1 Task Descriptions

To analyse the working and performance of our proposed NBOW2 model, we consider two common tasks: (a) binary sentiment classification on the Internet Movie Database (IMDB) movie review dataset [Maas et al., 2011] and the Rotten Tomatoes (RT) movie review dataset [Pang and Lee, 2005], and (b) topic classification on the 20 Newsgroup dataset. The IMDB dataset has longer movie reviews, in form of paragraphs, as compared to those in RT, which are just sentences. Each of the movie review is to be tagged as positive or negative. The 20 Newsgroup dataset has newsgroup documents, organised into 20 categories (for example misc.forsale, soc.religion.christian, etc.). These are among the standard tasks for evaluating text classification algorithms and we have made available our source code used in these experiments[6] so that the results can be reproduced.

6.6.1.1 Sentiment Analysis

For the IMDB task we use the original dataset[7] with 25000 train and 25000 test movie reviews. For Rotten Tomatoes (RT) we obtained the v1.0 dataset[8]. Following the standard evaluation scheme, we do 10-fold cross-validation over the balanced binary dataset of 10,662 sentences of RT. In both the IMDB and RT tasks, model training parameters[9] for NBOW2 are kept similar to those chosen

[6]Source code available at https://github.com/mranahmd/nbow2-text-class

[7]http://ai.stanford.edu/~amaas/data/sentiment/

[8]https://www.cs.cornell.edu/people/pabo/movie-review-data/

[9]word vector size 300, word dropout probability 0.3, L2 regularisation weight 1e-5

for NBOW by Iyyer [Iyyer et al., 2015] after cross validation. For NBOW and NBOW2 models, the '-RAND' suffix will denote random word vector initialisation and no suffix is initialisation with publicly available 300 dimensional Global Vectors (GloVe 300-d) [Pennington et al., 2014] trained on the Common Crawl[10].

6.6.1.2 The 20 Newsgroup Topic Classification

For the 20 Newsgroup topic classification task we use the 'bydate' train/test splits, cleaned and made available by Cardoso [Cardoso-Cachopo, 2007][11]. There are 11,293 text documents in the original training set and 7,528 in the test set. For training the NBOW and NBOW2 models, we randomly extract 15% of the original train set as the validation set and use the remaining 85% as the final training set. This is the most common approach for performing evaluation on this task. Training was performed with the ADADELTA [Zeiler, 2012] gradient descent algorithm. An L2 regularisation weight of 1e-5 was applied to all parameters. Further, to add robustness, we applied 75% word dropout[12]. Additionally we use an early stopping criterion to stop the training when the validation error starts to increase continuously for 5 training epochs. Similar to the sentiment analysis experiments '-RAND' suffix will denote random word vector initialisation and no suffix is initialisation with 300-d GloVe.

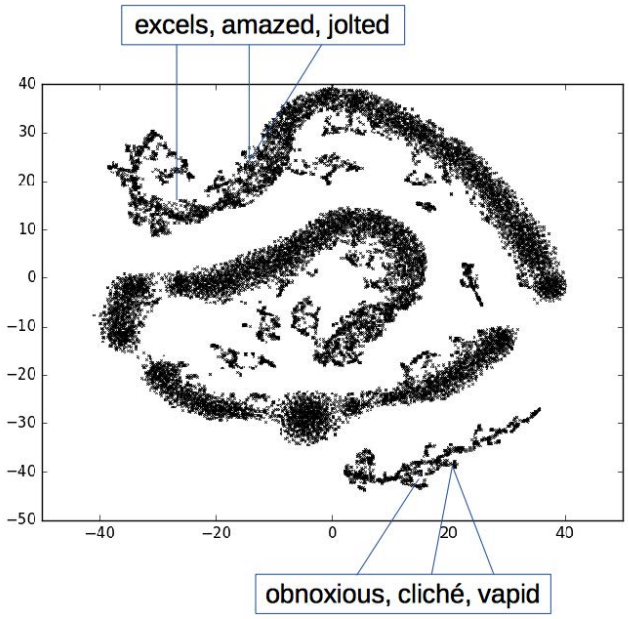
6.6.2 Word importance weights learned by the NBOW2 model

Before discussing the classification performance, we present an analysis of the word importance weights learned by the NBOW2 model by demonstrating some qualitative and quantitative results.

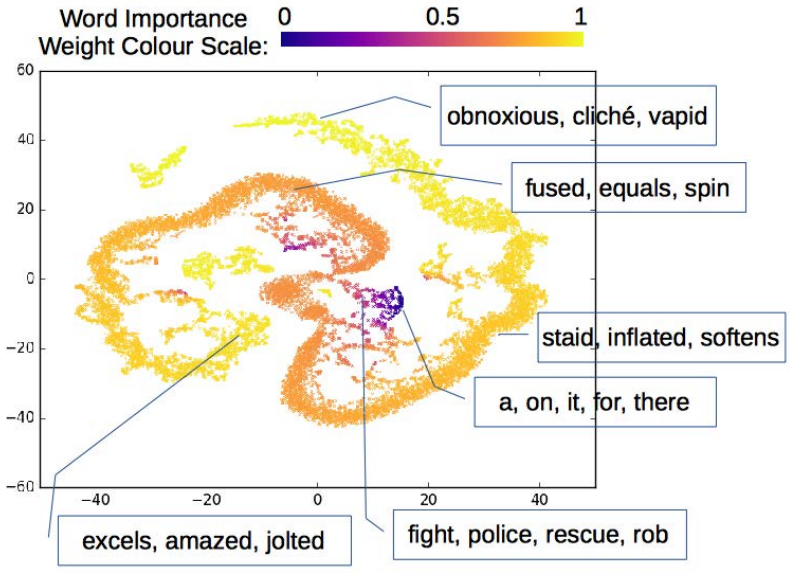
6.6.2.1 Visualisation of word vectors from the RT sentiment analysis task

We visually examine the word vectors learned by the NBOW and NBOW2 models. For visualisation, these word vectors are projected into a two dimensional space using the *t-Distributed Stochastic Neighbour Embedding* (t-SNE) technique [van der Maaten and Hinton, 2008]. Figure 6.6 shows the two dimensional t-SNE plot of word vectors learned by these models, with Figure 6.6 (a) representing those from the NBOW model and Figure 6.6 (b) representing those from

[10]http://nlp.stanford.edu/projects/glove/
[11]http://web.ist.utl.pt/acardoso/datasets/
[12]choice based on accuracy on validation set



(a)



(b)

Figure 6.6: Visualisation of word vectors learned by the NBOW and NBOW2 models in the RT task. Word vectors are reduced to 2 dimension using t-SNE technique and shown in each plot. Plot (a) represents word vectors from the NBOW model, (b) represents words from NBOW2 model, with colours indicating the word importance weights learned by the NBOW2 model.

the NBOW2 model. In Figure 6.6 (b) each word projection is given a colour based on the word importance assigned to it by the NBOW2 model.

In Figure 6.6 (a) we see that the NBOW model can separate the words in the word vector space. According to the word examples labelled in Figure 6.6 (a) the word projections are grouped into two regions/clusters corresponding to positive and negative sentiments of the RT movie review task. Similarly we can observe that the NBOW2 model also learns to separate the words into regions of positive and negative sentiments, as shown by the same word examples in Figure 6.6 (b). If we examine the word importance assigned by the NBOW2 model, indicated by colours in Figure 6.6 (b), it is evident that the NBOW2 model also learns to separate words based on their importance weights. To support this statement we show additional word examples labelled in different regions in Figure 6.6 (b). For instance the words *a, on, it, for, there* are not so important for the RT sentiment classification task[13] and are present together in a region of low word importance. On the contrary, the words *staid, inflated, softens* can contribute to negative polarity of movie reviews. Note that they have relatively higher importance weights and are present towards the negative sentiment region in the word vector space. Compared to both these set of examples, the words *obnoxious, cliché, vapid* are strongly negative. They have importance weights close to one and are present together at the tip of the negative sentiment region (completely opposite to the positive sentiment region with words like *excels, amazed, jolted*).

To further verify our claim that, in comparison to the NBOW model, the NBOW2 model is able to distinguish words based on their importance we present Figure 6.7. This figure shows the word vectors learned by the NBOW model (same as in Figure 6.6 (a)) but it depicts each word with a colour based on word importance weight learned by the NBOW2 model. It can be seen that the NBOW model does not separate/group words based on word importance, even if we look only to the example words *a, on, it, for, there*.

6.6.2.2 Word importance weights v/s TF-IDF weights as classification features

In this analysis, we compare the word importance weights learned by the NBOW2 model with Term Frequency-Inverse Document Frequency (TF-IDF) weights and other word weight features proposed in previous works. For this comparison, one Support Vector Machine (SVM) classifier is trained for the IMDB binary classification task and one for the RT binary classification task. The input to the SVM classifier is a train/test document represented as a sparse BOW feature

[13]from a BOW sentiment classification perspective; for other approaches or text analysis they might be essential

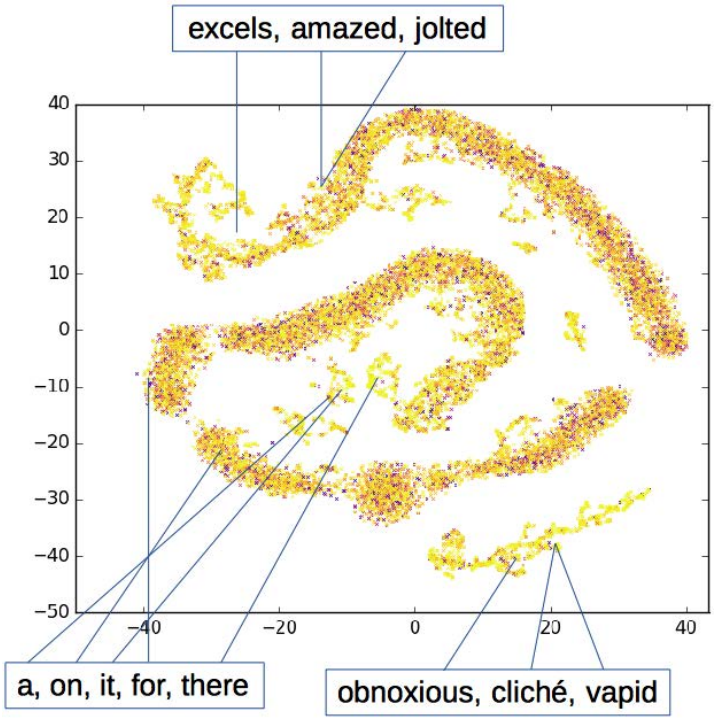


Figure 6.7: Visualisation of word vectors from the NBOW model, as in (a), but each word is depicted with word importance weight learned by the NBOW2 model.

vector in which each word feature is only the word weight. For NBOW2 model it is the scalar word importance weight learned by the model. For comparison we train separate SVM classifiers trained using the following word weight features:

- classical TF-IDF weights
- credibility adjusted TF-IDF (cred-TF-IDF) weights proposed by Kim [Kim and Zhang, 2014]
- binary cosine-normalised weights (bnc), binary delta-smoothed-idf cosine-normalised (bΔ'c) weights used by Maas [Maas et al., 2011]
- the Naive-Bayes SVM (NBSVM) method proposed by Wang [Wang and Manning, 2012]

TF-IDF, bnc and bΔ'c word weights are task independent word weights but cred-TF-IDF and NBSVM are built using class/task information. It must be noted that some of these features have given state-of-the-art results for IMDB and RT tasks.

Features for SVM Classifier	IMDB	RT
bnc [Maas et al., 2011]	87.8	-
bΔ'c [Maas et al., 2011]	88.2	-
TF-IDF-uni [Kim and Zhang, 2014]	88.6	77.1
cred-TF-IDF-uni [Kim and Zhang, 2014]	88.8	77.5
NBSVM-uni [Wang and Manning, 2012]	88.3	78.1
NBOW2-RAND Word Importance Weights	88.2	76.7
NBOW2 Word Importance Weights	88.3	76.3

Table 6.7: Quantitative evaluation of different word weight features, in terms of classification accuracy obtained using an SVM classifier. (For IMDB 0.1% corresponds to 25 test documents. For RT 1% is about 10 test sentences.)

The classification accuracies obtained by the SVM classifiers are reported in Table 6.7. The TF-IDF, cred-TF-IDF and NBSVM methods are denoted with a '-uni' suffix in Table 6.7 following the notation used by Kim [Kim and Zhang, 2014]. For the SVM classifier on 25k full length test documents of the IMDB task, the NBOW2 model weights are as good as NBSVM and bΔ'c and better than bnc. But they do not perform as good as the TF-IDF weights. Whereas for the RT task with 1066 test sentences, the NBOW2 model word weights perform closer to the TF-IDF variants.

6.6.3 NBOW2 Classification Performance

After the discussion on the word importance weights learnt by the NBOW2 model we compare the classification results obtained with our NBOW2 model. We compare the NBOW2 model classification accuracy to that obtained from the NBOW model [Iyyer et al., 2015], BOW approaches based on Restricted Boltzmann Machines (RBM) and Support Vector Machines (SVM) and more complex approaches based on RNN, CNN. It must be noted that the CNN and RNN based approaches operate on rich word sequence information and have been shown to perform better than BOW approaches on these tasks.

Table 6.8 compares the classification accuracy of the NBOW2 model on IMDB and Rotten Tomatoes (RT) movie reviews binary classification tasks. Table 6.9 compares the classification accuracy on 20 Newsgroup topic classification. Results in Table 6.8 and Table 6.9 indicate that the NBOW2 model gives the best accuracy among the BOW approaches. For the IMDB and the newsgroup task, the accuracy of the NBOW2 model is closer to that of NBOW (not statistically significant for the 20 Newsgroup). It is also evident that for RT and newsgroup classification, the performance of NBOW2 is not far from the CNN and LSTM

Model	IMDB	RT
NBOW-RAND [Iyyer et al., 2015]	88.9	76.2
NBOW [Iyyer et al., 2015]	89.0	79.0
NBOW2-RAND	88.7	78.2
NBOW2	**89.1**	**80.5**
NBSVM-uni [Wang and Manning, 2012]	88.3	78.1
NBSVM-bi [Wang and Manning, 2012]	91.2	79.4
CNN-MC [Kim, 2014]	-	81.1
CNN-non-static [Kim, 2014]	-	**81.5**
s2-bow*n*-CNN [Johnson and Zhang, 2015]	**92.3**	-
SA-LSTM [Dai and Le, 2015]	**92.8**	**83.3**
LM-LSTM [Dai and Le, 2015]	92.4	78.3

Table 6.8: IMDB and Rotten Tomatoes (RT) movie reviews sentiment classification accuracy. The first group lists BOW methods; including different initialisations of NBOW and NBOW2 (this work). The next group shows the best reported results with bi-gram BOW and CNN methods, followed by LSTM RNN. The best method in each group is shown in bold. (For IMDB 0.1% corresponds to 25 test documents. For RT 1% is about 10 test sentences.)

Model	Accuracy (%)
NBOW-RAND	83.2
NBOW	83.2
NBOW2-RAND	82.7
NBOW2	**83.4**
RBM-MLP [Dauphin and Bengio, 2013]	79.5
SVM + BoW [Cardoso-Cachopo, 2007]	82.8
SA-LSTM [Dai and Le, 2015]	84.4
LM-LSTM [Dai and Le, 2015]	**84.7**

Table 6.9: 20 Newsgroup topic classification accuracy. First group lists BOW methods; including different initialisations of NBOW [Iyyer et al., 2015] and NBOW2 (this work). The second group shows best reported results with LSTM RNN. Best method in each group is shown in bold. (0.2% corresponds to about 15 test set documents.)

methods. For further analysis we also trained the NBOW2 model by simply using fixed TF-IDF weights in Equation 6.5. This gave 87.6% and 79.4% accuracy for the the IMDB and the RT task. Thus we can state that the word importance weights of the NBOW2 model are themselves informative.

6.7 Summary of Contributions and Conclusion

In previous chapters, our evaluations of context representations learned with LSA, LDA, CBOW and Skip-gram models showed that LSA and Skip-gram representations outperform those from LDA and CBOW but LDA representations are more robust to LVCSR errors. Arguing that these representations, learned in an unsupervised manner, are not the most optimal for our task of OOV proper name retrieval, in this chapter we proposed discriminative context representations trained with an objective to maximise the OOV proper name retrieval performance.

Our first proposed model is a bag-of-word neural network model which learns a document context vector by averaging the vectors of words in the document and retrieves the relevant OOV proper names using a logistic regression. We named it the Neural Bag-Of-Word (NBOW) model following its resemblance to the model proposed in [Iyyer et al., 2015].

As an improvement to the NBOW model, we proposed the Neural Bag-Of-Weighted-Word (NBOW2) model, which learns task specific word importance weights and performs a weighted sum composition of the word vectors to obtain the document context.

While the proposed models share architectural similarity with the CBOW model our model training mechanism is completely different. We used a gradient descent learning algorithm with mini-batches of training samples and used ADADELTA [Zeiler, 2012] which provides an adaptive per-dimension learning rate for gradient descent. More importantly we proposed the use of word dropout to simulate document context variations and LVCSR deletion errors.

From the evaluation of the proposed NBOW and NBOW2 models on our OOV proper name retrieval task we can conclude that:

- The proposed NBOW and NBOW2 models outperform the methods based on LSA, LDA, CBOW and Skip-gram in terms of MAP of retrieval of OOV proper names. As compared to the best performing method (Skip-gram Method I, SG-I), they achieve MAP improvements of 25.8%, 16% and 11%

relative on ANTS LVCSR transcriptions, KATS LVCSR transcriptions and reference transcriptions respectively (see Figure 6.3).

- As opposed to the LSA, CBOW and Skip-gram models, which show substantial degradation on LVCSR transcriptions compared to working with the reference transcriptions, the proposed NBOW and NBOW2 models are more robust to LVCSR word errors.

- While training the NBOW and NBOW2 models takes a large number of training epochs, these can be significantly reduced with the proposed NBOW2+ model which concatenates the average context vector of NBOW and the weight sum context vector of NBOW2 into one model. For instance, for the (equally) good performing NBOW and NBOW2 models (of similar configurations) the number of training epochs was reduced from 405 and 648 to 273 with the NBOW2+ model (see Figure 6.4). This improvement in training convergence does not affect the MAP performance. While the number of epochs are quite large compared to those taken by CBOW and Skip-gram models, it must be noted that the amount of computation per epoch required by the NBOW, NBOW2 and NBOW2+ models is quite small as also evident from their architectures and the number of parameters.

The relevant OOV proper names retrieved by the LDA and NBOW models were further evaluated by including them as unigrams in the language model of the second pass speech recognition. These second pass speech recognition experiments showed a 7.8% relative drop in proper name error rate. On the other hand, if all possible OOV proper names were simply added to the LVCSR vocabulary, the proper name error rate increases by 10% relative. Further improvements are possible by using well designed language model adaptation schemes and by using diachronic text data from more sources.

As the NBOW and NBOW2 models gave very similar performance on our task of retrieval of OOV proper names, we evaluated these models on standard text classification tasks. Our experiments on standard topic and sentiment classification tasks showed that proposed NBOW2 model (a) learns meaningful word importance weights, and (b) gives the best accuracies among the bag-of-word approaches. The word importance weights learned by the NBOW2 model are comparable to TF-IDF based word weights when used as features in a bag-of-word SVM classifier.

CHAPTER 7

Conclusion

Diachronic audio/video documents exhibit frequent variations of topics over time, giving rise to many new proper names missing from the vocabulary and language model of a Large Vocabulary Continuous Speech Recognition (LVCSR) system. This leads to the Out-Of-Vocabulary (OOV) problem, where new proper names cannot be recognised by the LVCSR system. To address this problem, we propose to model the semantic and topic context of the OOV proper names. In our approach, a diachronic text corpus containing new proper names was collected from the internet and models were trained to learn the semantic/topic context of the new proper names. Given an audio document to recognize, our models infer the semantic/topic context of the spoken content and retrieve a list of context relevant OOV proper names. As a result, the target OOV proper names in the audio document can now be found within a small subset, about 1% of all possible OOV proper names. We focused on modelling the context of the OOV proper names and a robust retrieval of the ones relevant to the spoken content; the actual recovery and LVCSR update being beyond the scope of this dissertation.

To exploit semantic and topic context representations to retrieve OOV proper names, we presented two different methodologies. One measures the closeness between the context of the LVCSR hypothesis and a global contextual representation of each OOV proper name. Whereas the other relies on similarities with document specific representations of OOV proper names. Chapter 4 evaluated these methodologies on representations from the Latent Semantic Analysis (LSA) model and Latent Dirichlet Allocation (LDA) based topic models. Chapter 5 extended these methods to word embedding spaces from the Skip-gram and Continuous Bag-Of-Words (CBOW) models. A thorough analysis of the different models and methods enable us to conclude the following.

- Semantic and topic context representations outperform simple word co-occurrence statistics based on Pointwise Mutual Information (PMI). 15-25% improvements were seen in the MAP of retrieval of OOV proper names on our experiment corpus.

- The proposed document specific OOV proper name representations give significant improvements for the different modelling approaches. The amount of computation in this methodology increases with the number of documents in the diachronic text corpus. As compared to this, the computation for the methodology using global representation of OOV proper names is dependant only on the number of OOV proper names.

- While prior works show that LDA performs better than LSA on word prediction task, we found that representations from LSA give better MAP performance in our task. However LDA topic space representations are more robust to LVCSR word errors.

- The existing Entity Topic models did not perform better than LDA. But our proposed CorrLDA1-F model gives better MAP than LDA and other entity topic models, when using global OOV proper name representations. The document and word topic distributions in this model are centred around OOV proper names and this enables it to perform better.

- Representations from word embedding spaces gave the best MAP performance among all these (unsupervised) representations.

- Hyper-parameter exploration is very crucial to achieve a good OOV retrieval performance with representations from the LSA, LDA and word embedding models.

Our analysis on selection of diachronic text corpora, in Chapter 4, shows that for retrieval of relevant OOV proper names,

- the coverage of target OOV proper names and the total retrieval recall can be increased by augmenting data from multiple sources.

- it is better to rely on diachronic text corpora from multiple sources than on a single corpus from longer time span.

Chapter 6 presented models to learn discriminative context representations which significantly outperform the previous models and methods. After the LDA model, the proposed NBOW2+ model was the most robust to LVCSR errors. Table 7.1 shows the improvements in performance of retrieval of relevant OOV proper names obtained with the best retrieval methods for the different models.

To validate the achievement of our aim '*to retrieve a list of relevant OOV proper names*', we performed an evaluation with a second pass speech recognition in which the top-128 retrieved OOV proper names were added to the LVCSR vocabulary. The OOV proper name recognition results, in Chapter 6, showed

- a 10% relative increase in the proper name error rate when all known OOV proper names were simply added to the LVCSR vocabulary, and
- a 7.8% relative drop in the proper name error rate when the relevant OOV proper names, retrieved by the NBOW2 model, are added simply as unigrams into the LVCSR language model.

The diachronic text corpus used in these experiments had a target OOV proper name coverage of only about 40% and the gains would be higher with a larger diachronic text corpus with more new proper names.

Table 7.1: Performance of retrieval of relevant OOV proper names obtained with the best retrieval methods and the best model configuration for the different models.

	PMI	LDA	RP	LSA	CBOW	Skip-Gram	NBOW2+
MAP@128 for KATS LVCSR transcriptions	0.247	0.399	0.462	0.485	0.508	0.506	**0.588**

Evaluation of our proposed Neural Bag-Of-Weighted-Words (NBOW2) model on standard sentiment and topic classification tasks, in Chapter 6, showed that it gives the best performance among the bag-of-word approaches.

7.1 Contributions

The main contributions of this dissertation are presented below.

- New methodologies to exploit semantic and topic context to retrieve OOV proper names relevant to an audio/video document.
  - One methodology is to measure the closeness between the context of the LVCSR hypothesis and a global contextual representation of each OOV proper name.
  - The other methodology derives document specific representations of OOV proper names, and measures the closeness between the context of the LVCSR hypothesis and different instance specific representations of each OOV proper name [**Sheikh et al., 2016b**].

- A systematic analysis and exploration of different semantic and topic models, to model the context of OOV proper names.
  - We introduced our methodologies with the Latent Dirichlet Allocation (LDA) topic model [**Sheikh et al., 2015b**].
  - Being a Bayesian probabilistic model, new variables can be introduced into the LDA model. We explored this extensibility with Entity Topic models [**Sheikh et al., 2015a**] and proposed a new model variant, CorrLDA1-Flipped, which performs better than LDA and other entity topic models, when using global OOV proper name representations.
  - We evaluated the proposed methodologies on semantic representations from Latent Semantic Analysis (LSA) and extended these to word embedding spaces from Skip-gram & CBOW models.
- Models for learning discriminative context representations for noisy and mismatched text input [**Sheikh et al., 2016c**].
  - We presented methods to train bag-of-word neural network architectures efficiently, leading to improved performance and robustness to errors from automatic speech recognition.
- Neural Bag-Of-Weighted-Words (NBOW2) model [**Sheikh et al., 2016d**].
  - Our NBOW2 model learns to assign task specific importance weights to words/features in documents, using a simple BOW architecture.
  - The proposed weighted average composition of documents can improve learning and provide meaningful insights on the task corpora.
- A thorough evaluation and comparison of different semantic and topic space representations for a (generalisable) retrieval task.
  - While we focused on the task of retrieval of relevant OOV proper names, our problem setup readily extends and generalises to other document entities or meta information for instance non proper name OOVs, tags for audio/video documents and their genre or categories. As compared to related works in topic modelling area, the study in this dissertation would provide important pointers for (a) selection of appropriate models for context representations, (b) setups where output/retrieval labels are large in number and have a skewed distribution of number of training samples, and (c) dealing with erroneous speech recognition hypothesis in similar context representation tasks.

Source code of proposed models is shared at https://github.com/mranahmd/

7.2 Future Directions & Prospects

This dissertation has made important advances in modelling semantic and topical context to improve recognition of OOV proper names in large vocabulary speech recognition systems. We envision possible directions to enhance our proposed solutions, as well to improve automatic speech recognition and audio/video indexing systems in general.

The short term future works would be:

- **Automatic Indexing of audio/video archives**
  Throughout the dissertation we focussed on retrieval of relevant OOV proper names, however the proposed methodology would enable automatic indexing and structuring of large audio/video archives. Our proposed retrieval methods can reduce a longer list of possible indexes and suggest context relevant indexes. Proposed NBOW2 model can also be used for automatically suggesting important keywords and indexes for the audio/video documents.

- **Updating the LVCSR to recognise the retrieved Proper Names**
  It is necessary to develop methods to update the LVCSR language model with the retrieved list. This is essential for non indexing applications, where reliable transcriptions are required. Some recent works focusing on updating the LVCSR WFST [Allauzen and Riley, 2015, Ma et al., 2015a], in order to introduce new or relevant words, would be a direction.

- **Extensions to the Neural Bag-Of-Weighted-Words (NBOW2)**
  The proposed NBOW2 model learns task specific word importance weights and this can be used in keyword extraction and text summarization tasks. We foresee further extensions to learn word importance, such as (a) including additional parameters to learn class-specific word importance, and (b) learning document context specific word importance (as in [Sheikh et al., 2015c]).

- **Beyond document level context**
  In our approach, we chose to rely on the document level semantic/topic context. Lower levels of contextual information, including local word context and also phonetic information of proper names, can be exploited. The main challenge here is to model errors in word sequences. We performed preliminary experiments extending our bag-of-word neural network architectures to incorporate this information. The results are motivating and indicate that OOV proper name recovery can be improved significantly, and this approach can possibly eliminate the need for a second pass speech recognition or keyword search.

The long term future directions are:

- **From single event to multiple events per document**
  Throughout this dissertation we focused on broadcast news audio with a single news event and used a document level context derived with a bag-of-words representation. However there are news programs in which multiple news events are discussed one after the other. In such documents a single global bag-of-word representation may not work. Topic segmentation techniques [Tür et al., 2001, Mohri et al., 2010, Bouchekif et al., 2015] can be applied as a pre-processing step in such scenarios.

  A more interesting solution would be to design a sequential variant of our NBOW2 model using Recurrent or Convolutional Neural Network architectures. While this might converge to the popular attention based neural network models [Bahdanau et al., 2014], one of the challenges would be to model the many-to-few sequence learning problem, similar to Connectionist Temporal Classification [Graves et al., 2006].

- **End-to-End and Open Vocabulary recognition systems**
  A recent trend in automatic speech recognition is an End-to-End neural network pipeline [Graves and Jaitly, 2014, Hannun et al., 2014, Song and Cai, 2015, Miao et al., 2015, Bahdanau et al., 2016]. These systems mainly transcribe speech into a sequence of characters and use a language model or a WFST frameworks to obtain the word sequences. Similar to hybrid language models, even these systems would face the problem of unresolved OOV words and proper names. Our proposed approach to model semantic and topic context, would be useful in such systems. When incorporated in the later stages of these systems it would help to resolve the target OOVs and improve the word level transcriptions.

- **Other contextual information**
  Apart from the semantic and topic context in the spoken content, there are many other possible contextual cues which can be modelled, depending on the kind of audio/video documents being addressed. Simple contextual cues like document title, tags, timestamps, authors, etc. can be useful for online adaptation of LVCSR processing audio/video files. Some existing works have used these cues, but mostly in a ad-hoc manner. For large archives, like Youtube and Dailymotion, it could be worth investigating structured probabilistic and neural network models, as presented in this dissertation.

  Even more interesting contextual cues are the perceptual ones, like acoustic themes/genre [Kim et al., 2012, Doulaty et al., 2016] and visual cues from image/video features. Such contextual cues will be more challenging to

model, but they are more rich and they would bring new abilities to machine cognition; for example visual attention to improve multi-modal machine translation [Specia et al., 2016], grounding language models based on visual cues [Fleischman and Roy, 2008].

Bibliography

[Abdel-Hamid et al., 2014] Abdel-Hamid, O., Mohamed, A.-R., Jiang, H., Deng, L., Penn, G., and Yu, D. (2014). Convolutional neural networks for speech recognition. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 22(10):1533–1545.

[Allauzen and Gauvain, 2005a] Allauzen, A. and Gauvain, J.-L. (2005a). Diachronic vocabulary adaptation for broadcast news transcription. In *ISCA INTERSPEECH*, pages 1305–1308.

[Allauzen and Gauvain, 2005b] Allauzen, A. and Gauvain, J.-L. (2005b). Open vocabulary ASR for audiovisual document indexation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1013–1016.

[Allauzen and Riley, 2015] Allauzen, C. and Riley, M. (2015). Rapid vocabulary addition to context-dependent decoder graphs. In *16th Annual Conference of the International Speech Communication Association - INTERSPEECH*, pages 2112–2116.

[Anusuya and Katti, 2011] Anusuya, M. A. and Katti, S. K. (2011). Front end analysis of speech recognition: a review. *International Journal of Speech Technology*, 14(2):99–145.

[Asadi et al., 1991] Asadi, A., Schwartz, R., and Makhoul, J. (1991). Automatic modeling for adding new words to a large-vocabulary continuous speech recognition system. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 305–308 vol.1.

[Atal and Hanauer, 1971] Atal, B. S. and Hanauer, S. L. (1971). Speech analysis and synthesis by linear prediction of the speech wave. *The Journal of the Acoustical Society of America*, 50(2B):637–655.

[Bahdanau et al., 2014] Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

[Bahdanau et al., 2016] Bahdanau, D., Chorowski, J., Serdyuk, D., Brakel, P., and Bengio, Y. (2016). End-to-end attention-based large vocabulary speech

recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4945–4949.

[Bahl et al., 1978] Bahl, L., Baker, J., Cohen, P., Cole, A., Jelinek, F., Lewis, B., and Mercer, R. (1978). Automatic recognition of continuously spoken sentences from a finite state grammer. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP).*, volume 3, pages 418–421.

[Bahl et al., 1983] Bahl, L. R., Jelinek, F., and Mercer, R. L. (1983). A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-5(2):179–190.

[Baroni et al., 2014] Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247.

[Baum et al., 1970] Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *Ann. Math. Statist.*, 41(1):164–171.

[Bayer and Riccardi, 2012] Bayer, A. O. and Riccardi, G. (2012). Joint language models for automatic speech recognition and understanding. In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 199–203.

[Bayer and Riccardi, 2014] Bayer, A. O. and Riccardi, G. (2014). Semantic language models for automatic speech recognition. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 7–12.

[Bazzi and Glass, 2000] Bazzi, I. and Glass, J. R. (2000). Modeling out-of-vocabulary words for robust speech recognition. In *International Conference on Spoken Language Processing (ICSLP)*, pages 401–404.

[Bazzi and Glass, 2002] Bazzi, I. and Glass, J. R. (2002). A multi-class approach for modelling out-of-vocabulary words. In *ISCA INTERSPEECH*.

[Béchet et al., 2000] Béchet, F., Nasr, A., and Genet, F. (2000). Tagging unknown proper names using decision trees. In *38th Annual Meeting on Association for Computational Linguistics*, pages 77–84, PA, USA.

[Bellegarda, 1999] Bellegarda, J. R. (1999). Speech recognition experiments using multi-span statistical language models. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 2, pages 717–720.

[Bellegarda, 2004a] Bellegarda, J. R. (2004a). *Mathematical Foundations of Speech and Language Processing*, chapter Latent Semantic Language Modelling for Speech Recognition, pages 73–103. Springer New York, New York, NY.

[Bellegarda, 2004b] Bellegarda, J. R. (2004b). Statistical language model adaptation: review and perspectives. *Speech Communication*, 42(1):93 – 108. Adaptation Methods for Speech Recognition.

[Bengio, 2012] Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. *CoRR*, abs/1206.5533.

[Bengio et al., 2003] Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155.

[Bengio et al., 2001] Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C., K, J., Hofmann, T., Poggio, T., and Shawe-taylor, J. (2001). A neural probabilistic language model. In *Advances in Neural Information Processing Systems*.

[Bertoldi and Federico, 2001] Bertoldi, N. and Federico, M. (2001). Lexicon adaptation for broadcast news transcription. In *ISCA ITRW workshop on Adaptation Methods for Speech Recognition*, pages 187–190.

[Bigot et al., 2013] Bigot, B., Senay, G., Linarès, G., Fredouille, C., and Dufour, R. (2013). Person name recognition in ASR outputs using continuous context models. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 8470–8474.

[Bingham and Mannila, 2001] Bingham, E. and Mannila, H. (2001). Random projection in dimensionality reduction: Applications to image and text data. In *Proceedings of the Seventh ACM SIGKDD*, pages 245–250, New York, NY, USA.

[Bisani and Ney, 2005] Bisani, M. and Ney, H. (2005). Open vocabulary speech recognition with flat hybrid models. In *ISCA INTERSPEECH*, pages 725–728.

[Bisani and Ney, 2008] Bisani, M. and Ney, H. (2008). Joint-sequence models for grapheme-to-phoneme conversion. *Speech Communication*, 50(5):434 – 451.

[Blei, 2012] Blei, D. M. (2012). Probabilistic topic models. *Commun. ACM*, 55(4):77–84.

[Blei et al., 2001] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2001). Latent dirichlet allocation. In *Advances in Neural Information Processing Systems*, pages 601–608.

[Blei et al., 2003] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

[Bouchekif et al., 2015] Bouchekif, A., Damnati, G., Estève, Y., Charlet, D., and Camelin, N. (2015). Diachronic semantic cohesion for topic segmentation of tv broadcast news. In *16th Annual Conference of the International Speech Communication Association - INTERSPEECH*, pages 2932–2936.

[Bouma, 2009] Bouma, G. (2009). Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL*, pages 31–40.

[Bourlard and Morgan, 1993] Bourlard, H. A. and Morgan, N. (1993). *Connectionist Speech Recognition: A Hybrid Approach*. Kluwer Academic Publishers, Norwell, MA, USA.

[Bullinaria and Levy, 2007] Bullinaria, J. A. and Levy, J. P. (2007). Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39(3):510–526.

[Cardoso-Cachopo, 2007] Cardoso-Cachopo, A. (2007). Improving Methods for Single-label Text Categorization. PdD Thesis, Instituto Superior Tecnico, Universidade Tecnica de Lisboa.

[Carpenter, 2010] Carpenter, B. (2010). Integrating out multinomial parameters in latent dirichlet allocation and naive bayes for collapsed gibbs sampling. Available at http://lingpipe.files.wordpress.com/2010/07/lda3.pdf.

[Celikyilmaz et al., 2010] Celikyilmaz, A., Hakkani-Tur, D., and Tur, G. (2010). Lda based similarity modeling for question answering. In *Proceedings of the NAACL HLT 2010 Workshop on Semantic Search*, pages 1–9.

[Cha and Cho, 2012] Cha, Y. and Cho, J. (2012). Social-network analysis using topic models. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 565–574.

[Chan et al., 2015] Chan, W., Jaitly, N., Le, Q. V., and Vinyals, O. (2015). Listen, attend and spell. *CoRR*, abs/1508.01211.

[Chen et al., 2013a] Chen, G., Yilmaz, O., Trmal, J., Povey, D., and Khudanpur, S. (2013a). Using proxies for oov keywords in the keyword search task. In

IEEE Workshop on Automatic Speech Recognition Understanding (ASRU), pages 416–421.

[Chen, 2009] Chen, S. F. (2009). Shrinking exponential language models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 468–476.

[Chen et al., 2013b] Chen, W., Ananthakrishnan, S., Prasad, R., and Natarajan, P. (2013b). Variable-span out-of-vocabulary named entity detection. In *ISCA INTERSPEECH*, pages 3761–3765.

[Chien and Chueh, 2008] Chien, J.-T. and Chueh, C.-H. (2008). Latent dirichlet language model for speech recognition. In *IEEE Spoken Language Technology Workshop (SLT)*, pages 201–204.

[Choueiter, 2009] Choueiter, G. F. (2009). *Linguistically-motivated Sub-word Modeling with Applications to Speech Recognition*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA.

[Chueh and Chien, 2009] Chueh, C.-H. and Chien, J.-T. (2009). Nonstationary latent dirichlet allocation for speech recognition. In *INTERSPEECH*, pages 372–375.

[Church and Hanks, 1990] Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Comput. Linguist.*, 16(1):22–29.

[Dahl, 2015] Dahl, G. E. (2015). *Deep learning approaches to problems in speech recognition, computational chemistry, and natural language text processing*. PhD thesis, University of Toronto.

[Dahl et al., 2011] Dahl, G. E., Yu, D., Deng, L., and Acero, A. (2011). Large vocabulary continuous speech recognition with context-dependent DBN-HMMS. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASS)*, pages 4688–4691.

[Dai and Le, 2015] Dai, A. M. and Le, Q. V. (2015). Semi-supervised sequence learning. In *Advances in Neural Information Processing Systems*, pages 3061–3069.

[Damani, 2013] Damani, O. (2013). Improving pointwise mutual information (pmi) by incorporating significant co-occurrence. In *CoNLL*, pages 20–28.

[Dauphin and Bengio, 2013] Dauphin, Y. and Bengio, Y. (2013). Stochastic ratio matching of rbms for sparse high-dimensional inputs. In *Advances in Neural Information Processing Systems 26*, pages 1340–1348.

[David M. Blei, 2007] David M. Blei, J. D. L. (2007). A correlated topic model of science. *The Annals of Applied Statistics*, 1(1):17–35.

[Davis and Mermelstein, 1980] Davis, S. and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357–366.

[Deerwester, 1988] Deerwester, S. (1988). Improving information retrieval with latent semantic indexing. In *Proceedings of the 51st ASIS Annual Meeting (ASIS)*.

[Deerwester et al., 1990] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal Of The American Society For Information Science*, 41(6):391–407.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society Series B (Statistical Methodology)*, 39:1–38.

[Dong et al., 2014] Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., and Xu, K. (2014). Adaptive recursive neural network for target-dependent twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL, Baltimore, MD, USA, Volume 2: Short Papers*, pages 49–54.

[Doulaty et al., 2016] Doulaty, M., Saz, O., Ng, R. W. M., and Hain, T. (2016). Automatic genre and show identification of broadcast media. In *Proceedings of the 17th Annual Conference of the International Speech Communication Association (Interspeech)*, pages 2115–2119.

[Ellis and Morgan, 1999] Ellis, D. and Morgan, N. (1999). Size matters: an empirical study of neural network training for large vocabulary continuous speech recognition. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 2, pages 1013–1016 vol.2.

[Federico and Bertoldi, 2001] Federico, M. and Bertoldi, N. (2001). Broadcast news lm adaptation using contemporary texts. In *7th European Conference on Speech Communication and Technology*, pages 239–242.

[Fei-Fei and Perona, 2005] Fei-Fei, L. and Perona, P. (2005). A bayesian hierarchical model for learning natural scene categories. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 524–531 vol. 2.

[Fetter, 1998] Fetter, P. (1998). *Detection and transcription of OOV words*. PhD thesis, DFKI.

[Fleischman and Roy, 2008] Fleischman, M. and Roy, D. (2008). Grounded Language Modeling for Automatic Speech Recognition of Sports Video. In *Proceedings of ACL-08: HLT*, pages 121–129.

[Fohr and Illina, 2015] Fohr, D. and Illina, I. (2015). Continuous word representation using neural networks for proper name retrieval from diachronic documents. In *16th Annual Conference of the International Speech Communication Association - INTERSPEECH*, pages 1344–1348.

[Gales and Young, 2007] Gales, M. and Young, S. (2007). The application of hidden markov models in speech recognition. *Found. Trends Signal Process.*, 1(3):195–304.

[Gerosa and Federico, 2009] Gerosa, M. and Federico, M. (2009). Coping with out-of-vocabulary words: Open versus huge vocabulary asr. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4313–4316.

[Girolami and Kabán, 2003] Girolami, M. and Kabán, A. (2003). On an equivalence between plsi and lda. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 433–434.

[Goldberg, 2015] Goldberg, Y. (2015). A primer on neural network models for natural language processing. *CoRR*, abs/1510.00726.

[Goldberg and Levy, 2014] Goldberg, Y. and Levy, O. (2014). word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *CoRR*, abs/1402.3722.

[Goodman, 2001] Goodman, J. T. (2001). A bit of progress in language modeling. *Computer Speech & Language*, 15(4):403 – 434.

[Graves et al., 2006] Graves, A., Fernández, S., Gomez, F., and Schmidhuber, J. (2006). Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International*

Conference on Machine Learning, ICML '06, pages 369–376, New York, NY, USA. ACM.

[Graves and Jaitly, 2014] Graves, A. and Jaitly, N. (2014). Towards end-to-end speech recognition with recurrent neural networks. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1764–1772.

[Griffiths and Steyvers, 2004] Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101:5228–5235.

[Griffiths et al., 2007] Griffiths, T. L., Tenenbaum, J. B., and Steyvers, M. (2007). Topics in semantic representation. *Psychological Review*, 114:2007.

[Hannemann et al., 2010] Hannemann, M., Kombrink, S., and Martin Karafiát, L. B. (2010). Similarity scoring for recognizing repeated out-of-vocabulary words. In *ISCA INTERSPEECH*, pages 897–900.

[Hannun et al., 2014] Hannun, A. Y., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., and Ng, A. Y. (2014). Deep speech: Scaling up end-to-end speech recognition. *CoRR*, abs/1412.5567.

[Hayamizu et al., 1995] Hayamizu, S., Itou, K., and Tanaka, K. (1995). Detection of unknown words in large vocabulary speech recognition. *Journal of the Acoustical Society of Japan (E)*, 16(3):165–171.

[Heinrich, 2004] Heinrich, G. (2004). Parameter estimation for text analysis. http://www.arbylon.net/publications/text-est.pdf.

[Hemphill et al., 1990] Hemphill, C. T., Godfrey, J. J., and Doddington, G. R. (1990). The atis spoken language systems pilot corpus. In *Proceedings of the Workshop on Speech and Natural Language*, HLT '90, pages 96–101.

[Hermann and Blunsom, 2013] Hermann, K. M. and Blunsom, P. (2013). The Role of Syntax in Vector Space Models of Compositional Semantics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 894–904.

[Hermansky, 1990] Hermansky, H. (1990). Perceptual linear predictive (plp) analysis of speech. *The Journal of the Acoustical Society of America*, 87(4):1738–1752.

[Hetherington, 1995] Hetherington, I. L. (1995). *A Characterization of the Problem of New, Out-of-vocabulary Words in Continuous Speech Recognition and Understanding*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA. AAI0576118.

[Hinton et al., 2012a] Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., and Kingsbury, B. (2012a). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97.

[Hinton et al., 2012b] Hinton, G. E., Deng, L., Yu, D., Dahl, G. E., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., and Kingsbury, B. (2012b). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Process. Mag.*, 29(6):82–97.

[Hinton et al., 2006] Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Comput.*, 18(7):1527–1554.

[Hofmann, 1999] Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Uncertainty in Artificial Intelligence*, pages 289–296.

[Hu et al., 2013] Hu, L., Li, J., Li, Z., Shao, C., and Li, Z. (2013). Incorporating entities in news topic modeling. In *Natural Language Processing and Chinese Computing*, volume 400, pages 139–150. Springer Berlin Heidelberg.

[Illina et al., 2004] Illina, I., Fohr, D., Mella, O., and Cerisara, C. (2004). The Automatic News Transcription System: ANTS some Real Time experiments. In *8th International Conference on Spoken Language Processing (INTERSPEECH'2004 - ICSLP)*, pages 377–380.

[Iyyer et al., 2015] Iyyer, M., Manjunatha, V., Boyd-Graber, J., and Daumé III, H. (2015). Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1681–1691.

[Jelinek, 1997] Jelinek, F. (1997). *Statistical Methods for Speech Recognition*. MIT Press, Cambridge, MA, USA.

[Jelinek et al., 1991] Jelinek, F., Merialdo, B., Roukos, S., and Strauss, M. (1991). A dynamic language model for speech recognition. In *Proceedings of the Workshop on Speech and Natural Language*, HLT '91, pages 293–295, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Johnson and Zhang, 2015] Johnson, R. and Zhang, T. (2015). Effective use of word order for text categorization with convolutional neural networks. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 103–112.

[Jouvet and Langlois, 2013] Jouvet, D. and Langlois, D. (2013). *Proceedings of 16th International Conference on Text, Speech, and Dialogue (TSD)*, chapter A Machine Learning Based Approach for Vocabulary Selection for Speech Transcription, pages 60–67. Springer Berlin Heidelberg.

[Józefowicz et al., 2016] Józefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., and Wu, Y. (2016). Exploring the limits of language modeling. *CoRR*, abs/1602.02410.

[Kalchbrenner et al., 2014] Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 655–665.

[Karakos and Schwartz, 2014] Karakos, D. and Schwartz, R. (2014). Subword and phonetic search for detecting out-of-vocabulary keywords. In *ISCA INTERSPEECH*, pages 2469–2473.

[Karakos and Schwartz, 2015] Karakos, D. and Schwartz, R. M. (2015). Combination of search techniques for improved spotting of oov keywords. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5336–5340.

[Katunobu et al., 1992] Katunobu, I., Satoru, H., and Hozumi, T. (1992). Detection of unknown words and automatic estimation of their transcriptions in continuous speech recognition. In *International Conference on Spoken Language Processing (ICSLP)*, pages 799–802.

[Katz, 1987] Katz, S. (1987). Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(3):400–401.

[Ketabdar et al., 2007] Ketabdar, H., Hannemann, M., and Hermansky, H. (2007). Detection of out-of-vocabulary words in posterior based asr. In *ISCA INTERSPEECH*, pages 1757–1760.

[Kim et al., 2012] Kim, S., Georgiou, P., and Narayanan, S. (2012). Latent acoustic topic models for unstructured audio classification. *APSIPA Transactions on Signal and Information Processing*, volume 1:1–15.

[Kim et al., 2009] Kim, S., Sundaram, S., Georgiou, P. G., and Narayanan, S. (2009). Audio Scene Understanding using Topic Models. In *Proceedings of the Neural Information Processing Systems (NIPS) Workshop on Applications for Topic Models: Text and Beyond*.

[Kim, 2014] Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

[Kim and Zhang, 2014] Kim, Y. and Zhang, O. (2014). Credibility adjusted term frequency: A supervised term weighting scheme for sentiment analysis and text classification. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 79–83, Baltimore, Maryland.

[Klakow et al., 1999] Klakow, D., Rose, G., and Aubert, X. L. (1999). OOV-detection in large vocabulary system using automatically defined word-fragments as fillers. In *EUROSPEECH*, pages 49–52.

[Kneser and Ney, 1995] Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 181–184 vol.1.

[Kombrink et al., 2009] Kombrink, S., Burget, L., Matejka, P., Karafiát, M., and Hermansky, H. (2009). Posterior-based out of vocabulary word detection in telephone speech. In *ISCA INTERSPEECH*, pages 80–83.

[Kombrink et al., 2012] Kombrink, S., Hannemann, M., and Lukáš Burget (2012). *Detection and Identification of Rare Audiovisual Cues*, chapter Out-of-Vocabulary Word Detection and Beyond, pages 57–65. Springer Berlin Heidelberg.

[Krstovski et al., 2013] Krstovski, K., Smith, D. A., Wallach, H. M., and McGregor, A. (2013). Efficient nearest-neighbor search in the probability simplex. In *Proceedings of the 2013 Conference on the Theory of Information Retrieval*, ICTIR '13, pages 22:101–22:108, New York, NY, USA. ACM.

[Kumar et al., 2012] Kumar, R., Prasad, R., Ananthakrishnan, S., Vembu, A. N., Stallard, D., Tsakalidis, S., and Natarajan, P. (2012). Detecting oov named entities in conversational speech. In *ISCA INTERSPEECH*, pages 2354–2357.

[Lamel et al., 1991] Lamel, L. F., luc Gauvain, J., Eskenazi, M., and Limsi-cnrs, M. E. (1991). Bref, a large vocabulary spoken corpus for french. In *Eurospeech*, pages 505–508.

[Larochelle et al., 2009] Larochelle, H., Bengio, Y., Louradour, J., and Lamblin, P. (2009). Exploring strategies for training deep neural networks. *J. Mach. Learn. Res.*, 10:1–40.

[Lau et al., 1993] Lau, R., Rosenfeld, R., and Roukos, S. (1993). Trigger-based language models: a maximum entropy approach. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 2, pages 45–48 vol.2.

[Lecorvé et al., 2011] Lecorvé, G., Gravier, G., and Sébillot, P. (2011). Automatically finding semantically consistent n-grams to add new words in LVCSR systems. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4676–4679.

[Lecouteux et al., 2009] Lecouteux, B., Linarès, G., and Favre, B. (2009). Combined low level and high level features for out-of-vocabulary word detection. In *ISCA INTERSPEECH*, pages 1187–1190.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Orr, G. B., and Müller, K. R. (1998). *Neural Networks: Tricks of the Trade*, chapter Efficient BackProp, pages 9–50. Springer Berlin Heidelberg.

[Lee and Kawahara, 2009] Lee, A. and Kawahara, T. (2009). Recent development of open-source speech recognition engine julius. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 131–137.

[Lee and Seung, 1999] Lee, D. D. and Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791.

[Levenshtein, 1966] Levenshtein, V. I. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707.

[Levy and Goldberg, 2014] Levy, O. and Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, NIPS'14, pages 2177–2185, Cambridge, MA, USA. MIT Press.

[Lin et al., 2007] Lin, H., Bilmes, J., Vergyri, D., and Kirchhoff, K. (2007). Oov detection by joint word/phone lattice alignment. In *IEEE Workshop on Automatic Speech Recognition Understanding (ASRU)*, pages 478–483.

[Ling et al., 2015] Ling, W., Tsvetkov, Y., Amir, S., Fermandez, R., Dyer, C., Black, A. W., Trancoso, I., and Lin, C.-C. (2015). Not all contexts are created equal: Better word representations with variable attention. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1367–1372.

[Liu et al., 2007] Liu, C.-E., Thambiratnam, K., and Seide, F. (2007). Online vocabulary adaptation using limited adaptation data. In *ISCA INTERSPEECH*, pages 1821–1824.

[Lund and Burgess, 1996] Lund, K. and Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, & Computers*, 28(2):203–208.

[Lund et al., 1995] Lund, K., Burgess, C., and Atchley, R. A. (1995). Semantic and associative priming in high-dimensional semantic space. In *Proceedings of the 17th annual conference of the Cognitive Science Society*, volume 17, pages 660–665.

[Ma et al., 2015a] Ma, X., Wang, X., and Wang, D. (2015a). Low-frequency word enhancement with similar pairs in speech recognition. In *IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pages 343–347.

[Ma et al., 2015b] Ma, X., Wang, X., Wang, D., and Zhang, Z. (2015b). Recognize foreign low-frequency words with similar pairs. In *16th Annual Conference of the International Speech Communication Association - INTERSPEECH*, pages 458–462.

[Maas et al., 2011] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 142–150, Stroudsburg, PA, USA.

[Maergner et al., 2012] Maergner, P., Waibel, A., and Lane, I. (2012). Unsupervised vocabulary selection for real-time speech recognition of lectures. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4417–4420.

[Manning et al., 2008a] Manning, C. D., Raghavan, P., and Schütze, H. (2008a). *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK.

[Manning et al., 2008b] Manning, C. D., Raghavan, P., and Schütze, H. (2008b). Scoring, term weighting, and the vector space model. In *Introduction to Information Retrieval*, pages 100–123. Cambridge University Press. Cambridge Books Online.

[Marin et al., 2012] Marin, A., Kwiatkowski, T., Ostendorf, M., and Zettlemoyer, L. (2012). Using syntactic and confusion network structure for out-of-vocabulary word detection. In *IEEE Spoken Language Technology Workshop (SLT)*, pages 159–164.

[Martins et al., 2006] Martins, C., Texeira, A., and Neto, J. (2006). Dynamic vocabulary adaptation for a daily and real-time broadcast news transcription system. In *IEEE Spoken Language Technology Workshop (SLT)*, pages 146–149.

[Martins et al., 2007] Martins, C., Texeira, A., and Neto, J. (2007). Dynamic language modeling for a daily broadcast news transcription system. In *IEEE Workshop on Automatic Speech Recognition Understanding (ASRU)*, pages 165–170.

[Meng et al., 2010] Meng, S., Wang, L.-F., Lin, Y.-M., Li, G., Thambiratnam, K., and Seide, F. (2010). Vocabulary and language model adaptation using just one file. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5410–5413.

[Miao et al., 2015] Miao, Y., Gowayyed, M., and Metze, F. (2015). Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 167–174.

[Mikolov, 2012] Mikolov, T. (2012). *Statistical Language Models Based on Neural Networks*. PhD thesis, Ph. D. thesis, Brno University of Technology.

[Mikolov et al., 2013a] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

[Mikolov et al., 2010] Mikolov, T., Karafiát, M., Burget, L., Černocký, J., and Khudanpur, S. (2010). Recurrent neural network based language model. In

Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH 2010), pages 1045–1048.

[Mikolov et al., 2013b] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

[Mikolov et al., 2013c] Mikolov, T., Yih, W.-t., and Zweig, G. (2013c). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751.

[Ming Sun, 2015] Ming Sun, Yun-Nung Chen, A. I. R. (2015). Learning OOV through semantic relatedness in spoken dialog systems. In *ISCA INTERSPEECH*, pages 1453–1457.

[Minka and Lafferty, 2002] Minka, T. and Lafferty, J. (2002). Expectation-propagation for the generative aspect model. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, UAI'02, pages 352–359, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

[Mohamed et al., 2012] Mohamed, A., Dahl, G. E., and Hinton, G. (2012). Acoustic modeling using deep belief networks. *Trans. Audio, Speech and Lang. Proc.*, 20(1):14–22.

[Mohamed et al., 2009] Mohamed, A., Dahl, G. E., and Hinton, G. E. (2009). Deep belief networks for phone recognition. In *NIPS Workshop on Deep Learning for Speech Recognition and Related Applications*.

[Mohri et al., 2010] Mohri, M., Moreno, P. J., and Weinstein, E. (2010). Discriminative topic segmentation of text and speech. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, pages 533–540.

[Morchid et al., 2014] Morchid, M., Dufour, R., Bousquet, P. M., Bouallegue, M., Linarès, G., and Mori, R. D. (2014). Improving dialogue classification using a topic space representation and a gaussian classifier based on the decision rule. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 126–130.

[Mori et al., 2008] Mori, R. D., Bechet, F., Hakkani-Tur, D., McTear, M., Riccardi, G., and Tur, G. (2008). Spoken language understanding. *IEEE Signal Processing Magazine*, 25(3):50–58.

[Morin and Bengio, 2005] Morin, F. and Bengio, Y. (2005). Hierarchical probabilistic neural network language model. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, pages 246–252. Citeseer.

[Nam et al., 2014] Nam, J., Kim, J., Loza Mencía, E., Gurevych, I., and Fürnkranz, J. (2014). Large-scale multi-label text classification - revisiting neural networks. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD-14), Part 2*, volume 8725, pages 437–452.

[Naptali et al., 2012] Naptali, W., Tsuchiya, M., and Nakagawa, S. (2012). Class-based n-gram language model for new words using out-of-vocabulary to in-vocabulary similarity. *IEICE Transactions*, 95-D(9):2308–2317.

[Navarro, 2001] Navarro, G. (2001). A guided tour to approximate string matching. *ACM Comput. Surv.*, 33(1):31–88.

[Newman et al., 2006] Newman, D., Chemudugunta, C., and Smyth, P. (2006). Statistical entity-topic models. In *12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 680–686.

[Niraula et al., 2013] Niraula, N., Banjade, R., Ştefănescu, D., and Rus, V. (2013). *Statistical Language and Speech Processing: First International Conference, SLSP 2013, Tarragona, Spain, July 29-31, 2013. Proceedings*, chapter Experiments with Semantic Similarity Measures Based on LDA and LSA, pages 188–199. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Nkairi et al., 2013] Nkairi, I., Illina, I., Linarès, G., and Fohr, D. (2013). Exploring temporal context in diachronic text documents for automatic OOV proper name retrieval. In *Language & Technology Conference*, pages 540–544.

[Oger et al., 2008a] Oger, S., Linarès, G., and Bechet, F. (2008a). Local methods for on-demand out-of-vocabulary word retrieval. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*.

[Oger et al., 2008b] Oger, S., Linarès, G., Béchet, F., and Nocera, P. (2008b). On-demand new word learning using world wide web. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4305–4308.

[Oger et al., 2009] Oger, S., Popescu, V., and Linarès, G. (2009). Using the world wide web for learning new words in continuous speech recognition tasks: two case studies. In *Speech and Computer Conference (SPECOM)*, pages 76–81.

[Orosanu and Jouvet, 2015] Orosanu, L. and Jouvet, D. (2015). Adding new words into a language model using parameters of known words with similar behavior. In *International Conference on Natural Language and Speech Processing*, Alger, Algeria.

[Palmer and Ostendorf, 2005] Palmer, D. D. and Ostendorf, M. (2005). Improving out-of-vocabulary name resolution. *Computer Speech & Language*, 19(1):107 – 128.

[Pan et al., 2005] Pan, Y.-C., Liu, Y.-Y., and Lee, L.-S. (2005). Named entity recognition from spoken documents using global evidences and external knowledge sources with applications on mandarin chinese. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 296–301.

[Pang and Lee, 2005] Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 115–124, Stroudsburg, PA, USA.

[Parada et al., 2010a] Parada, C., Dredze, M., Filimonov, D., and Jelinek, F. (2010a). Contextual information improves oov detection in speech. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 216–224.

[Parada et al., 2011a] Parada, C., Dredze, M., and Jelinek, F. (2011a). OOV sensitive named-entity recognition in speech. In *INTERSPEECH 2011, 12th Annual Conference of the International Speech Communication Association*, pages 2085–2088.

[Parada et al., 2011b] Parada, C., Dredze, M., Sethy, A., and Rastrow, A. (2011b). Learning sub-word units for open vocabulary speech recognition. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, pages 712–721.

[Parada et al., 2010b] Parada, C., Sethy, A., Dredze, M., and Jelinek, F. (2010b). A spoken term detection framework for recovering out-of-vocabulary words using the web. In *ISCA INTERSPEECH*, pages 1269–1272.

[Parada, 2011] Parada, M. C. (2011). *Learning Sub-word Units and Exploiting Contextual Information for Open Vocabulary Speech Recognition*. PhD thesis, Johns Hopkins University. AAI3483278.

[Paul and Baker, 1992] Paul, D. B. and Baker, J. M. (1992). The design for the wall street journal-based csr corpus. In *Proceedings of the Workshop on Speech and Natural Language*, HLT '91, pages 357–362, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Pennington et al., 2014] Pennington, J., Socher, R., and Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

[Povey et al., 2011] Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (2011). The kaldi speech recognition toolkit. In *IEEE Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society. IEEE Catalog No.: CFP11SRW-USB.

[Powers, 1998] Powers, D. M. W. (1998). Applications and explanations of zipf's law. In *Proceedings of the Joint Conferences on New Methods in Language Processing and Computational Natural Language Learning*, NeMLaP3/CoNLL '98, pages 151–160, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Pražák et al., 2007] Pražák, A., Ircing, P., and Müller, L. (2007). Language model adaptation using different class-based models. In *SPECOM 2007 Proceedings*, pages 449–454, Moscow.

[Pritchard et al., 2000] Pritchard, J. K., Stephens, M., and Donnelly, P. (2000). Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959.

[Qin, 2013] Qin, L. (2013). *Learning Out-of-Vocabulary Words in Automatic Speech Recognition*. PhD thesis, Language Technologies Institute, School of Computer Science, Carnegie Mellon University.

[Qin and Rudnicky, 2012] Qin, L. and Rudnicky, A. (2012). Oov word detection using hybrid models with mixed types of fragments. In *ISCA INTERSPEECH*, pages 2450–2453.

[Qin et al., 2011] Qin, L., Sun, M., and Rudnicky, A. (2011). Oov detection and recovery using hybrid models with different fragments. In *ISCA INTERSPEECH*, pages 1913–1916.

[Qin et al., 2012] Qin, L., Sun, M., and Rudnicky, A. (2012). System combination for out-of-vocabulary word detection. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4817–4820.

[Rabiner and Juang, 1993] Rabiner, L. and Juang, B.-H. (1993). *Fundamentals of Speech Recognition*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[Rabiner, 1989] Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

[Rastrow et al., 2009a] Rastrow, A., Sethy, A., and Ramabhadran, B. (2009a). A new method for oov detection using hybrid word/fragment system. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3953–3956.

[Rastrow et al., 2009b] Rastrow, A., Sethy, A., Ramabhadran, B., and Jelinek, F. (2009b). Towards using hybrid word and fragment units for vocabulary independent LVCSR systems. In *ISCA INTERSPEECH*, pages 1931–1934.

[Riccardi and Gorin, 1998] Riccardi, G. and Gorin, A. L. (1998). Stochastic language models for speech recognition and understanding. In *5th International Conference on Spoken Language Processing (ICSLP)*.

[Role and Nadif, 2011] Role, F. and Nadif, M. (2011). Handling the impact of low frequency events on co-occurrence based measures of word similarity - a case study of pointwise mutual information. In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval*, pages 226–231.

[Rong, 2014] Rong, X. (2014). word2vec parameter learning explained. *CoRR*, abs/1411.2738.

[Rosenfeld, 1996] Rosenfeld, R. (1996). A maximum entropy approach to adaptive statistical language modelling. *Computer Speech & Language*, 10(3):187 – 228.

[Rosenfeld, 2000] Rosenfeld, R. (2000). Two decades of statistical language modeling: where do we go from here? *Proceedings of the IEEE*, 88(8):1270–1278.

[Sainath et al., 2013] Sainath, T. N., Mohamed, A., Kingsbury, B., and Ramabhadran, B. (2013). Deep convolutional neural networks for LVCSR. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8614–8618.

[Schmid, 1994a] Schmid, H. (1994a). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

[Schmid, 1994b] Schmid, H. (1994b). TreeTagger - a language independent part-of-speech tagger. http://www.cis.uni-muenchen.de/ schmid/tools/TreeTagger/. Accessed: 2016-03-10.

[Schmidhuber, 2014] Schmidhuber, J. (2014). Deep learning in neural networks: An overview. *CoRR*, abs/1404.7828.

[Schwenk and Gauvain, 2002] Schwenk, H. and Gauvain, J. L. (2002). Connectionist language modeling for large vocabulary continuous speech recognition. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 765–768.

[Senay et al., 2013] Senay, G., Bigot, B., Dufour, R., Linarès, G., and Fredouille, C. (2013). Person name spotting by combining acoustic matching and LDA topic models. In *14th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 1584–1588.

[Seneff, 2005] Seneff, O. S. S. (2005). A two-pass strategy for handling oovs in a large vocabulary recognition task. In *ISCA INTERSPEECH*, pages 1669–1672.

[Shaik et al., 2015] Shaik, M. A. B., Mousa, A. E. D., Hahn, S., Schlüter, R., and Ney, H. (2015). Improved strategies for a zero oov rate LVCSR system. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5048–5052.

[Shaik et al., 2012] Shaik, M. A. B., Rybach, D., Hahn, S., Schlüter, R., and Ney, H. (2012). Hierarchical hybrid language models for open vocabulary continuous speech recognition using wfst. In *Workshop on Statistical and Perceptual Audition*, pages 46–51.

[Sheikh et al., 2015a] Sheikh, I., Illina, I., and Fohr, D. (2015a). Study of entity-topic models for OOV proper name retrieval. In *16th Annual Conference of the International Speech Communication Association - INTERSPEECH*, pages 3506–3510.

[Sheikh et al., 2016a] Sheikh, I., Illina, I., and Fohr, D. (2016a). How diachronic text corpora affect context based retrieval of oov proper names for audio news. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3851–3855.

[Sheikh et al., 2015b] Sheikh, I., Illina, I., Fohr, D., and Linarès, G. (2015b). OOV proper name retrieval using topic and lexical context models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5291–5295.

[Sheikh et al., 2016b] Sheikh, I., Illina, I., Fohr, D., and Linarès, G. (2016b). Document level semantic context for retrieving OOV proper names. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6050–6054.

[Sheikh et al., 2016c] Sheikh, I., Illina, I., Fohr, D., and Linarès, G. (2016c). Improved neural bag-of-words model to retrieve out-of-vocabulary words in speech recognition. In *17th Annual Conference of the International Speech Communication Association (INTERSPEECH)*.

[Sheikh et al., 2016d] Sheikh, I., Illina, I., Fohr, D., and Linarès, G. (2016d). Learning word importance with the neural bag-of-words model. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 222–229, Berlin, Germany. Association for Computational Linguistics.

[Sheikh et al., 2015c] Sheikh, I. A., Illina, I., Fohr, D., and Linarès, G. (2015c). Learning to retrieve out-of-vocabulary words in speech recognition. *CoRR*, abs/1511.05389.

[Smucker et al., 2007] Smucker, M. D., Allan, J., and Carterette, B. (2007). A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, pages 623–632.

[Socher et al., 2013] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642.

[Sønderby et al., 2015] Sønderby, S. K., Sønderby, C. K., Nielsen, H., and Winther, O. (2015). Convolutional lstm networks for subcellular localization of proteins. In *Second International Conference on Algorithms for Computational Biology (AlCoB)*, pages 68–80. Springer International Publishing.

[Song and Cai, 2015] Song, W. and Cai, J. (2015). End-to-end deep neural network for automatic speech recognition. http://cs224d.stanford.edu/reports/SongWilliam.pdf.

[Specia et al., 2016] Specia, L., Frank, S., Sima'an, K., and Elliott, D. (2016). A shared task on multimodal machine translation and crosslingual image description. In *Proceedings of the First Conference on Machine Translation*, pages 543–553, Berlin, Germany. Association for Computational Linguistics.

[Srivastava et al., 2014] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

[Stolcke, 2002] Stolcke, A. (2002). SRILM - an extensible language modeling toolkit. In *Proceedings International Conference on Spoken Language Processing*, pages 257–286.

[Suhm et al., 1993] Suhm, B., Woszczyna, M., and Waibel, A. (1993). Detection and transcription of new words. In *EUROSPEECH*, pages 2179–2182.

[Sundermeyer et al., 2015] Sundermeyer, M., Ney, H., and Schl uter, R. (2015). From feedforward to recurrent lstm neural networks for language modeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):517–529.

[Tai et al., 2015] Tai, K. S., Socher, R., and Manning, C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566.

[Tur and De Mori, 2011] Tur, G. and De Mori, R. (2011). *Spoken Language Understanding*, chapter Introduction, pages 1–7. John Wiley & Sons, Ltd.

[Tür et al., 2001] Tür, G., Stolcke, A., Hakkani-Tür, D., and Shriberg, E. (2001). Integrating prosodic and lexical cues for automatic topic segmentation. *Comput. Linguist.*, 27(1):31–57.

[Turney and Pantel, 2010] Turney, P. D. and Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *J. Artif. Int. Res.*, 37(1):141–188.

[van der Maaten and Hinton, 2008] van der Maaten, L. and Hinton, G. E. (2008). Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

[Wallach et al., 2009] Wallach, H. M., Mimno, D. M., and McCallum, A. (2009). Rethinking lda: Why priors matter. In Bengio, Y., Schuurmans, D., Lafferty, J. D., Williams, C. K. I., and Culotta, A., editors, *Advances in Neural Information Processing Systems 22*, pages 1973–1981. Curran Associates, Inc.

[Wang et al., 2015] Wang, P., Xu, J., Xu, B., Liu, C., Zhang, H., Wang, F., and Hao, H. (2015). Semantic clustering and convolutional neural network for short text categorization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 352–357.

[Wang and Manning, 2012] Wang, S. and Manning, C. D. (2012). Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers - Volume 2*, ACL '12, pages 90–94, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Wang, 2003] Wang, W. (2003). Techniques for effective vocabulary selection. In *8th European Conference on Speech Communication and Technology*, pages 245–248.

[White et al., 2008] White, C., Zweig, G., Burget, L., Schwarz, P., and Hermansky, H. (2008). Confidence estimation, OOV detection and language ID using phone-to-word transduction and phone-level alignments. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4085–4088.

[Wintrode, 2011] Wintrode, J. (2011). Using latent topic features to improve binary classification of spoken documents. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5544–5547.

[Xu et al., 2015] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A. C., Salakhutdinov, R., Zemel, R. S., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 2048–2057.

[Yazgan and Saraclar, 2004] Yazgan, A. and Saraclar, M. (2004). Hybrid language models for out of vocabulary word detection in large vocabulary conver-

sational speech recognition. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages I–745–8 vol.1.

[Young, 1996] Young, S. (1996). A review of large-vocabulary continuous-speech. *IEEE Signal Processing Magazine*, 13(5):45–.

[Young and Chase, 1998] Young, S. and Chase, L. (1998). Speech recognition evaluation: a review of the U.S. CSR and LVCSR programmes. *Computer Speech & Language*, 12(4):263 – 279.

[Young et al., 2006] Young, S. J., Kershaw, D., Odell, J., Ollason, D., Valtchev, V., and Woodland, P. (2006). *The HTK Book Version 3.4*. Cambridge University Press.

[Young, 1994] Young, S. R. (1994). Recognition confidence measures: Detection of misrecognitions and out-of-vocabulary words. Technical report, Carnegie Mellon University, Pittsburgh, PA, USA.

[Yu and Deng, 2014] Yu, D. and Deng, L. (2014). *Automatic Speech Recognition: A Deep Learning Approach*. Springer Publishing Company.

[Zeiler, 2012] Zeiler, M. D. (2012). ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701.

[Zhang and Wallace, 2015] Zhang, Y. and Wallace, B. (2015). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *CoRR*, abs/1510.03820.

[Zue et al., 1990] Zue, V., Glass, J., Goodine, D., Leung, H., McCandless, M., Phillips, M., Polifroni, J., and Seneff, S. (1990). Recent progress on the voyager system. In *Proceedings of the Workshop on Speech and Natural Language*, HLT '90, pages 206–211, Stroudsburg, PA, USA. Association for Computational Linguistics.

APPENDIX A

Dirichlet-Multinomial Distribution and Latent Dirichlet Allocation

A.1 Posterior Inference for Dirichlet-Multinomial Compound Distribution

Let us consider a set of outcomes X consisting of N i.i.d. draws from a multinomial random variable w. Let V be the number of different possible outcomes, for example number of sides of a dice or the number of words in the vocabulary of corpus. If p_t is the probability of each $t \in V$ then $\sum p_t = 1$. Also if $n^{(t)}$ is the count of each t in X then $\sum n^{(t)} = N$. Let us denote $\vec{p} = \{p_t\}_{t=1,2,...,V}$, then the likelihood of generating X can be written as:

$$\begin{aligned} L(X|\vec{p}) &= \prod_{i=1}^{N} \Pr(w_i|\vec{p}) \\ &= \prod_{t=1}^{V} p_t^{n^{(t)}} \end{aligned} \tag{A.1}$$

Following a Bayesian framework the parameters $\vec{p}$ can can be modelled with a conjugate Dirichlet distribution. This is depicted in the plate diagram in Figure A.1.

The Dirichlet prior distribution itself is given as:

$$\text{Dirichlet}(\vec{p}\,|\vec{\alpha}) = \frac{1}{B(\vec{\alpha})} \prod_{k=1}^{V} p_k^{\alpha_k - 1} \tag{A.2}$$

in which normalising constant $B(\vec{\alpha})$ is the multivariate Beta function, which can be expressed as: $B(\vec{\alpha}) = \frac{\prod_{k=1}^{V} \Gamma(\alpha_k)}{\Gamma(\sum_{k=1}^{V} \alpha_k)}$ where $\Gamma(n) = (n-1)!$ for positive integer.

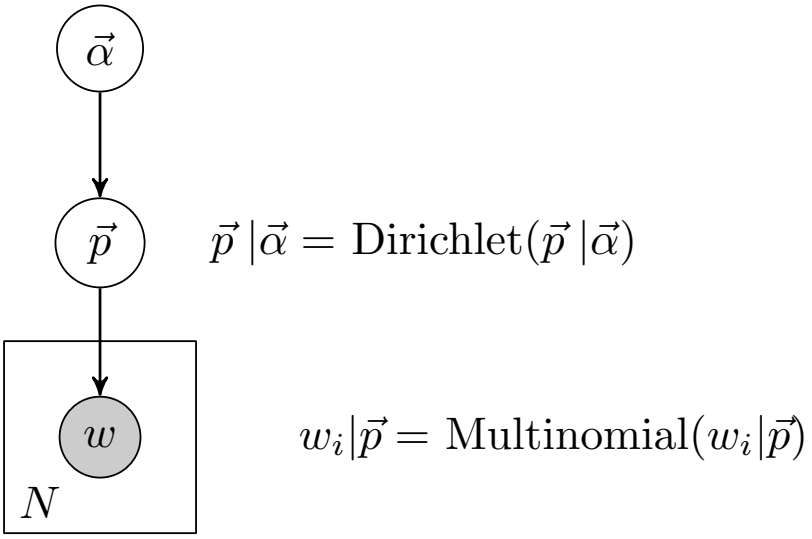


Figure A.1: Plate Diagram for the Dirichlet-Multinomial Compound Distribution. ($\rightarrow$ on variables is used to denote the multivariate chracteristics.)

The conjugate Dirichlet prior enables us to simplify the (Bayesian) posterior inference of the parameters $\vec{p}$, which can be formulated as:

$$\begin{aligned} \vec{p}\,|X, \vec{\alpha} &= \frac{\Pr(X|\vec{p})\ \Pr(\vec{p}\,|\vec{\alpha})}{\Pr(X)} \\ &= \frac{\prod_{i=1}^{N} \Pr(w_i|\vec{p})\ \Pr(\vec{p}\,|\vec{\alpha})}{\int_p \prod_{i=1}^{N} \Pr(w_i|\vec{p})\ \Pr(\vec{p}\,|\vec{\alpha})\ d\vec{p}} \end{aligned} \tag{A.3}$$

Using Equation A.1 and A.2 we can rewrite Equation A.3 as:

$$\begin{aligned} \vec{p}\,|X, \vec{\alpha} &= \frac{\prod_{t=1}^{V} p_t^{n^{(t)}} \frac{1}{B(\vec{\alpha})}\ p_t^{\alpha_t - 1}}{\int_p \prod_{t=1}^{V} p_t^{n^{(t)}} \frac{1}{B(\vec{\alpha})}\ p_t^{\alpha_t - 1}\ d\vec{p}} \\ &= \frac{\prod_{t=1}^{V} p^{n^{(t)} + \alpha_t - 1}}{\int_p \prod_{t=1}^{V} p_t^{n^{(t)} + \alpha_t - 1}\ d\vec{p}} \end{aligned} \tag{A.4}$$

It can be seen that the numerator is of the form of a Dirichlet distribution without the normalising constant, which indeed is provide by the integral in the denominator. Thus the posterior estimate is also a Dirichlet distribution which merges the multinomial observations with the prior pseudo-counts $\vec{\alpha}$, and can be expressed as:

$$\vec{p}\,|X, \vec{\alpha} = \text{Dirichlet}(\vec{p}\,|\vec{n} + \vec{\alpha}) \tag{A.5}$$

Using this generative model, we can write the probability for a set of individual outcomes as:

$$\begin{aligned} p(W|\vec{\alpha}) &= \int_p \Pr(W|\vec{p})\ \Pr(\vec{p}\,|\vec{\alpha})\ d\vec{p} \\ &= \int_p \prod_{i=1}^{N} \Pr(w_i|\vec{p})\ \Pr(\vec{p}\,|\vec{\alpha})\ d\vec{p} \\ &= \int_p \prod_{t=1}^{V} p_t^{n^{(t)}} \frac{1}{B(\vec{\alpha})}\ p_t^{\alpha_t - 1}\ d\vec{p} \\ &= \int_p \frac{1}{B(\vec{\alpha})}\ \prod_{t=1}^{V} p_t^{n^{(t)} + \alpha_t - 1}\ d\vec{p} \\ &= \frac{B(\vec{n} + \vec{\alpha})}{B(\vec{\alpha})} \end{aligned} \tag{A.6}$$

A.2 Gibbs Sampling to Estimate LDA Model Parameters

Algorithm 1 : LDA model parameter estimation (Part I)

Input: words $w_{d,i}$ observed in each document d

Output: topic assignments $z_{d,i}$; counts $n_{d,k}$, $n_{k,v}$

```
procedure ESTIMATION OF TOPIC ASSIGNMENTS z_{d,i}
    INIT()
    for large number of iterations do
        for d = 1 : D do
            for i = 1 : N_d do
                z_{d,i} ← GIBBS_SAMPLER(d, i)
            end for
        end for
    end for
    θ ← ESTIMATE_THETA()
    φ ← ESTIMATE_PHI()
end procedure

function INIT()
    {n_{d,k}} = 0, {n_{k,v}} = 0
    for d = 1 : D do
        for i = 1 : N_d do
            z_{d,i} ← randomly assign from {1, 2, 3, ..., T}
            n_{d,k}++, n_{k,v}++
        end for
    end for
end function

function GIBBS_SAMPLER(d, i)
    v = w_{d,i}
    topic = z_{d,i}
    n_{d,topic}−−, n_{topic,v}−−
    for k = 1 : T do
        p(z = k|·) = (n_{d,k} + α)  (n_{k,v} + β)/(Σ_{v=1}^{V} n_{k,v} + β)
    end for
    topic = sample from p(z|·)
    n_{d,topic}++, n_{topic,v}++
    return topic
end function
```

Algorithm 2 : LDA model parameter estimation (Part II)

Input: topic assignments $z_{d,i}$ for each word in each document; counts $n_{d,k}$, $n_{k,v}$
Output: parameters θ, ϕ

```
function ESTIMATE_THETA()
    for d = 1 : D do
        for k = 1 : T do
            θ_{d,k} ← (n_{d,k} + α)/(Σ_{k=1}^{T} n_{d,k} + α)
        end for
    end for
end function

function ESTIMATE_PHI()
    for k = 1 : T do
        for v = 1 : V do
            φ_{k,v} ← (n_{k,v} + β)/(Σ_{v=1}^{V} n_{k,v} + β)
        end for
    end for
end function
```

APPENDIX B

OOV PN Retrieval Performances

B.1 Rank-Frequency Distribution for Retrieval with Word Embedding Methods

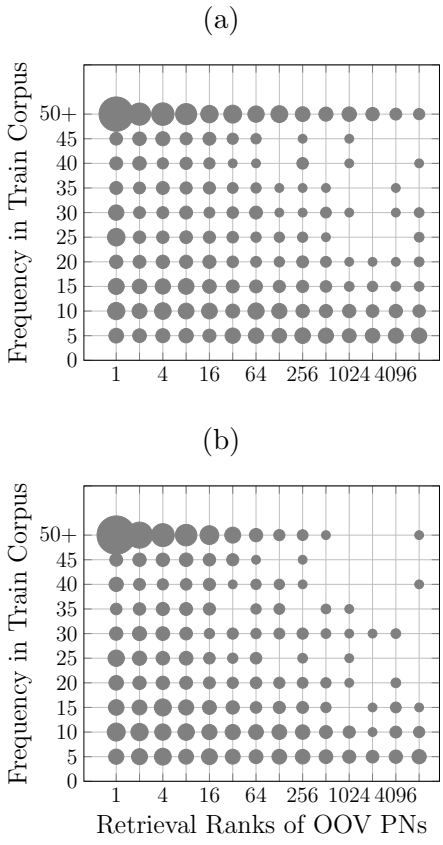


Figure B.1: Rank-Frequency distribution for retrieval of OOV PNs with (a) CBOW Method I, CBOW-MI in Figure 5.5 (b) CBOW Method II, CBOW-MII.

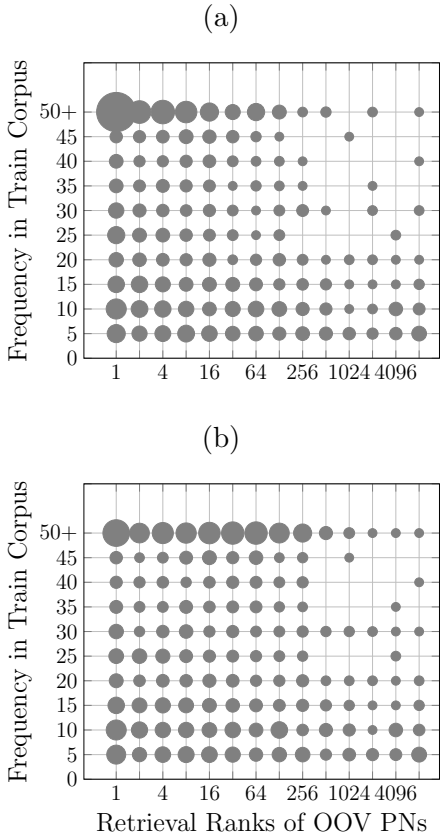


Figure B.2: Rank-Frequency distribution for retrieval of OOV PNs with (a) Skip-gram Method I, SG-MI in Figure 5.5 (b) Skip-gram Method II, SG-MII.