Document Embeddings as a Means of Domain Adaptation for the Machine Translation
Ph.D. Thesis Proposal

Tom Kocmi
Faculty of Mathematics and Physics, Charles University
Institute of Formal and Applied Linguistics
kocmi@ufal.mff.cuni.cz

1 Introduction

In recent years the field of Natural Language Processing (NLP) went through massive changes as neural systems reached better performance in many NLP tasks compared to previous mostly statistical approaches.

This development has been very apparent also in machine translation (MT). MT started with rule based approaches which worked successfully for small domains. Generic MT was first reached with statistical methods, the early word-based and the late phrase-based dominant approaches, that build upon large training data. The current change is due to the first successful application of deep-learning methods (neural networks) to the task, giving rise to neural MT (NMT; Collobert et al. (2011); Sutskever et al. (2014)).

MT systems are very sensitive to the domain(s) they were trained on because each domain has its own style, sentence structure, and terminology. There is often a mismatch between the domain in which training data are available and the target domain in which the MT system is used. If there is a strong disparity between training and testing data, translation quality will be dramatically deteriorated. Word ambiguities are often an issue for machine translation systems. For instance, the English word ”administer” has to be translated differently if it appears in medical or political contexts.

Koehn and Knowles (2017) have done a thorough comparison of statistical and neural MT and described six challenges of NMT. One of the main challenges is that the NMT systems have lower quality for out of domain translation, to the point that they completely sacrifice adequacy for the sake of fluency.

This problem is usually solved with the use of domain adaptation, where the trained generic NMT system is further trained on in-domain data, which improves the performance of in-domain translation, but deteriorates the quality of generic translation. It assumes that users will translate only within trained domain or have separate model for each of the domains, which is not the scenario in this work.

We focus on a generic NMT system, which utilizes the context of source side of translated document in order to improve the translation for the particular document domain. We define the model which is not specialized for one domain but rather changes its behavior based on features of the given document to be translated. As a solution we propose to use the vector document representation (also called the embedding) as an additional source of information about the document domain.

The question of knowledge representation is central to many language understanding problems: How to capture the essential meaning of a text in a machine-understandable format (or representation). Formal representation of language has been at the heart of linguistic studies for centuries. They developed many various theories and representations. Chomsky defined a generative grammar (Chomsky, 1964) and used a system of rules to generate grammatical sentences. The Functional Generative Description (Sgall et al., 1986) describes language in five layers from phonetics up to the tectogramatical layer and defines various dependencies between them. Many natural language processing approaches use these theories for representation of language in a machine understandable way.

The question of knowledge representation emerged again with the rise of neural networks and become even more important, since neural networks favor soft vector descriptions.

The first experiments with neural networks in NLP used the so called one-hot representation, where each linguistic token, such as a character, word, phrase, etc. is represented by a vector of the dimension equal to the total size of the vocabulary with one non-zero value on the position assigned to the token. This representation is extremely inefficient, therefore Bengio et al. (2003) suggested using relatively low dimensional embeddings where each token is represented by vector of real values that are trained together with the neural network.

Not only characters or words can be represented as embeddings but also whole paragraphs or even documents. Such embeddings could contain essential information about the sentences and documents, such as genre, topic, writing style, specialized vocabulary, etc. And we want to use the soft document representation for the NMT improvement of in-domain translation.

This thesis proposal is structured as follows: in Section 2 we introduce embeddings and explain various approaches to embedding training as well as examples of the usage of document embeddings in NLP. In Section 3 we explain a method of domain adaptation in the NMT and also define what is the domain for us. Our proposed approach of using document embeddings as a means of domain adaptation is described in Section 4. We follow by summarization of experiments that we have already conducted as well as some preliminary results of our method in Section 5. The future work plan and a brief summary are outlined in the Sections 6 and 7.

2 Embeddings

Embeddings also known as vector representations (Bengio et al., 2003) are *the* interface between the world of discrete units of text and the continuous, differentiable world of neural networks. Embeddings are used for units of different granularity, from characters (Lee et al., 2016) through subword units (Sennrich et al., 2016c; Wu et al., 2016) and words up to sentences (Kiros et al., 2015), paragraphs or even whole documents (Le and Mikolov, 2014).

Embeddings represent the respective text unit as a vector in a highly dimensional space. They are almost never provided manually but discovered automatically in a neural network trained to carry out a particular task.

Most commonly used embeddings are the word embeddings [Pennington et al. 2014, Kocmi and Bojar 2016]. The best known are those by Mikolov et al. (2013) (word2vec), where the task is to predict the word from its neighboring words (CBOW) or the neighbors from the given word (Skip-gram). After the training on a huge corpus (usually billions of words), we extract for each word its corresponding weights from the neural network and consider them as the word embedding.

Word representations can exhibit an interesting correspondence between lexical relations and arithmetic operations in the vector space. The most famous example is the following:

$$v(\mathit{king}) - v(\mathit{man}) + v(\mathit{woman}) \approx v(\mathit{queen})$$

In other words, adding the vectors associated with the words ‘*king*’ and ‘*woman*’ while subtracting ‘*man*’ should be equal to the vector associated with the word ‘*queen*’. We can also say that the difference vectors $v(\mathit{king}) - v(\mathit{queen})$ and $v(\mathit{man}) - v(\mathit{woman})$ are almost identical and describe the gender relationship.

Following these successful techniques, researchers have tried to extend the models to go beyond word level to achieve phrase-level or sentence-level representations (Le and Mikolov, 2014; Wieting et al., 2015; Kiros et al., 2015).

2.1 Document Embeddings

For the vector representation of documents, usual **non-neural** ways are to use bag-of-words (BOW) or term frequency-inverse document frequency (TF-IDF) representations. Other widely adopted methods are generative topic models, such as latent semantic analysis (LSA) (Deerwester et al., 1990) and latent dirichlet allocation (LDA) (Blei et al., 2003). The former uses SVD to lower the dimensionality of TF-IDF matrix and the latter generates the mixture of topics based on the word assigned to clusters with the use of Gibbs sampling.

Previous methods are widely used in many real-world application for tasks of document clustering or keyword search. Unfortunately, they are not suitable to be used as document embeddings within neural networks, mainly due to the vast dimensionality up to the size of vocabulary or discrete representation of discovered features.

A simple low dimensional approach is to use a weighted average of word embeddings of all the words in the document. A more sophisticated approach is to combine the word vectors in an order given by a parse tree of a sentence using vector operations (Socher et al., 2013). Both approaches have weaknesses. The former approach, weighted averaging of word vectors, loses the word order in the same way as the standard bag-of-words models do. The latter approach, which uses a parse tree to combine word embeddings, has been shown to work for only sentences and not larger text units, due to its dependence on the sentence parsing.

A more robust approach is to utilize the neural network for the training of the document representations. We can divide the **neural approaches** into two separate groups: pretrained and specialized. The pretrained approaches are trained in isolation from the task they will be used on, in our case NMT. This allows them to be trained on huge monolingual corpora without the need for task-specific corpora. The specialized approaches are trained jointly with the task (MT) from a randomly initialized matrix.

2.1.1 Pretrained Embeddings

The former group of isolated approaches creates generalized vector representations usually trained on huge monolingual corpora. Inspired by the success of word2vec, Le and Mikolov (2014) developed doc2vec, which produces a vector representation of documents as well as words by solving a task of predicting words contained within the given document. Zhu and Hu (2017) improved doc2vec by focusing on the context of words in the document and assigning weights to more important words.

Dai et al. (2015) further examined doc2vec and found analogy features on Wikipedia articles similar to the analysis done on word2vec:

$$\begin{aligned} \mathcal{V}(\textit{Lady Gaga}) - v(\textit{American}) \\ + v(\textit{Japanese}) \approx \mathcal{V}(\textit{Ayumi Hamasaki}) \end{aligned}$$

The formula can be interpreted as that the subtraction of the word embedding for ‘*American*’ from a vector representing the document about ‘*Lady Gaga*’ with an addition of the word embedding for ‘*Japanese*’ generates a vector similar to the document embedding of ‘*Ayumi Hamasaki*’, the famous Japanese pop singer often dubbed the ”Empress of Pop”.

Kiros et al. (2015) laid down another interesting approach where, instead of predicting words from the document, they used a sentence as a token and predicted whole sentences in a similar manner as word2vec.

Wieting et al. (2015) utilized a paraphrase corpus in order to help embeddings of similar sentences to be close together in the vector space. They showed the ability of embeddings to improve general text similarity and entailment models.

The pretrained embeddings have been shown to improve the performance in various tasks (e.g. document similarity). Pretrained document embeddings can be used as another features for the neural network solving final task. This is advantageous because the NMT model does not grow notably in size, but the disadvantage is the inability to improve the document embeddings during the training of the final model. To the best of our knowledge pretrained document embeddings have never been used in the NMT task.

2.1.2 Specialized Embeddings

In this section we describe document embeddings defined as a part of a neural model solving a given task (coreference, sentiment analysis, MT, etc.). This approach has an advantage of being able to learn specific features of a document especially useful for a given task in contrast to general features in pretrained embeddings. The disadvantage is a substantial increase in the neural model size.

Lee et al. (2017) used a bidirectional LSTM recurrent neural network to generate embeddings of documents in order to extract coreference resolution. The authors showed state-of-the-art results. They noted as a disadvantage that the neural model size increases with the length of the document by $\mathcal{O}(length^4)$. They solved it by pruning long spans between words that are unlikely to belong to a coreference cluster. We believe that the pruning also helped them to overcome the problem with lengthy inputs, since as the LSTM network processes the input word by word from the beginning to the end (and backwards) it slowly forgets the knowledge about the earlier seen inputs, therefore the middle of the document is the least represented.

The problem with forgetting earlier information can be solved by separately encoding first the sentences from words followed by encoding the whole document from the sentence embeddings, this approach is called *hierarchical*.

Tang et al. (2015) used a convolutional neural network in order to encode sentences followed by LSTM recurrent neural network generating document embeddings. They used the document embedding for the task of sentiment analysis. Yang et al. (2016) have improved the hierarchical approach with the use of attention and the use of GRU layers for both encoding sentences from words and then documents from sentences. They received a significant performance improvement with this setup over other approaches in sentiment analysis.

The common problem with specialized embeddings is the need to fit the model as well as the whole document into the memory. This becomes a serious problem when training embeddings of long documents such as books. This can be one of the main reasons why there is a lack of research in field of neural document embeddings. We want to note that the focus of this work is not to solve the memory issue.

2.2 Uses of Document Embeddings

Vector representations of documents are useful for various applications. For example by the nearest neighbor search in the document vector space, we can address several important tasks:

1. Search for similar documents to a sample document. Useful for news stream personalization and recommendation: Quadrana et al. (2017); Wieting et al. (2015).

2. Automatic classification of documents, which can be used for document categorization (Djuric et al., 2015) or sentiment analysis of user feedback (Pang et al., 2008; Baroni et al., 2014).

3. Given a paragraph from a document, search for documents containing similar text, like in tasks of document retrieval and plagiarism detection (Lau and Baldwin, 2016; Engels et al., 2007).

4. Generate the most representative keywords or summarization passages from the given document. Useful for native advertising and summarization (Habibi and Popescu-Belis, 2015; Cheng and Lapata, 2016).

All these tasks are essential for multiple online applications. Document embeddings can also be used as an additional source of knowledge about a domain to improve other more difficult tasks.

Furthermore, there is yet a lack of research in the document level NMT and with that the use of document embeddings in the NMT. Garcia et al. (2017) uses word embeddings in order to improve document level consistency. We are unaware of any other related work to the best of our knowledge

3 Domain Adaptation

Domain adaptation is one of the key issues in Machine Translation. It generally encompasses terminology, domain and style adaptation. It has been successfully used in Statistical MT as well as in NMT (Gao and Zhang, 2002; Hildebrand et al., 2005; Luong and Manning, 2015). It is well known that an optimized model on a specific genre (news, speech, medical, literature...) obtains higher accuracy results than a generic system.

The main idea of the approach is to specialize a generic model already trained on generic data by adapting it on a specialized in-domain data.

In a typical domain adaptation setup, we have a large amount of out-of-domain bilingual training data for which we already have a trained neural network model. Given only a small additional amount of in-domain data, the challenge is to improve the translation performance in the new domain, which often leads to deteriorating the performance in the general domain.

3.1 Definition of Domain in Machine Translation

The definition of domain varies among the papers and in general, it is considered any set of instances from a dataset containing a common feature. In this section, we define what can be considered as a domain and what we want to pursue in our work.

In most of the papers concerning domain adaptation, the authors define the domain as the source of the dataset, this domain is closely related to the topic or genre of the documents (Hildebrand et al., 2005; Chu et al., 2017; Servan et al., 2016). Examples of such domain are subtitles, literature, news, medical reports, patents, IT and many more. All of them vary in the used vocabulary, style of writing and content. We demonstrate this domain on a following example:

Source EN: The trial ended in March.
News CS: Soudní proces skončil v březnu.
Scientific CS: Studie byla ukončena v březnu.

Another feature is the formality or informality of a document. Which is closely related to the honorifics in languages like Czech, German or

Japanese (Sennrich et al., 2016a). It is a way of encoding the relative social status of speakers to the readers and for many styles, like official documents, it is an important feature determining the quality of the translation.

Further, we can distinguish documents based on a sentiment. The sentiment tone of a text can change with machine translation (Glorot et al., 2011; Mohammad et al., 2016) mainly because of language differences and ambiguity. Another issue with the sentiment is that the same information can be written in positive, neutral or negative stance. For example:

Positive: The childhood is unforgettably playful.
Neutral: The childhood is the time to play.
Negative: The childhood is terrible without games.

We can go even further and distinguish documents based on the writing style of the author or expected style of a reader, as of formality of a speech, specialized vocabulary or dialects. Similarly to the sentiment, we can write the same information in various writing styles, dialects or slangs (Jeblee et al., 2014). For example:

Formal: My girlfriend is not enraged.
Casual: My darling isn't angry.
Slang: Bae ain't ticked off.

Lastly, we want to examine the structure of a document and consider different problems that arise in machine translation of poems, official letters, and other structured texts.

All these aspects can be used to improve the translation quality by trying to assure coherence throughout a document with the use of context represented through document embeddings. In those conditions we want document embeddings to represent a rough context of the whole text, which can be called a domain.

In our planned work, we do not focus on hard labeled domains but leave the neural model to discover useful features for the improvement of NMT performance by itself, in a continuous vector space. This could be useful to improve the translation performance also on sentences from documents that cannot be categorized exactly, for example news about the weather report or medical article in a popular science magazine.

Since during the training our NMT system does not have access to any hard-coded labels about the domain, we want to use them to find out how the document embeddings can represent domain information. We examine it by document classification, where documents from similar domains should have document embeddings close together in the vector space.

3.2 Domain Adaptation Approaches

A typical approach to domain adaptation in MT is to adapt a trained NMT system to a new domain with further training mainly on the in-domain data. Various methods of how to combine out-of-domain and in-domain data have been proposed (Luong and Manning, 2015; Freitag and Al-Onaizan, 2016; Servan et al., 2016; Chen et al., 2017). However, the adaptation process takes only a small portion of the training and it can quickly over-fit on the in-domain data as well as significantly lose performance on the general domain. The first problem is usually connected with the lack of in-domain data.

The lack of bilingual in-domain data can be solved by semi-supervised training, where large in-domain monolingual data of target language are first translated with a machine translation engine into the source language to generate parallel data, this approach is called *backtranslation*. Despite the lower quality of the translation, researchers showed significant improvements of neural translation in a given domain [Sennrich et al. 2016b, Sudarikov et al. 2017].

These approaches increase the performance on the domain in the expense of translation quality in the general domain. Researchers have tried to overcome this issue by adding the information about the domain to the neural network. This is usually done by introducing tag such as $\langle 2domain \rangle$ as starting symbol of the translated output (Sennrich et al., 2016a; Kobus et al., 2016; Chu et al., 2017). Although it slightly overcomes the problem with losing the performance on the general data it does not reach such improvements as the previous approaches. Another problem with this approach is the sparsity of document tags, where documents are labeled with hard-coded tags and the neural network cannot discover any other features of documents. For example finding out that the document is a news article about a new research discovery.

4 Proposed Approach

In this section we first describe how we want to proceed with the use of document embeddings as a means of domain adaptation.

Document embeddings can contain more information about a domain than a single tag distinguishing a small number of domains. In our work we want to investigate a question if document embeddings could improve translation quality in various domains without lowering the performance in a general domain.

The straightforward approach is exchanging the word embedding of the tag $\langle 2domain \rangle$ with a document embedding. It is the easiest way how to insert additional information into the neural network without changing the model architecture. However in the NMT models there could be a better way of appending the embedding into the neural network to better utilize the information. There are many cases where to append or add document embeddings. We list several places in the Bahdanau et al. (2014) architecture where domain information could lead to NMT improvement:[1]

- Adding document embedding to the word embedding of all input words in a similar manner as the positional embeddings are added in convolutional MT (Gehring et al., 2017). This could help the network to distinguish homonyms or polysemy in the early stage of encoding.

- Appending it to the attention mechanism, where it could improve the word alignment, especially when translating sentences with unusual wording like poems, European legislation, etc.

- Appending document embeddings to each state of either encoder, decoder or both. This could help to improve the coreference or the representation of source sentence.

In order to discover the best place to include the document embeddings we propose following experiment. We plan to run a baseline translation in a general domain and after reaching the best score we save the model and follow with standard domain adaptation on in-domain data until the model reaches the new best score. Afterwards we compare both models across all trained parameters and try to identify places which have been adapted the most, i.e. where the weights differ the most between models. We believe that this experiment could pinpoint possible places where we could include document embeddings.

After determining where the document embeddings should be placed, we start to compare various document embedding architectures, as described in Section 2.1 in order to determine which lead to the best performance and if they behave differently in various domain. The best technique is not yet determined.

Based on our intuition we believe that various places in the model and various document embeddings will lead to different behavior concerning domain, writing style, sentiment etc.

5 Experiments Conducted So Far

In this section we describe experiments we have concluded so far, starting from the creation of training corpora, followed by experiments with various embeddings, domain adaptation and back-translation. At the end we present so far unpublished results.

5.1 Document level corpus (Bojar et al., 2016a)

In the machine translation the usual procedure is to translate at the sentence level. This practice influences the creation of corpora, where most of the corpora are shuffled at the sentence level and the document information is lost. Another cause behind the sentence-level organization of corpora is the legal reason, where authors cannot publish data which could be reconstructed back into the original documents due to the license restrictions.

We have built a Czech-English bilingual corpus for machine translation CzEng 1.6 (Bojar et al., 2016a). This corpus contains 62M parallel sentences from various sources like subtitles, news, European legislation, medical documents, literature, technical reports and many more. All sentences are labeled with the source identification and an ID that allows reconstructing document parts of length up to 15 sentences.

This is the training corpus we use in our work. The corpus also contains development data with a distribution close to the training data. Furthermore we want to evaluate our experiments in vari-

[1] Due to the length restriction of the thesis proposal, we are omitting the description of the architecture and details of the document embeddings appending to the architecture.

ous domains and data distributions. Therefore we have generated another two document level bilingual development tests out of a News and Medical WMT 2017 translation task datasets.[2]

The news development set contains 2999 sentence pairs, which are contained within 116 documents. The medical domain contain 1511 sentences from 50 documents. Both corpora are from different sources than news and medical domain in the CzEng.

5.2 Character-Level Embeddings (Kocmi and Bojar, 2017b)

We have started our research with embeddings of individual characters, through embeddings of subwords units to the embeddings of words, where we have acquired useful knowledge about the embeddings space, which will help us with the document embeddings.

In the case study (Kocmi and Bojar, 2017b) we have proposed a neural network with a character level embeddings and demonstrated state-of-the-art results in a multilingual language identification task, where the goal is to detect multiple languages within one document. And on the monolingual language identification we get close to the state-of-the-art. We have showed that character level embeddings are a viable option due to the small size of the vocabulary, which is equal to number of characters in contrast to tens or hundreds of thousands words when using word embeddings.

5.3 Subword Units Embeddings (Kocmi and Bojar, 2016)

Recently subword units gained popularity in most of the neural network architectures (Sennrich et al., 2016c; Wu et al., 2016), mainly because the size of vocabulary is reasonable and in contrast to the character level embeddings they hold some information about the words. We have extended the renowned word2vec embeddings with subword units (Kocmi and Bojar, 2016) and showed their ability to attain morphosyntactic features on the same level as a word embeddings with the advantage that there are no out-of-vocabulary words, since each word can be created from several subword units.

5.4 Word-Level Embeddings (Kocmi and Bojar, 2017c)

We have concluded our work with word embeddings in the study Kocmi and Bojar (2017c) . We have compared several pretrained word embeddings as well as various random initializations and examined their influence on the performance of four various NLP tasks and two deep neural network architectures namely recurrent neural networks and convolutional neural networks. We have showed, that pretrained embeddings can help the neural network in a faster convergence to the best performance. On the other hand, without a further training the fixed pretrained embeddings never reach same performance as randomly initialized ones. This result could imply why pretrained document embeddings could fail and we would need to focus our further research on document embeddings trained with the NMT model.

5.5 Domain Adaptation (Bojar et al., 2016b; Kocmi et al., 2017)

We have examined domain adaptation in the work (Bojar et al., 2016b) and marginally in the (Kocmi et al., 2017). We have showed that domain adaptation quickly overfits to the new domain and loses the performance on the general domain. The issue of quick overfitting is a known problem of the neural networks. Our conclusion is that no more than one or two epochs over all in-domain data should be performed.

5.6 In-domain Backtranslation (Sudarikov et al., 2017)

As mentioned in Section 3, the size of in-domain data is crucial and usually there is only a small in-domain corpus. In the work (Sudarikov et al., 2017), we have extended the size of the training corpus by in-domain monolingual data with machine backtranslation and got a significant performance improvement.

5.7 Curriculum Learning (Kocmi and Bojar, 2017a; Bojar et al., 2017)

Closely related to the domain adaptation is curriculum learning. This is based on the concept that when humans are learning, they start with easier tasks from some close domain and gradually, as they gain experience and abstraction, they are able to learn to handle more and more complex situations. It has been shown by Bengio

[2] http://www.statmt.org/wmt17/

et al. (2009) that even neural networks can improve their performance when they are presented with the easier examples first. We have followed on this research in the works (Kocmi and Bojar, 2017a; Bojar et al., 2017). We have examined various linguistically-motivated domains and concluded that an improvement of up to 1 BLEU could be acquired with this method.

5.8 Unpublished Results

In this section we summarize our experiments with document embeddings on the task of machine translation. We have started with document embeddings only recently and it must be noted that at this stage, the results are preliminary, and experiments have been conducted only for some intuitively selected options.

We use Neural Monkey (Helcl and Libovický, 2017), an open-source neural machine translation and general sequence-to-sequence learning system built using the TensorFlow machine learning library. The neural architecture for the machine translation is based on the Bahdanau et al. (2014) paper, where we set the hyperparameters as follows:

The encoder uses word embeddings of size 400, with maximal length of 50 tokens. The hidden bidirectional GRU recurrent layer have size 600. The decoder has analogical settings only differing in the use of conditional GRU cells. As an optimization algorithm we are using Adam with learning rate 10^{-4} (Kingma and Ba, 2014). During evaluation we are using the beam search (Graves, 2012) with a beam size of 20 and length normalization 0.6.

We are preprocessing data to use byte-pair-encoding (Sennrich et al., 2016c) to overcome a problem with OOV words. We use 30 000 merges.

As the training, development and test set we use CzEng 1.6 as described in Section 5.1.

The following variants are compared with the baseline:

- Using a new starting token $\langle 2domain \rangle$ instead of the standard $\langle s \rangle$ token in baseline. An approach described in Section 3.2.

- Replacing starting token word embedding with document embedding. We compare both pretrained and specialized embeddings.

We experiment with two document embeddings from both pretrained and specialized category. As the pretrained ones we have selected doc2vec (Le and Mikolov, 2014) with embedding size 400. For the specialized embeddings we use a plain convolutional neural network with three layers of convolutions and ReLU nonlinearities. The number of features is 400, convolutional kernel width is 5 and the stride is 1. Both embeddings lead into vectors of the same size, namely 400. This allows us to exchange the word embedding of the starting symbol with the document embedding.

Table 5.8 shows improvements over a baseline with the use of document embeddings. The results are measured on the validation set of CzEng 1.6, which matches the distribution of the training set, and the news testset, which shows a drop in the performance on the validation set due to the different domain of the data.

It is notable from the results, that both types of embeddings lead to the improvement of the score. On the other hand we do not see any significant improvement with the use of the domain tag. We plan to make more thorough experiments of improvement seen with the doc2vec and send the results to the NAACL conference.

We showed only two possible ways how to combine document embeddings with the NMT. We want to begin our future work with exploring various combinations as described in Section 4.

6 Future Work Plan

In this section, we summarize our planned future work on document embeddings as a means of domain adaptation. We plan to solve them in the following order:

1. Run the experiment defined in Section 4 to figure out the best place where to include document embeddings into the neural network.

2. Compare various document embeddings as described in Section 2.1 and select the best approach. Focus on comparing pretrained embeddings with embeddings trained with the model because we hope to get similar results as with the word embeddings (Kocmi and Bojar, 2017a).

3. Prepare various development sets covering all types of domains as defined in Section 3.1. We are especially interested in the creation of a dataset with sentiment and writing style

Setup	Score (validation set)	Score (testset)
Baseline	38.59	20.14
Token ⟨*2domain*⟩	38.45	20.37
Starting embedding with doc2vec	40.73*	-
Starting embedding with CNN	38.59	20.55*

Table 1: Result comparing various modifications of NMT system with the document embeddings. We forget to precompute the doc2vec of the testset and therefore the improvement over validation data should be taken with caution.
* The results are significantly better over the baseline, tested with bootstrapping method with 1000 resamples and alpha level of 0.05

variation, since we already have a development set based on topics of documents.

4. Use the development set from previous point to compare which domains can benefit from document embeddings. We examine document embeddings on the task of document classification, where embeddings of documents from similar domain should already be closer to each other.

5. Compare our approach with the classical domain adaptation. We are most interested to learn if our approach can maintain its performance in the general domain and reach the same performance as different models adapted to various domains.

7 Summary

Document-level machine translation is a complex open problem that can have a significant impact on the quality of translation.

In this thesis proposal we have presented an overview over document embeddings and domain adaptation techniques used in neural NLP. We propose a novel usage of document embeddings as a means of domain adaptation.

The main expected contribution of this work is a thorough examination of various document embeddings and their effect on the learning process of neural networks. Furthermore the creation of corpora for various domains in machine translation will be an invaluable resource for further research.

Bibliography

Ondřej Bojar, Ondřej Dušek, Tom Kocmi, Jindřich Libovickỳ, Michal Novák, Martin Popel, Roman Sudarikov, and Dušan Variš. 2016a. CzEng 1.6: Enlarged Czech-English Parallel Corpus with Processing Tools Dockered. In *International Conference on Text, Speech, and Dialogue*, pages 231–238. Springer.

Ondřej Bojar, Jindřich Helcl, Tom Kocmi, Jindřich Libovický, and Tomáš Musil. 2017. Results of the WMT17 Neural MT Training Task. In *Proceedings of the 2nd Conference on Machine Translation (WMT)*, Copenhagen, Denmark.

Ondrej Bojar, Roman Sudarikov, Tom Kocmi, Jindrich Helcl, and Ondrej Cıfka. 2016b. UFAL Submissions to the IWSLT 2016 MT Track. *IWSLT. Seattle, WA*.

Tom Kocmi and Ondřej Bojar. 2016. SubGram: Extending Skip-Gram Word Representation with Substrings. In *International Conference on Text, Speech, and Dialogue*, pages 182–189. Springer.

Tom Kocmi and Ondřej Bojar. 2017a. Curriculum Learning and Minibatch Bucketing in Neural Machine Translation. In *Recent Advances in Natural Language Processing 2017*.

Tom Kocmi and Ondřej Bojar. 2017b. LanideNN: Multilingual Language Identification on Character Window. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 1, pages 927–936. Association for Computational Linguistics.

Tom Kocmi and Ondřej Bojar. 2017c. An Exploration of Word Embedding Initialization in Deep-Learning Tasks. In *Sumbited to ICON 2017*.

Tom Kocmi, Dušan Variš, and Ondřej Bojar. 2017. CUNI NMT System for WAT 2017 Translation Tasks. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2017)*.

Roman Sudarikov, David Mareček, Tom Kocmi, Dušan Variš, and Ondřej Bojar. 2017. CUNI Submission in WMT17: Chimera Goes Neural. In *Proceedings of the 2nd Conference on Machine Translation (WMT)*, Copenhagen, Denmark.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *ICLR 2015*.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL (1)*, pages 238–247.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Boxing Chen, Colin Cherry, George Foster, and Samuel Larkin. 2017. Cost weighting for neural machine translation domain adaptation. *ACL 2017*, page 40.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 484–494, Berlin, Germany. Association for Computational Linguistics.

Noam Chomsky. 1964. Aspects of the theory of syntax. Technical report, MASSACHUSETTS INST OF TECH CAMBRIDGE RESEARCH LAB OF ELECTRONICS.

Chenhui Chu, Raj Dabre, and Sadao Kurohashi. 2017. An empirical comparison of simple domain adaptation methods for neural machine translation. *arXiv preprint arXiv:1701.03214*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Andrew M Dai, Christopher Olah, and Quoc V Le. 2015. Document embedding with paragraph vectors. *arXiv preprint arXiv:1507.07998*.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391.

Nemanja Djuric, Hao Wu, Vladan Radosavljevic, Mihajlo Grbovic, and Narayan Bhamidipati. 2015. Hierarchical neural language models for joint representation of streaming documents and their content. In *Proceedings of the 24th International Conference on World Wide Web*, pages 248–255. International World Wide Web Conferences Steering Committee.

Steve Engels, Vivek Lakshmanan, and Michelle Craig. 2007. Plagiarism detection using feature-based neural networks. *SIGCSE Bull.*, 39(1):34–38.

Markus Freitag and Yaser Al-Onaizan. 2016. Fast domain adaptation for neural machine translation. *arXiv preprint arXiv:1612.06897*.

Jianfeng Gao and Min Zhang. 2002. Improving language model size reduction using better pruning criteria. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 176–182. Association for Computational Linguistics.

Eva Martínez Garcia, Carles Creus, Cristina España-Bonet, and Lluís Màrquez. 2017. Using word embeddings to enforce document-level lexical consistency in machine translation. *The Prague Bulletin of Mathematical Linguistics*, 108(1):85–96.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122*.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 513–520.

Alex Graves. 2012. Sequence transduction with recurrent neural networks. *Proceedings of the 29th International Conference on Machine Learning (ICML 2012)*.

Maryam Habibi and Andrei Popescu-Belis. 2015. Keyword extraction and clustering for document recommendation in conversations. *IEEE/ACM Transactions on audio, speech, and language processing*, 23(4):746–759.

Jindřich Helcl and Jindřich Libovický. 2017. Neural monkey: An open-source tool for sequence learning. *The Prague Bulletin of Mathematical Linguistics*, 107:5–17.

Almut Silja Hildebrand, Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Adaptation of the translation model for statistical machine translation based on information retrieval. In *Proceedings of EAMT*, volume 2005, pages 133–142.

Serena Jeblee, Weston Feely, Houda Bouamor, Alon Lavie, Nizar Habash, and Kemal Oflazer. 2014. Domain and dialect adaptation for machine translation into egyptian arabic. *ANLP 2014*, page 196.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. *arXiv preprint arXiv:1506.06726*.

Catherine Kobus, Josep Crego, and Jean Senellart. 2016. Domain control for neural machine translation. *arXiv preprint arXiv:1612.06140*.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Jey Han Lau and Timothy Baldwin. 2016. An empirical evaluation of doc2vec with practical insights into document embedding generation. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 78–86, Berlin, Germany. Association for Computational Linguistics.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2016. Fully character-level neural machine translation without explicit segmentation. *CoRR*, abs/1610.03017.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. *arXiv preprint arXiv:1707.07045*.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the International Workshop on Spoken Language Translation*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Saif M Mohammad, Mohammad Salameh, and Svetlana Kiritchenko. 2016. How translation alters sentiment. *J. Artif. Intell. Res.(JAIR)*, 55:95–130.

Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. 2017. Personalizing session-based recommendations with hierarchical recurrent neural networks. *arXiv preprint arXiv:1706.04148*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Controlling politeness in neural machine translation via side constraints. In *HLT-NAACL*, pages 35–40.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016c. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Christophe Servan, Josep Crego, and Jean Senellart. 2016. Domain specialization: a post-training domain adaptation for neural machine translation. *arXiv preprint arXiv:1612.06141*.

Petr Sgall, Eva Hajicová, and Jarmila Panevová. 1986. *The meaning of the sentence in its semantic and pragmatic aspects*. Springer Science & Business Media.

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, Christopher Potts, et al. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, volume 1631, page 1642.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Document modeling with gated recurrent neural network for sentiment classification. In *EMNLP*, pages 1422–1432.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2015. Towards universal paraphrastic sentence embeddings. *CoRR*, abs/1511.08198.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J Smola, and Eduard H Hovy. 2016. Hierarchical attention networks for document classification. In *HLT-NAACL*, pages 1480–1489.

Zhaocheng Zhu and Junfeng Hu. 2017. Context aware document embedding. *arXiv preprint arXiv:1707.01521*.