WHO: International Classification of Diseases ICD11 English to French Translation

AGENCE DU NUMÉRIQUE EN SANTÉ

Konstantinos Skianis
Laboratoire d'Informatique (LIX), École Polytechnique, France
skianis.konstantinos@gmail.com

May 9, 2023

Abstract

Medical terminologies resources and standards play vital roles in clinical data exchanges, enabling significantly the services' interoperability within health national information networks. Health and medical science are constantly evolving causing requirements to advance the terminologies editions. WHO International Classification of Diseases (ICD) terminology is the diagnostic classification standard for all clinical and research purposes. WHO is entrusted to revise the ICD terminology editions in English. The WHO member states take the tasks to translate the new ICD versions in their local languages. In this deliverable, we present our focused work on translating medical terminologies with machine translation techniques. The devised procedure is tested on ICD-11 translation from English to French. Experiments have been conducted with use of statistical (SMT) and neural machine translation (NMT) methods. A live web tool that enables searching the proposed ICD-11 translations is online, via this link: `http://anstranslation.ddns.net:5000`.

Contents

1 Introduction

The use of medical terminology is of critical importance for the community to be able to store, organize and process all medical-related data that are generated in labs, hospitals, institutions and other entities. These terms are arranged systematically in dictionaries and lexicons, that follow specific structures and rules. In order to facilitate hierarchies and connections, the terms are represented by ontologies, enabling us to keep additional information (e.g. family of disease).

Medical terminologies are a necessary resource to any kind of health care information task (e.g. coding, free text indexing, information retrieval). These terminologies can be standardized llike ICD or SNOMED CT. The standardization process is required for the modern medical and health systems, not only in a national but in an international level as well. Adopting a specific dictionary or terminology can be a controversial topic for public organizations like governments and private institutions like hospitals.

The International Classification of Diseases (ICD) is the international standard diagnostic tool for epidemiology, health management and clinical purposes, consisting one of the most used medical dictionaries across many organizations and countries. The ICD dictionary is maintained by the World Health Organization (WHO), and its last version ICD-11 is a major step forward, because it has the necessary terminological and ontological elements for seamless use in digital health.

As the initial medical lexicons which contain these ontologies are created in English, there is an evident need for translation in other languages. This translation process can be expensive both in terms of time and resources, while the vocabulary and number of medical terms can reach high numbers. The common policy is to rely on the work of translators. However this manual approach is time-consuming and requires skilled specialized translators.

The project of ICD-11 translation from English to French will constitute a fast and open methodology for medical terminology management. During this project, we will attempt to develop a first baseline machine translation (bootstrap) for the ICD-11 dictionary, based on automated machine translation approaches. Machine translation refers to the automatic translation of text or speech in one specific source language into another target language. The main goal of machine translation is to translate a given string or sentence in the source text into a string in the target language.

This work constitutes a generic, language-independent and open methodology for medical terminology management. To illustrate this methodology, which is based on automated machine translation approaches, we will attempt to develop a first baseline translation for the ICD-11 dictionary.

First, in Section 2, we are going to investigate existing machine translation studies and papers concerning medical terms and documents, with a comparison of the relative methods shown. Next, in Section 4, we introduce the datasets used as input in the training process are Section 5 presents the proposed methodology. The statistical and neural machine translation methods are shown in Sections 6 and 7 respectively. In Section 8 we present the final approach with setup and results. Last, we conclude with comments and recommendations for future work.

2 Related work

In order to achieve that, we will focus on the study of popular methods from statistical machine translation as well as the current state-of-the-art approaches in neural machine translation. To the best of our knowledge, this work will be one of the first that provides a broad and comprehensive study of state-of-the-art machine translation approaches for medical terminology translation.

In this deliverable, we describe state-of-the-art approaches for machine translation, with regards as well to medical terminology and not only. First, we will present popular approaches in statistical machine translation as well as current state-of-the-art methods in the area of neural machine translation. Afterwards, we are going to investigate existing machine translation studies and papers concerning medical terms and documents, with a comparison of the relative methods shown. Next, an abstractive architecture of our proposed methodology is presented. Last, we conclude with remarks.

2.1 Statistical machine translation

Statistical machine translation (SMT) is a machine translation paradigm where translations are generated on the basis of statistical models whose parameters are derived from the analysis of bilingual text corpora.

The idea behind statistical translation is to compute the probability that a sentence in the source language, s, be translated into a sentence in the target language, t. It is the task of a decoder to find the most probable translation. This probability is formulated through Bayes' theorem as follows:

$$\text{arg max}_t p(t|s) = \text{arg max}_t p(s|t)p(t) \tag{1}$$

where $p(t)$ can be treated as a language model and $p(s|t)$ as a translation model as explained below [Koehn et al., 2003]. The expression portrays that the most probable translation for one sentence equals the probability that that same translation would be translated into that one sentence and the probability that the probable translation actually is a sentence in the target language.

The statistical approach contrasts with the rule-based approaches to machine translation as well as with example-based machine translation.

Phrase-based statistical machine translation (PB-SMT) The dominant paradigm in statistical machine translation today is clearly phrase-based statistical machine translation [Marcu and Wong, 2002, Koehn et al., 2003]. Essentially a joint probability model is built, which automatically learns word and phrase equivalents from bilingual corpora. More specifically, Marcu and Wong [2002] describe a translation model that assumes that lexical correspondences can be established not only at the word level, but at the phrase level as well. In contrast with many previous approaches, their model does not try to capture how **Source** sentences can be mapped into **Target** sentences, but rather how **Source** and **Target** sentences can be generated simultaneously. In other words, a joint probability model is estimated that can be easily marginalized in order to yield conditional probability models for both source-to-target and target-to-source machine translation applications. The main difference between this work and previous state-of-the-art methods is that joint probability models of translation equivalence are learned not only between words but also between phrases and we show that these models can be used not only for the extraction of bilingual lexicons but also for the automatic translation of unseen sentences.

Training The training algorithm for the phrase-based joint probability model can be summarized in the following steps:

1. Determine high-frequency ngrams in the bilingual corpus.
2. Initialize the t-distribution table.
3. Apply EM training on the Viterbi alignments, while using smoothing.
4. Generate conditional model probabilities.

Limitations The main shortcoming of the phrase-based model in Marcu and Wong [2002] concerns the size of the t-table and the cost of the training procedure we currently apply. To keep the memory requirements manageable, the system is arbitrarily restricted to learning phrase translations of at most six words on each side. Also the swap, break, and merge operations used during the Viterbi training are computationally expensive.

While statistical machine translation research has gained much by building on the insight that probabilities may be used to make informed choices, current models are deficient because they lack crucial information. Much of the translation process is best explained with morphological, syntactic, semantic, or other information that is not typically contained in parallel corpora. In their work [Koehn et al., 2006, Koehn and Hoang, 2007], the authors show that when such information is incorporated to the training data, we can build richer models of translation, which we call factored translation models. Since the data are automatically tagged, there are often many ways of marking up input sentences. This further increases the multitude of choices that the machine translation system must deal with, and requires an efficient method for dealing with potentially ambiguous input. Finally, they investigated confusion network decoding as a way of addressing this challenge.

2.2 Neural machine translation

Neural machine translation (NMT) is an approach to machine translation that uses a large artificial neural network to predict the likelihood of a sequence of words, typically modeling entire sentences in a single integrated model. NMT departs from phrase-based statistical approaches that use separately engineered subcomponents [Wołk and Marasek, 2015]. Neural machine translation (NMT) is not a drastic step beyond what has been traditionally done in statistical machine translation (SMT). Its main departure is the use of vector representations ("embeddings", "continuous space representations") for words and internal states. The structure of the models is simpler than phrase-based models. There is no separate language model, translation model, and reordering model, but just a single sequence model that predicts one word at a time. However, this sequence prediction is conditioned on the entire source sentence and the entire already produced target sequence. NMT models use deep learning and representation learning.

The word sequence modeling was at first typically done using a recurrent neural network (RNN). A bidirectional recurrent neural network, known as an encoder, is used by the neural network to encode a source sentence for a second RNN, known as a decoder, that is used to predict words in the target language [Bahdanau et al., 2015].

Convolutional Neural Networks (Convnets or CNNs) are in principle somewhat better for long continuous sequences, but were initially not used due to several weaknesses that were successfully compensated for by 2017 by using so-called "attention"-based approaches [Bahdanau et al., 2015]. There are further Coverage Models addressing the issues in traditional attention mechanism, such as ignoring of past alignment information leading to over-translation and under-translation.

Despite their flexibility and power, DNNs and more specifically CNNs can only be applied to problems whose inputs and targets can be sensibly encoded with vectors of fixed dimensionality. It is a significant limitation, since many important problems are best expressed with sequences whose lengths are not known a-priori. For example, speech recognition and machine translation are sequential problems.

In Sutskever et al. [2014], the authors presented a general end-to-end approach to sequence learning that makes minimal assumptions on the sequence structure. Our method uses a multilayered Long Short-Term Memory (LSTM) to map the input sequence to a vector of a fixed dimensionality, and then another deep LSTM to decode the target sequence from the vector. Our main result is that on an English to French translation task from the WMT-14 dataset, the translations produced by the LSTM achieve a BLEU score of 34.8 on the entire test set, where the LSTM's BLEU score was penalized on out-of-vocabulary words. Additionally, the LSTM did not have difficulty on long sentences. For comparison, a phrase-based SMT system achieves a BLEU score of 33.3 on the same dataset. An illustration of the seq2seq architecture is presented in Figure 1.

A neural machine translation (seq2seq) tutorial is shown here[1]. Another great visualization of neural machine translation techniques can be found here[2].

Having gone through the seq2seq model [Sutskever et al., 2014], we will proceed with more advanced techniques. To build state-of-the-art neural machine translation systems, we will need the attention mechanism, which was first introduced by Bahdanau et al. [2015], then later refined by Luong et al.

[1] https://github.com/tensorflow/nmt

[2] jalammar.github.io/visualizing-neural-machine-translation-mechanics-of-seq2seq-models-with-attention/

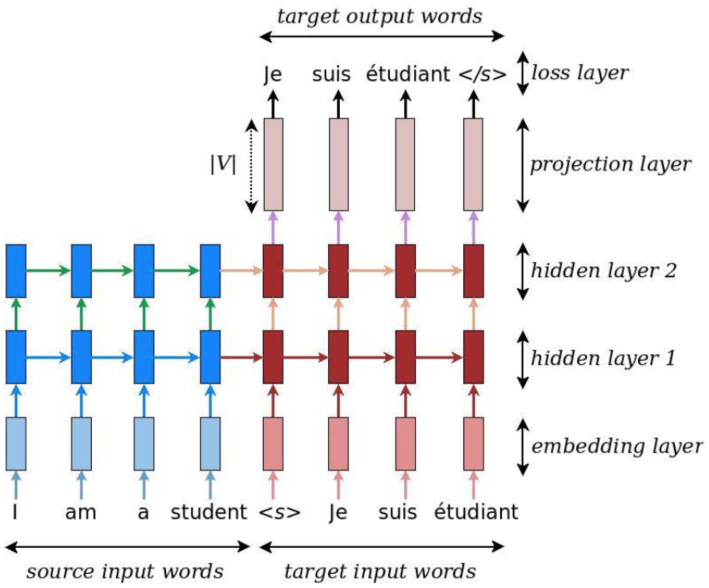


Figure 1: Example of the seq2seq technique.

[2015] and others. The key idea of the attention mechanism is to establish direct short-cut connections between the target and the source by paying "attention" to relevant source content as we translate.

With the adoption of deep learning techniques in the machine translation field, differently from the phrase-based paradigm, neural machine translation (NMT) operates on word and sentence representations in a continuous space. The need to enforce fixed translations of certain source words is a well known problem in machine translation (MT). For instance, this is an issue in application scenarios in which the translation process has to comply with specific terminology and/or style guides. In such situations it is generally necessary to consider external resources to guide the decoder in order to ensure consistency or meet other specific requirements. Chatterjee et al. [2017] propose a “guide" mechanism that enhances an existing NMT decoder with the ability to prioritize and adequately handle translation options presented in the form of XML annotations of source words.

Semi-supervised neural machine translation with language models [Cheng et al., 2016, Skorokhodov et al., 2018]concatenation of labeled (parallel corpora) and unlabeled (monolingual corpora) data and reconstruct the monolingual corpora using an autoencoder, in which the source-to-target and target-to-source translation models serve as the encoder and decoder, respectively. As shown in Figure 2, in a source autoencoder, the source-to-target model $P(y|x; \overrightarrow{\theta})$ serves as an encoder to transform the observed source sentence x into a latent target sentence y (highlighted in grey), from which the target-to-source model $P(x'|y; \overleftarrow{\theta})$ reconstructs a copy of the observed source sentence x' from the latent target sentence. As a result, monolingual corpora can be combined with parallel corpora to train bidirectional NMT models in a semi-supervised setting.

Unsupervised neural machine translation (Artetxe et al. [2018], Lample et al. [2018]) trains an NMT system in a completely unsupervised manner, relying on nothing but monolingual corpora. It builds upon unsupervised embedding mappings, and consists of a slightly modified attentional encoder-decoder model.

Peters et al. [2018] introduced a new type of deep contextualized word representation that models both (1) complex characteristics of word use (e.g., syntax and semantics), and (2) how these uses vary across linguistic contexts (i.e., to model polysemy). The word vectors are learned functions of the internal states of a deep bidirectional language model (biLM), which is pretrained on a large text corpus.

Later, Devlin et al. [2019] introduced a new language representation model called BERT, which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models [Peters et al., 2018], BERT is designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. As a result, the

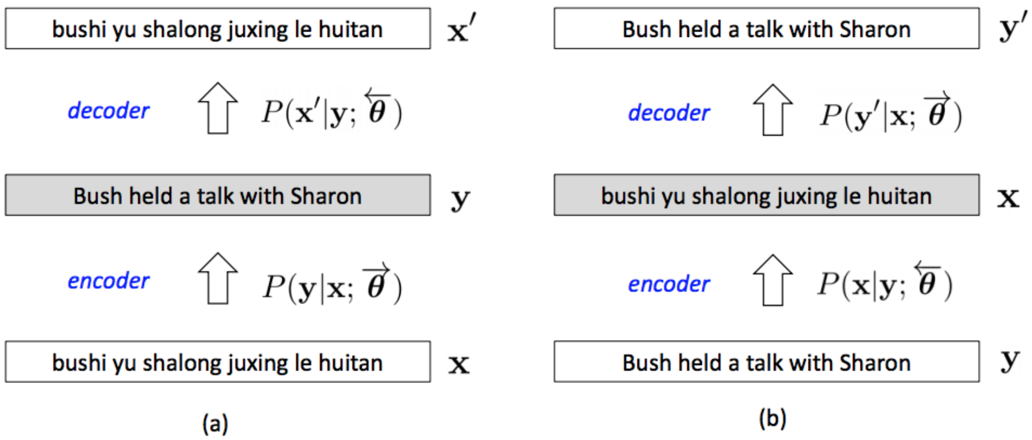

bushi yu shalong juxing le huitan
x'
decoder
P(x'|y; ←θ)
Bush held a talk with Sharon
y
encoder
P(y|x; →θ)
bushi yu shalong juxing le huitan
x
(a)
Bush held a talk with Sharon
y'
decoder
P(y'|x; →θ)
bushi yu shalong juxing le huitan
x
encoder
P(x|y; ←θ)
Bush held a talk with Sharon
y
(b)

Figure 2: Examples of (a) source autoencoder and (b) target autoencoder on monolingual corpora by Cheng et al. [2016]. The main idea is to leverage autoencoders to exploit monolingual corpora for NMT.

pre-trained BERT model can be finetuned with just one additional output layer to create state-of-the-art models for a wide range of NLP tasks.

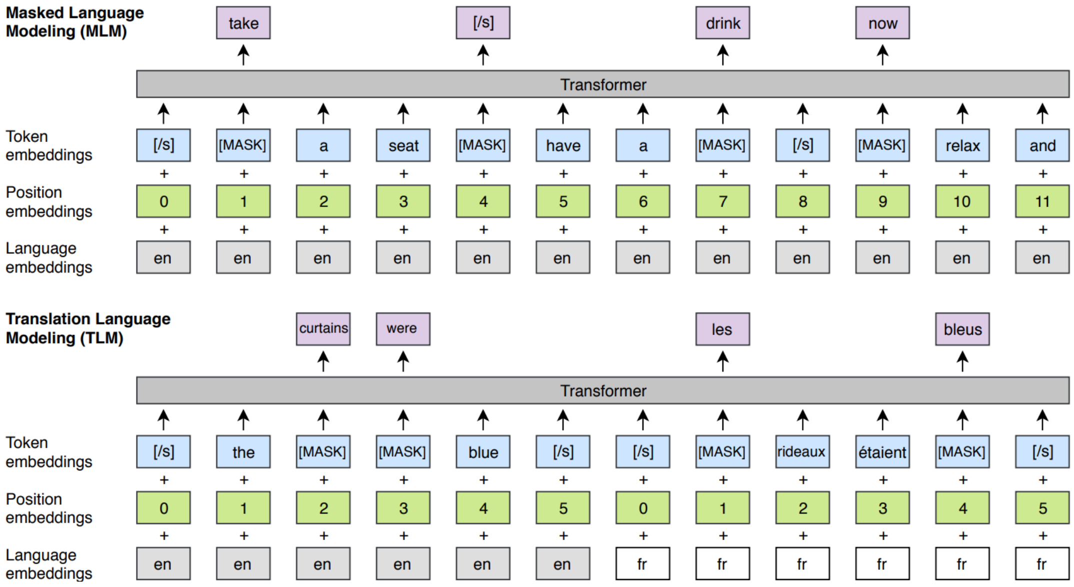

Masked Language Modeling (MLM)
take
[/s]
drink
now
Transformer
Token embeddings
[/s] [MASK] a seat [MASK] have a [MASK] [/s] [MASK] relax and
Position embeddings
0 1 2 3 4 5 6 7 8 9 10 11
Language embeddings
en en en en en en en en en en en en
Translation Language Modeling (TLM)
curtains
were
les
bleus
Transformer
Token embeddings
[/s] the [MASK] [MASK] blue [/s] [/s] [MASK] rideaux étaient [MASK] [/s]
Position embeddings
0 1 2 3 4 5 0 1 2 3 4 5
Language embeddings
en en en en en en fr fr fr fr fr fr

Figure 3: Cross-lingual language model pretraining. The MLM objective is similar to the one of Devlin et al. [2019], but with continuous streams of text as opposed to sentence pairs. The TLM objective extends MLM to pairs of parallel sentences. To predict a masked English word, the model can attend to both the English sentence and its French translation, and is encouraged to align English and French representations. Position embeddings of the target sentence are reset to facilitate the alignment.

Last, Lample and Conneau [2019] extended generative pretraining for English natural language understanding to multiple languages and show the effectiveness of cross-lingual pretraining. They proposed two methods to learn cross-lingual language models (XLMs): one unsupervised that only relies on monolingual data, and one supervised that leverages parallel data with a new cross-lingual language model objective. The method obtained state-of-the-art results on cross-lingual classification, unsupervised and supervised machine translation. An abstractive illustration of the architecture is shown in Figure 3.

2.3 Machine translation for medical terms

Pease and Boushaba [1996] developed a method for automatic translation of medical terminology and texts in the Arabic language. The authors explain in detail how a language like Arabic can be relatively easily included in a machine translation system which was originally developed for European languages. They investigated how general linguistic and terminological knowledge is integrated into one framework of analysis which, using a few principles of syntactic and semantic composition, can translate both general language and medical texts. The Aramed project, financed by the INCO programme of the European Commission, is developing a system which translates medical classifications (based on the SNOMED medical codes) from English to Arabic, and German and English medical texts into Arabic. Its aim is to provide a usable tool for automatic translation in the area of medicine in the Arab world. It also provides the basis of a German-Arabic machine translation system. The system consists of two main components: an Arabic morphological generator (NALG) and a transfer, constraint and unification-based machine translation system (CAT2), developed as a sideline to the Eurotra machine translation project.

Texts from the medical domain are a very important resource for natural language processing and could be critical for a machine translation task. Eck et al. [2004] investigate the usefulness of a large medical database (the Unified Medical Language System) for the translation of dialogues between doctors and patients using a statistical machine translation system. They showed that the extraction of a large dictionary and the usage of semantic type information to generalize the training data significantly improves the translation performance.

Claveau and Zweigenbaum [2005] presents a method to automatically translate a large class of terms in the biomedical domain from one language to another; it is evaluated on translations between French and English. It relies on a machine-learning technique that infers transducers from examples of bilingual word pairs; no additional resource or knowledge is needed. Then, these transducers, making the most of the high regularity of translation discovered in the examples, can be used to translate unseen French terms into English or vice versa. More specifically, their approach relies on two major hypotheses: (i) a large class of French and English terms are morphologically related; (ii) differences between French and English terms are regular enough to be automatically learned. These two hypotheses make the most of the fact that biomedical terms often share a common Greek or Latin basis, and that their morphological derivations are very regular (e.g. ophtalmorragie/ophthalmorrhagia, leucorragie/leukorrhagia...). Their technique relies on a supervised machine-learning algorithm, called OSTIA [Oncina, 1991], that infers transducers (cf. next section) from examples of bilingual termpairs. Such transducers, when given a new term in English (respectively French), must propose the corresponding French (resp. English) term.

2.3.1 Alignment of medical lexicons

Bitext word alignment or simply word alignment is the natural language processing task of identifying translation relationships among the words (or more rarely multiword units) in a bitext, resulting in a bipartite graph between the two sides of the bitext, with an arc between two words if and only if they are translations of one another. Word alignment is typically done after sentence alignment has already identified pairs of sentences that are translations of one another. An example of word aligmnent is shown in Figure 4.

Word alignment is an important supporting task for most methods of statistical machine translation. The parameters of statistical machine translation models are typically estimated by observing word-aligned bitexts, and conversely automatic word alignment is typically done by choosing that alignment which best fits a statistical machine translation model. Circular application of these two ideas results in an instance of the expectation-maximization algorithm.

Nyström et al. [2006] reports on a parallel collection of rubrics from the medical terminology systems ICD-10, ICF, MeSH, NCSP and KSH97-P and its use for semi-automatic creation of an English-Swedish dictionary of medical terminology. The methods presented are relevant for many other West European language pairs than English-Swedish. The medical terminology systems were collected in electronic format in both English and Swedish and the rubrics were extracted in parallel language pairs.

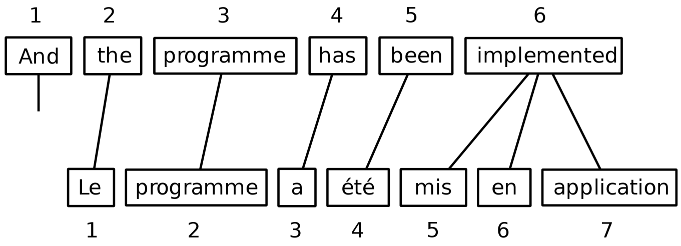


Figure 4: An example of word alignment, taken by WikiPedia.

Initially, interactive word alignment was used to create training data from a sample. Then the training data were utilised in automatic word alignment in order to generate candidate term pairs. The last step was manual verification of the term pair candidates. A dictionary of 31,000 verified entries has been created in less than three man weeks, thus with considerably less time and effort needed compared to a manual approach, and without compromising quality. As a side effect of their work they found 40 different translation problems in the terminology systems and these results indicate the power of the method for finding inconsistencies in terminology translations. They also report on some factors that may contribute to making the process of dictionary creation with similar tools even more expedient.

Markó et al. [2006] presented an approach for the creation of a multilingual medical dictionary for the biomedical domain. In a first step, available monolingual lexical resources are compiled into a common interchange format. Secondly, according to a linking format defined by the authors, the cross-lingual mappings of lexical entries are added. They showed how these mappings can be generated using a morpho-semantic term normalization engine, which captures intra as well as interlingual synonymy relationships on the level of subwords. Using the algorithm introduced, they obtained a list of 300,894 bi-directional relations between lexemes, out of which 16,123 translations have been generated for English-French.

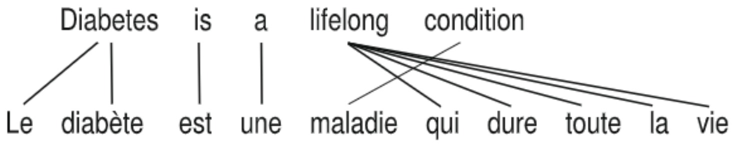


Figure 5: Word alignment for medical documents.

Deléger et al. [2009] presented a methodology which aims to ease this process by automatically acquiring new translations of medical terms based on word alignment in parallel text corpora, and test it on English and French. After collecting a parallel, English-French corpus, we detected French translations of English terms from three terminologies-MeSH, SNOMED CT and the MedlinePlus Health Topics. They obtained respectively for each terminology 74.8%, 77.8% and 76.3% of linguistically correct new translations. A sample of the MeSH translations was submitted to expert review and 61.5% were deemed desirable additions to the French MeSH. In conclusion, they successfully obtained good quality new translations, which underlines the suitability of using alignment in text corpora to help translating terminologies. Their method may be applied to different European languages and provides a methodological framework that may be used with different processing tools.

2.3.2 Projection-based variants

Identifying translations of terms in comparable corpora is a challenge that has attracted many researchers. A popular idea that emerged for solving this problem is based on the assumption that the context of a term and its translation share similarities that can be used to rank translation candidates [Fung, 1998, Rapp, 1999]. Many variants of this idea have been implemented.

While a few studies have investigated pattern matching approaches to compare source and target contexts [Diab and Finch, 2000, Yu and Tsujii, 2009], most variants make use of a bilingual lexicon

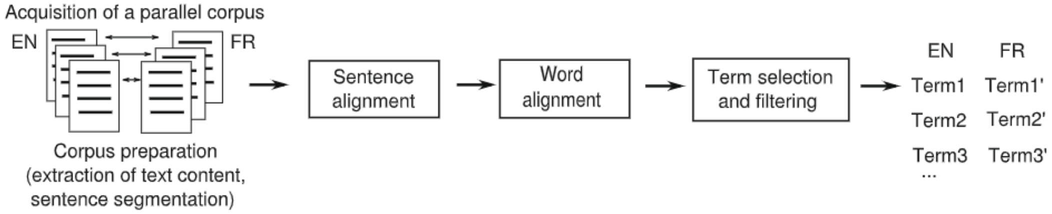


Figure 6: Description of the methodology followed by Deléger et al. [2009].

in order to translate the words of the context of a term (often called seed words). Déjean et al. [2005] instead use a bilingual thesaurus for translating these by illustrating their use in multi-language information retrieval (English/German) in the medical domains.

Laroche and Langlais [2010] have systematically investigated the impact of the many parameters controlling their approach, like context, association measure, similarity measure and seed lexicon. As a test case, they addressed the task of translating terms of the medical domain by exploiting pages mined from Wikipedia. Instead, they studied the impact of some major factors influencing projection-based approaches on a task of translating 5,000 terms of the medical domain (the most studied domain), making use of French and English Wikipedia pages extracted monolingually thanks to an information retrieval engine.

2.3.3 Knowledge-based approaches

Medical language is highly compositional and makes extensive use of common roots, especially Latin-Greek roots. Besides words devoted to common sense, medical language presents some typical characteristics, especially on morphological and semantic aspects of word formation. Morphological decomposition and identification precedes semantic analysis. It is only when these two prerequisites are fulfilled that an attempt to grasp the meaning of a whole expression is made possible.

Lovis et al. [1998] proposed an approach to cover 'the lack of coverage of the medical lexical knowledge', in order to help physicians find the correct international classification for diseases (ICD) codes for a written diagnosis. The methodology allows the development of a powerful dynamic dictionary dedicated to natural language processing in the field of diagnoses and narrative procedures. It describes the design of an analyser that can profit from a dictionary. The methods used have proved to be efficient for various classifications, as well as for multiple languages, as the system presently supports French, German, English and Dutch for ICD-9 and ICD-10 classifications.

Later, Deléger et al. [2010] wanted to translate the MedlinePlus terminology by following a twofold strategy: initially a knowledge-based approach, and a corpus-based NLP method. The knowledge-based approach implies that each term to be translated must be included into the Metathesaurus. They used four French terminologies: MeSH, SNOMED International (SNMI), MedDRA 17 and WHO-ART. The principle of the method is based on the conceptual construction of the UMLS metathesaurus. For each English MEDLINEPlus term included in the UMLS we extract its Concept Unique Identifiers. The next step of this method collects for each UMLS concept all French terms cor- responding to the given concept, that is all French terms possessing the same Concept Unique Identifier (CUI). Then, the corpus-based approach [Deléger et al., 2009] relies on alignment. Here, alignment is performed in a parallel English/French corpus at the sentence and word levels, and medical terms are selected from the results of this process.

Lindgren [2011] explored whether the contents of SNOMED CT can be translated in a satisfactory way through semi-automatic methods using already existing machine translation systems, semi-automatic indicating that a level of human input is necessary to perform all the steps in the translation process. Terms will be translated in a translation memory-based machine translation system and a statistical machine translation system. The translations are then compared with human translations via BLEU - a commonly used evaluation metric in corpus linguistics. The starting position is that an automatic translation would require fewer resources, in working hours as well as in money, than a translation performed manually. Different subsets of terms could provide different possibilities and difficulties

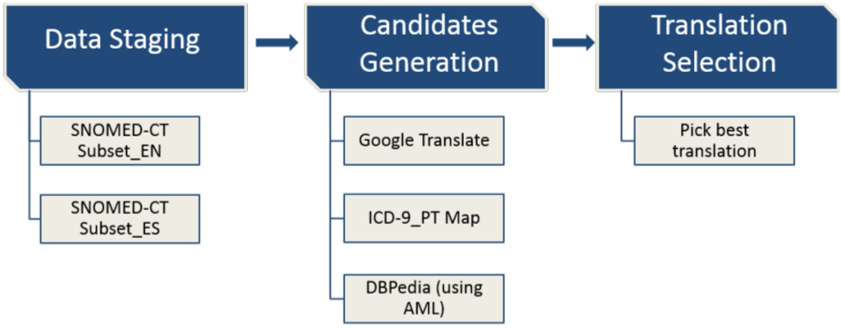


Figure 7: The translation pipeline of SNOMED CT. The best translation candidate is selected using an ensemble model trained that chooses the best method for each class of SNOMED CT terms, based on known translations.

depending on their contents; thus an analysis of the structure of the subsets of SNOMED CT used in this study is prepared.

Schulz et al. [2013] compared translations of SNOMED CT fully specified names which are provided by professional medical translators, the Web-based general scope translation software Google Translate, and by lay translators (medical students). The results demonstrate that low-cost translation solutions of medical terms may produce surprisingly good results.

2.3.4 Domain adaptation

Many works on domain adaptation examine the usage of available in-domain data to directly improve in-domain performance of SMT. Some authors attempt to combine the predictions of two separate (in-domain and general-domain) translation models [Bisazza et al., 2011] or language models [Koehn and Schroeder, 2007]. Wu and Wang [2004] use in-domain data to improve word alignment in the training phase.

Later, Dušek et al. [2014] developed systems within the Khresmoi project, a large integrated project aiming to deliver a multi-lingual multi-modal search and access system for biomedical information and documents. They are based on the Moses phrase-based translation toolkit and standard methods for domain adaptation, where two systems were created: one constrained and one unconstrained system for each subtask and translation direction. The constrained and unconstrained systems differ in training data only: the former use all allowed training data, the latter take advantage of additional web-crawled data.

Research by Wołk and Marasek [2015] examines the effects of different training methods on a Polish-English machine translation system used for medical data. The goal of this paper is to present experiments on neural based machine translation in comparison to statistical machine translation.

A Biomedical Translation task was first run at the First Conference on Machine Translation (WMT16) [Bojar et al., 2016]. This task aims to evaluate systems for the translation of biomedical titles and abstracts from scientific publications. In this first edition of the challenge, three language pairs were tested (considering both translation directions), namely, English-Portuguese (En-Por), English-Spanish (En-Sp) and English-French (En-Fr), and documents in the two sub-domains of biological sciences and health sciences.

Villegas et al. [2018] presented a Resource for English-Spanish Medical Machine Translation and Terminologies that is unique in the sense of integrating heterogeneous types of resources for medical machine translation, and harmonizing all the medical literature resources to a common standardized format.

2.3.5 Ontologies

A major stumbling block for existing NLP applications is automatic sense disambiguation. An automatic system can detect with high reliability that a given occurrence of a word like feel or dead is a verb or adjective. But it cannot easily determine which of a variety of alternative meanings such polysemous words have in any given context.

Smith and Fellbaum [2004] described an attempt to create a new lexical database called Medical WordNet (MWN), consisting of medically relevant terms used by and intelligible to non-expert subjects and supplemented by a corpus of natural-language sentences that is designed to provide medically validated contexts for MWN terms. Their resource contributes towards a solution of the aforementioned automatic word sense disambiguation problem. The corpus derives primarily from online health information sources targeted to consumers, and involves two sub-corpora, called Medical FactNet (MFN) and Medical BeliefNet (MBN), respectively.

A comprehensive multilingual class hierarchy of medical terms used in clinical records is included in the SNOMED CT system. Few translations are available, but, as new concepts and revisions are continuously being added, the manual translation and revision of the terms will remain a major endeavour. Silva et al. [2015] proposed a new approach for translating SNOMED CT terms (or named entities) using ontology mapping methods and various existing multilingual resources with translated concepts. Their purpose is generating initial candidate translations, already close to those proposed by medical experts, to be later used in a curated translation process. The method for automatically translating SNOMED CT is being developed for Portuguese, using DBPedia, ICD-9 and Google Translate as sources of candidate translations of the clinical terms, which could later be verified. Initial results, using a manually translated Portuguese catalog of allergies and adverse reactions (CPARA) to SNOMED CT as ground truth, show that it has high potential. Their approach for generating the translations of SNOMED CT terms into Portuguese is illustrated in Figure 7.

Most of the labels stored in semantically structured resources, like ontologies, taxonomies or knowledge graphs, are represented in English only. To enable knowledge access across languages, these resources need to be enriched with multilingual information.

Translating ontologies comes also with the challenge to disambiguate an ontology label with respect to the domain modelled by ontology itself. Machine translation services may help in this task; however, a crucial requirement is to have translations validated by experts before the ontologies are deployed. For this reason, Arcan et al. [2016] presented ESSOT, a collaborative knowledge management platform with a domain-aware SMT system for supporting language experts in the task of translating ontologies.

Later, Arcan and Buitelaar [2017] presented a performance comparison between SMT and NMT on translating highly domain-specific expressions, i.e. terminological expressions, documented in the ICD ontology in the medical domain. They showed that domain adaptation with only terminological expressions significantly improves the translation quality, which is specifically evident if an existing generic neural network is retrained with a limited vocabulary of the targeted domain. Last, they observed the benefit of subword models over word-based NMT models for terminology translation.

Renato et al. [2018] built a statistical machine translation system using in-domain parallel corpora and available machine learning tools for the task of translating clinical term descriptions from Spanish to Brazilian Portuguese. Their approach included different techniques to validate the result of the different systems, using reference domain terminology and the occurrence of translated descriptions in a corpus of medical scientific literature and in domain specific web pages. In order to implement their SMT method, they used Moses [Koehn et al., 2007] software, a phrasal-based probabilistic machine translation engine, which was used by many teams at the recent First Conference on Machine Translation (WMT16) [Bojar et al., 2016]. Input sequences are segmented into a number of (nonlinguistic) phrases, each phrase is translated using a phrase translation table and allow for reordering of phrases in the output.

3 Evaluation Metrics

The automatic translation evaluation is based on the correspondence between the output and reference translation (ground truth/gold standard). We use popular metrics that cover several approaches:

- BLEU (Bilingual Evaluation Understudy) Papineni et al. [2002] is calculated for individual translated segments (n-grams) by comparing them with a dataset of reference translations. Low BLEU score means high mismatch and higher score means a better match.

	Resources	MT	Method	Languages	Evaluation
Nyström et al. [2006]	ICD-10, ICF, MeSH	SMT	Alignment	En-Swe	Re, Pr, F1
Deléger et al. [2010]	MeSH, SNMI, MedDRA 17, WHO-ART	SMT	Knowledge, Corpus	En-Fr	Human
Laroche and Langlais [2010]	Wiki	SMT	Projection-based	fr-en	P_N, R_N, F_N
Dušek et al. [2014]	EMEA, UMLS, MAREC	SMT	Domain	Multi	BLEU
Silva et al. [2015]	SNOMED CT, DBPedia	Auto	Alignment	en-por	Jaro distance
Wołk and Marasek [2015]	EMEA	NMT	Encoder-Decoder	pol-en	BLEU, METEOR, TER, NIST
Arcan et al. [2016]	Organic.Lingua	SMT	Domain	en-(ge, it, sp)	BLEU, METEOR, TER
Arcan and Buitelaar [2017]	ICD, Wiki	Comp	Knowledge Base	en-ge	BLEU, METEOR, chrF3
Renato et al. [2018]	DeCS, Dicionario Medico, Wiki	SMT	Domain	sp-por	BLEU, METEOR, TER
Khan et al. [2018]	UFAL, PatTR	NMT	Domain	en-fr	BLEU, METEOR, TER

Table 1: Comparison of the most popular techniques for medical terms and texts translation.

- SacreBLEU Post [2018] computes scores on detokenized outputs, using WMT (Conference on Machine Translation) tokenization and it produces the same values as the official script (`mteval-v13a.pl`) used by WMT.
- METEOR (Metric for Evaluation of Translation with Explicit ORdering) by Lavie and Agarwal [2007] includes exact word, stem and synonym matching while producing a good correlation with human judgement at the sentence or segment level (unlike BLEU which seeks correlation at the corpus level).
- TER (Translation Edit Rate) Snover et al. [2006]: the metric detects the number of edits (words deletion, addition and substitution) required to make a machine translation match exactly to the closest reference translation in fluency and semantics. High TER means high mismatch, while lower score means smaller distance from the reference text.

3.1 Comparison

In this section, we present a comparison of the most recognised and cited studies. We show papers with resources, family of machine translation approach, specific method used, languages studied and evaluation metrics, sorted by year. Table 1 summarizes the related work on medical terms and texts translation.

While Nyström et al. [2006] is very strong as it enables interactive alignment, it is also highly user-dependent as a semi-automatic approach. Nevertheless, it is one of the methods with the largest number of resources used. Deléger et al. [2010] managed to identify attested translations that a human translator might not have thought of, especially when translating terminologies without textual context, providing a fair part of the MedLinePlus terminology translation. On the other hand, their method is highly affected by the corpus used and as they state, the content of the documents could be examined, so as to select the most relevant texts and thus process more focused data.

Later work by Wołk and Marasek [2015], while utilizing more popular neural machine translation (NMT) tools, shows that NMT for medical terms requires more research and is not straightforward, providing worse results than traditional statistical machine translation (SMT) systems.

Last, Renato et al. [2018] shows the advantages of using in-domain parallel corpora but is limited to traditional statistical machine translation methods.

4 Input data

4.1 Datasets in glossary v4

In this subsection, we present some initial statistics on the given glossary. In Table 2 we show the frequency of unique values per column and frequency of unique groups of **property** in glossary v4.

row ID	159371
uri	68210
property	195
string_en	139049
string_fr	141910
terminologie	4
method	4
string_en_rev	111342
distance (string_en, string_en_rev)	335
string_fr_length	1184
string_fr_length_suffix	10840
rank	10844
string_en_bis	136489
string_fr_bis	138872

synonym	26802
exclusion	10562
narrowerTerm	14872
title	56312
definition	15732
inclusion	14322
preferred	12435
shortTitle	743
criteria	324
consider	268
note	212
label	912
text	146
coding-hint	212
introduction	9
footnote	1
preferredLong	797
nan	4710

Table 2: Frequency of unique values per column in glossary-v4 (left), and frequency of unique groups of **property** in glossary-v4 (right).

The unique values of columns **terminologie** and **method** are given below:
terminologie: icd-11, icpc, dbpedia, icd-10
method: google, CISPClub, dbpedia, fde

In Table 3 we show the number of rows grouped by columns **terminologie** and **method**.

dbpedia	dbpedia	912
icd-10	fde	32515
icpc	CISPClub	3254
icd-11	google	122690
	Sum	**159371**

Table 3: Number of rows with unique values for **terminologie** and **method** in glossary-v4.

Inside the glossary, there are 1133 unique names and 18 unique group names for the column **property**. In order to extract them, we keep only the first part of the name.

property: definition, nan, label, consider, text, title, shortTitle, preferredLong, preferred, introduction, coding-hint, footnote, exclusion, inclusion, synonym, narrowerTerm, note, criteria

Last, in Table 2 we observe the frequency of each group in the column **property**.

Finally to create our parallel corpus, we keep only the ICD10, dbpedia and ICPC lines, ending up with 36705 lines. We keep only the lines where both english and french translation were available.

4.2 Datasets in glossary v7

During our study we experimented upon numerous medical dictionaries and datasets:

1. ATC [Anatomical Therapeutic Chemical, 2019]. The Anatomical Therapeutic Chemical (ATC) Classification System is a drug classification system that classifies the active ingredients of drugs according to the organ or system on which they act and their therapeutic,

pharmacological and chemical properties. It is controlled by the World Health Organization Collaborating Centre for Drug Statistics Methodology (WHOCC), and was first published in 1976. Namely, the dataset includes descriptions on metabolism, blood, dermatological and other contents.

2. CLADIMED [CLADIMED, 2019]. CLADIMED is a five levels classification for medical devices, based on the ATC classification approach (same families). Devices are classified according to their main use and validated indications. It was originally developed by AP-HP (hospitals of Paris).
3. DICT_ACAD_MED [Académie de Médecine, 2019]. The "dictionnaire médical de l'académie de médecine" identifies 63000 terms used in health and defines them under the supervision of the french national academy of medecin. Most of the terms are precised with a english translation.
4. ICD-O [World Health Organization, 2019]. The International Classification of Diseases for Oncology (ICD-O) (1) has been used for nearly 35 years, principally in tumor or cancer registries, for coding the site (topography) and the histology (morphology) of the neoplasm, usually obtained from a pathology report.
5. MESH_INSERM [FR MESH, 2019]. MeSH (Medical Subject Headings) is a reference thesaurus in the biomedical field. The NLM (U.S. National Library of Medicine), which built and updates it every year, uses it to index and query its databases, including MEDLINE/PubMed. INSERM, which has been the French partner of the NLM since 1969, translated the MeSH in 1986, and has been updating the French version every year since then. The bilingual version is often used as a translation tool, as well as for indexing and querying databases in French.
6. MedDRA [ICH, 2019]. MedDRA was developed in the late 1990s by the International Council for Harmonisation of Technical Requirements for Pharmaceuticals for Human Use (ICH). It constitutes a rich and highly specific standardised medical terminology to facilitate sharing of regulatory information internationally for medical products. MedDRA is available to all for use in the registration, documentation and safety monitoring of medical products both before and after a product has been authorised for sale.
7. ORDO [Vasant et al., 2014]. The Orphanet Rare Disease Ontology (ORDO) is a structured vocabulary for rare diseases derived from the Orphanet database, capturing relationships between diseases, genes and other relevant features. Orphanet was established in France by the INSERM (French National Institute for Health and Medical Research) in 1997. ORDO provides integrated, re-usable data for computational analysis.
   types of diseases = ['Disease', 'Malformation syndrome', 'Clinical subtype', 'Group of phenomes', 'Particular clinical situation in a disease or syndrome', 'Morphological anomaly', 'Etiological subtype', 'Biological anomaly', 'Clinical syndrome', 'Histopathological subtype']
   In Table 4 we show some statistics on the Epidemiological data in ORDO.

Number of diseases	9406
Types	10

Table 4: Number of types and diseases in Epidemiological data in ORDO.

8. dbpedia [Auer et al., 2007]. Through its API, dbpedia exposes multilingual fields and then can be used as a source to consolidate bi-lingual corpora.
9. ICD-10 [World Health Organization, 2016]. ICD-10 is the 10th revision of the International Statistical Classification of Diseases and Related Health Problems (ICD), a medical classification list by the World Health Organization (WHO). It contains codes for diseases, signs and symptoms, abnormal findings, complaints, social circumstances, and external causes of injury or diseases. Work on ICD-10 began in 1983, became endorsed by the Forty-third World Health Assembly in 1990, and was first used by member states in 1994.
10. ICPC-2E [Verbeke et al., 2006]. ICPC-2 classifies patient data and clinical activity in the domains of General/Family Practice and primary care, taking into account the frequency distribution of problems seen in these domains. It allows classification of the patient's reason for encounter (RFE), the problems/diagnosis managed, interventions, and the ordering of these data in an episode of care structure.

Medical dictionaries & documents (En-Fr)	**#pairs**
ATC [Anatomical Therapeutic Chemical, 2019]	5536
CLADIMED [CLADIMED, 2019]	4169
DICT_ACAD_MED [Académie de Médecine, 2019]	47603
ICD-O [World Health Organization, 2019]	3671
MESH_INSERM [FR MESH, 2019]	29351
MedDRA [ICH, 2019]	23954
ORDO [Vasant et al., 2014]	50425
dbpedia [Auer et al., 2007]	912
ICD-10 [World Health Organization, 2016]	32515
ICPC-2E [Verbeke et al., 2006]	3046
Total:	**201181**

Table 5: The medical dictionaries and documents collection we experimented on, along with the number of parallel sentences extracted.

In Table 5 we present the collection of medical dictionaries and documents we explored during our research studies, and the number of parallel sentences we extracted from each of them.

5 Proposed Methodology

Here we briefly describe the architecture of the approach we are going to design. In our attempt we are going to combine previous state-of-the-art methods for medical terminology machine translation systems like Deléger et al. [2010] and Renato et al. [2018], with the power of neural machine translation systems like semi-supervised learning [Cheng et al., 2016, Skorokhodov et al., 2018] and recently introduced cross-lingual language models (XLMs) Lample and Conneau [2019].

In Figure 8, we present an abstractive illustration of our proposed architecture. Previous statistical machine translation approaches are shown in the translation and language model parts of the schema, while more recent neural methods are presented in the bottom part of our design. Our architecture will enable us to use both bilingual (ICD9, UMLS, ICPC, LOINC) and target monolingual corpora (Snomed 3.5-VF) of the medical domain, and make the most out of traditional statistical phrase-based machine translation tools and state-of-the-art neural methods on source monolingual corpora as well (ICD10). At the final stage of our architecture, we will use popular evaluation metrics for machine translation, like BLEU, METEOR and ROUGE.

The technologies we are going to use will be open and available. As programming language we select Python, as it consists one the most used by the research community for machine translation. Moreover, the file formats can be databases with json and xml supported, with active APIs for user requests. Last, we are going to utilize graphics processing units (GPUs) for faster learning, when neural approaches will be applied.

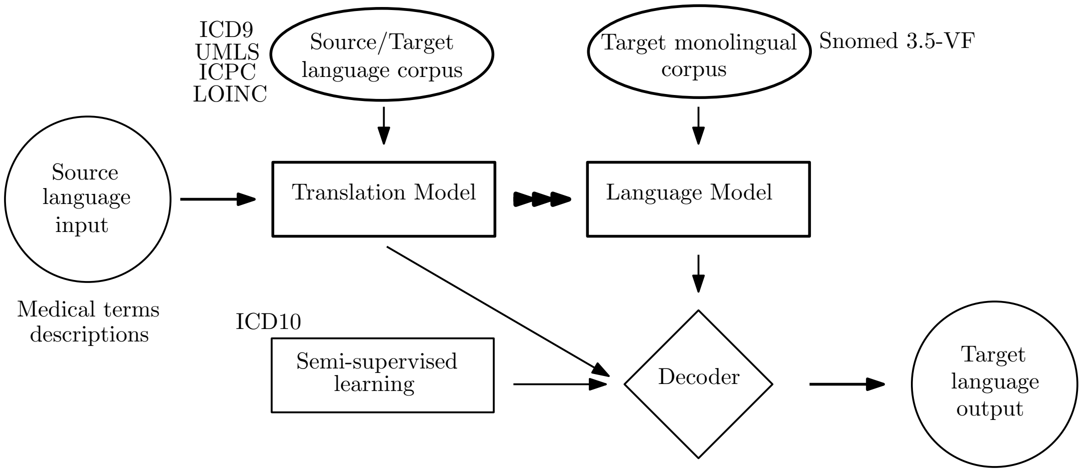


Figure 8: Our proposed methodology.

Within the related work, we tracked and identified the state-of-the-art approaches for machine translation and more specifically for the task of medical terminology translation. This attempt will enable us to initiate the ICD-11 translation bootstrap with end-to-end statistical, as well as neural machine translation methodologies. The problem requires a diverse and multilevel approach as it is also needed to be open and easily edited for future updates of medical terms and their descriptions.

As we discussed already, the closest approaches for addressing our task are Deléger et al. [2010] and Renato et al. [2018]. These are the methods that we will try to reproduce first, utilize as baselines and build our proposed methodology on top of them.

Finally, we strongly believe that state-of-the-art approaches, like Cheng et al. [2016], Skorokhodov et al. [2018] and Lample and Conneau [2019] that have never been tested in medical terminology translation are the most promising and will probably be the most effective.

6 The MOSES baseline (statistical machine translation approach)

In this section, we present the commands needed to run the MOSES baseline, trained on ICD-10, ICPC and dbpedia corpora, provided by ASIP Santé. We require access to a UNIX environment with a terminal in order to run the following commands. In Windows, the commands can be run in powershell.

6.1 Installing MOSES

The following command is for installing MOSES in MacOS systems:

```
./bjam --with-boost=/usr/local/Cellar/boost/1.70.0/ --with-cmph
    =/Users/konstantinosskianis/Documents/icd11-translation/cmph
    -j8 -toolset=clang -q -d2 -a
```

For Linux and Windows we point the readers to the following link: http://www.statmt.org/moses/?n=Development.GetStarted.

6.2 Running MOSES

Assuming we have our parallel corpus ready, both english and french text files, we can start creating our translation model. We are going to use MOSES [Koehn et al., 2007], which is one of the most popular statistical machine translation tools. Again, in this baseline we make use only of ICD10, dbpedia and ICPC definitions and descriptions. After the training, we are going to test our model to ICD11 definitions and descriptions.

Step 1. Running tokenizers for both english and french corpora. Spaces have to be inserted between words and punctuation.

```
scripts/tokenizer/tokenizer.perl -l en -no-escape < /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    parallel_en.txt > /Users/konstantinosskianis/Documents/icd11-
    translation/corpus/parallel_medical.tok.en

scripts/tokenizer/tokenizer.perl -l fr -no-escape < /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    parallel_fr.txt > /Users/konstantinosskianis/Documents/icd11-
    translation/corpus/parallel_medical.tok.fr
```

Step 2. With truecasing, the initial words in each sentence are converted to their most probable casing. This helps reduce data sparsity. Since we do not care about casing (definitions and descriptions), all can be transformed to lower case.

```
scripts/recaser/train-truecaser.perl --model /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    truecase-model-medical.en --corpus /Users/konstantinosskianis
    /Documents/icd11-translation/corpus/parallel_medical.tok.en

scripts/recaser/train-truecaser.perl --model /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    truecase-model-medical.fr --corpus /Users/konstantinosskianis
    /Documents/icd11-translation/corpus/parallel_medical.tok.fr

scripts/recaser/truecase.perl --model /Users/konstantinosskianis
    /Documents/icd11-translation/corpus/truecase-model-medical.en
     < /Users/konstantinosskianis/Documents/icd11-translation/
    corpus/parallel_medical.tok.en > /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical.true.en
```

```
scripts/recaser/truecase.perl --model /Users/konstantinosskianis
    /Documents/icd11-translation/corpus/truecase-model-medical.fr
     < /Users/konstantinosskianis/Documents/icd11-translation/
    corpus/parallel_medical.tok.fr > /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical.true.fr
```

Step 3. In the cleaning process, long and empty sentences are removed as they can cause problems with the training pipeline. Mis-aligned sentences are also removed.

```
scripts/training/clean-corpus-n.perl /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical.true en fr /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    medical.clean 1 80
```

Depending on the training set, after the cleaning process:

- Input sentences: 36454 Output sentences: 36308 (Glossary v4)
- Input sentences: 75211 Output sentences: 75058 (Glossary v7, without far)
- Input sentences: 87622 Output sentences: 87465 (Glossary v7)
- Input sentences: 700416 Output sentences: 700098 (Glossary v7 with large MeSH)

Step 4. The language model (LM) training is used to ensure fluent output, so it is built with the target language (i.e. French in this case).

```
mosesdecoder/bin/lmplz -o 3 </Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical.true.fr > /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    medical.arpa.fr
```

Step 5. Here we binarise the *.arpa.fr file for faster loading.

```
mosesdecoder/bin/build_binary /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical.arpa.fr /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    medical.blm.fr
```

Step 6. Next, we are going to train the translation model. To do this, we run word-alignment (using GIZA++, or MGIZA for MacOS), phrase extraction and scoring, create lexicalised reordering tables, which will produce the final Moses configuration file, all with a single command.

```
nohup nice /Users/konstantinosskianis/Documents/icd11-
    translation/mosesdecoder/scripts/training/train-model.perl -
    cores 4 -root-dir train -corpus /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical.clean -f en -e fr
    -alignment grow-diag-final-and -reordering msd-bidirectional-
    fe -lm 0:3:/Users/konstantinosskianis/Documents/icd11-
    translation/corpus/medical.blm.fr:8  -external-bin-dir /Users
    /konstantinosskianis/Documents/icd11-translation/
    word_align_tools -mgiza -mgiza-cpus 8 -parallel >&
    training_en_fr_medical.out
```

Step 7. You'll notice, though, that the decoder takes at least a couple of minutes to start-up. In order to make it start quickly, we can binarise the phrase-table and lexicalised reordering models. To do this, create a suitable directory and binarise the models as follows:

```
bin/processPhraseTableMin -in /Users/konstantinosskianis/
    Documents/icd11-translation/train/model/phrase-table.gz -
    nscores 4 -out /Users/konstantinosskianis/Documents/icd11-
    translation/mosesdecoder/binarised-model/phrase-table
```

```
bin/processLexicalTableMin -in /Users/konstantinosskianis/
    Documents/icd11-translation/train/model/reordering-table.wbe-
    msd-bidirectional-fe.gz -out binarised-model/reordering-table
```

Step 8. Testing our trained model:

```
/Users/konstantinosskianis/Documents/icd11-translation/
    mosesdecoder/bin/moses -f /Users/konstantinosskianis/train/
    model/moses.ini
```

This will output a screen where the user can insert an english word or sentence and the system will output the most probable translation available. Apart from the console, we can give the system a text file with the terms and sentences which we want to translate.

Step 9. We need to preprocess the target file as well:

```
scripts/tokenizer/tokenizer.perl -l en -no-escape < /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    icd_11_en.txt > /Users/konstantinosskianis/Documents/icd11-
    translation/corpus/medical_ICD11.tok.en

scripts/recaser/truecase.perl --model /Users/konstantinosskianis
    /Documents/icd11-translation/corpus/truecase-model-medical.en
     < /Users/konstantinosskianis/Documents/icd11-translation/
    corpus/medical_ICD11.tok.en > /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical_ICD11.true.en

scripts/tokenizer/tokenizer.perl -l fr -no-escape < /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    icd_11_fr.txt > /Users/konstantinosskianis/Documents/icd11-
    translation/corpus/medical_ICD11.tok.fr

scripts/recaser/truecase.perl --model /Users/konstantinosskianis
    /Documents/icd11-translation/corpus/truecase-model-medical.fr
     < /Users/konstantinosskianis/Documents/icd11-translation/
    corpus/medical_ICD11.tok.fr > /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical_ICD11.true.fr
```

Step 10. The model that we've trained can then be filtered for this test set, meaning that we only retain the entries needed translate the test set. This will make the translation a lot faster.

```
/Users/konstantinosskianis/Documents/icd11-translation/
    mosesdecoder/scripts/training/filter-model-given-input.pl
    filtered-medical /Users/konstantinosskianis/Documents/icd11-
    translation/train/model/moses.ini /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical_ICD11.true.en -
    binarizer /Users/konstantinosskianis/Documents/icd11-
    translation/mosesdecoder/bin/processPhraseTableMin

/Users/konstantinosskianis/Documents/icd11-translation/
    mosesdecoder/scripts/training/filter-model-given-input.pl
    filtered-medical /Users/konstantinosskianis/Documents/icd11-
    translation/mosesdecoder/binarised-model/moses.ini /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    medical_ICD11.true.en -Binarizer /Users/konstantinosskianis/
    Documents/icd11-translation/mosesdecoder/bin/
    processPhraseTableMin
```

Step 11. In order to feed the file that will be translated, we run the first command in case we do not apply filtering on the test set, and the second one if we did:

```
nohup nice /Users/konstantinosskianis/Documents/icd11-
    translation/mosesdecoder/bin/moses -f /Users/
    konstantinosskianis/Documents/icd11-translation/train/model/
    moses.ini -threads 8 < /Users/konstantinosskianis/Documents/
    icd11-translation/corpus/medical_ICD11.true.en > /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    medical_ICD11.translated_NEW.fr 2> /Users/konstantinosskianis
    /Documents/icd11-translation/corpus/medical_ICD11.out

nohup nice /Users/konstantinosskianis/Documents/icd11-
    translation/mosesdecoder/bin/moses -f /Users/
    konstantinosskianis/Documents/icd11-translation/mosesdecoder/
    filtered-medical/moses.ini < /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical_ICD11.true.en > /
    Users/konstantinosskianis/Documents/icd11-translation/corpus/
    medical_ICD11.translated.fr 2> /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical_ICD11.out
```

Step 12. We can then compare it to the ground truth translation and get the BLEU metric:

```
/Users/konstantinosskianis/Documents/icd11-translation/
    mosesdecoder/scripts/generic/multi-bleu.perl -lc /Users/
    konstantinosskianis/Documents/icd11-translation/corpus/
    medical_ICD11.true.fr < /Users/konstantinosskianis/Documents/
    icd11-translation/corpus/medical_ICD11.translated_ALL.fr
```

Generating n-Best Lists The generation of n-best lists (the top n translations found by the search according to the model) is pretty straight-forward. You simple have to specify the file where the n-best list will be stored and the size of the n-best list for each sentence.

Example: The command

```
/Users/konstantinosskianis/Documents/icd11-translation/
    mosesdecoder/bin/moses -f /Users/konstantinosskianis/
    Documents/icd11-translation/train/model/moses.ini -threads 8
    -n-best-list listfile 1 < /Users/konstantinosskianis/
    Documents/icd11-translation/corpus/medical_ICD11.true.en
```

6.3 Running with the Experiment Management System

Instead of typing in all the commands, the user can run EMS from the experiments directory, you can use the command:

```
nohup nice mosesdecoder/scripts/ems/experiment.perl -config
    config -exec &> log &
```

then the BLEU score will appear in evaluation/report.1.

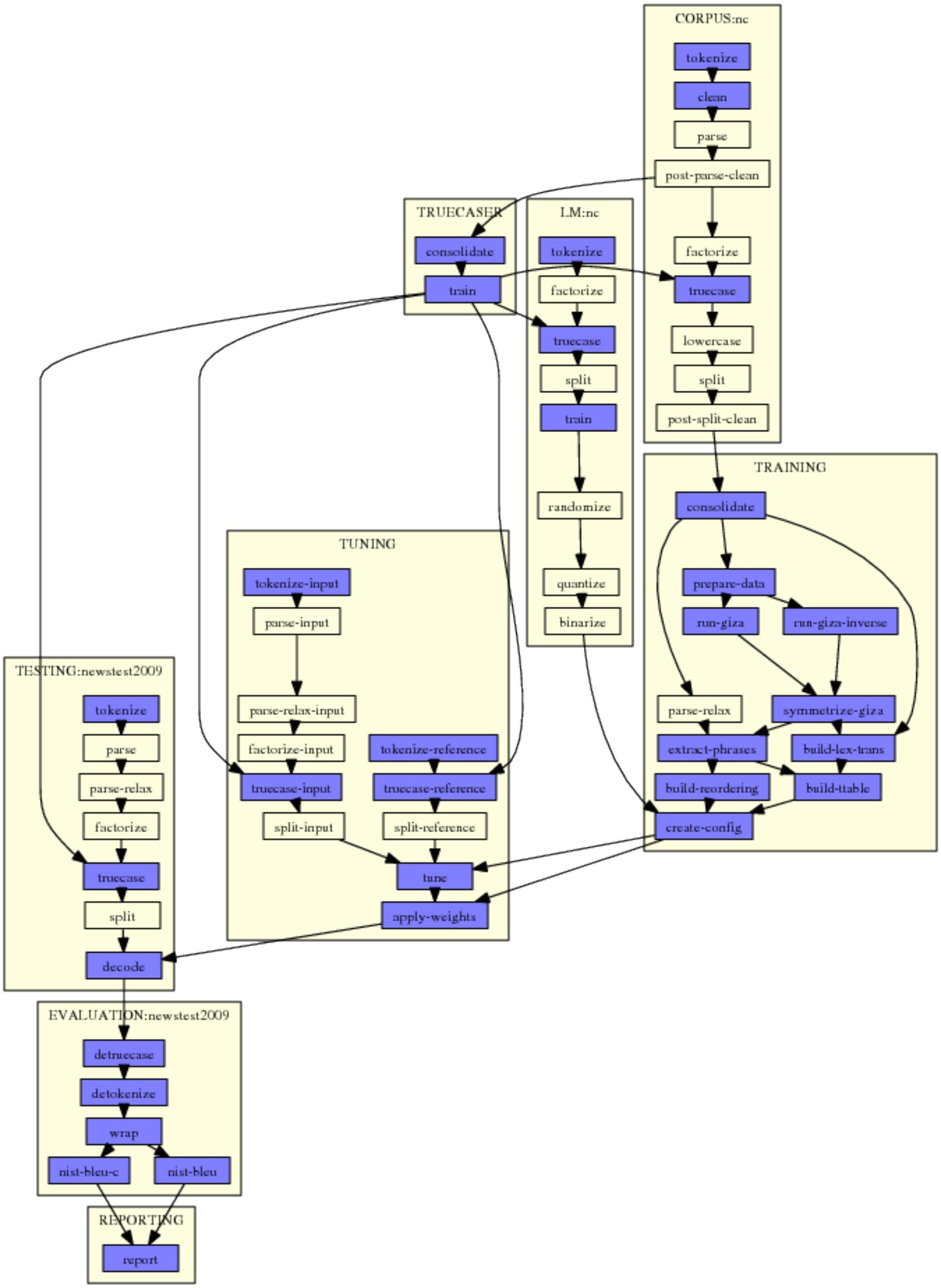


Figure 9: The EMS schema.

6.4 Results with MOSES

Train data	**Test data**	**Ours(MOSES)**	Arcan and Buitelaar [2017]
ICD10, dbpedia, ACPC	ICD11	22.59	ICD10 (eval): 9.11
ICD10, dbpedia, ACPC, Mesh, Orpha, ACAC	ICD11	25.03	»
ICD10, dbpedia, ACPC, Mesh, Orpha, ACAC, PubMED	ICD11	31.52	»

Table 6: MOSES results with the BLEU metric. Here we compare with the SMT method by Arcan and Buitelaar [2017], where their test set was part of ICD10.

Compare with state-of-the-art

6.5 Examples of the MOSES baseline

Next, we offer some testing examples from ICD11 on our trained MOSES baseline.

Successful or acceptable

1. internal intercostal muscle
   BEST TRANSLATION: muscle intercostaux interne [111] [total=-15.667] core=(0.000,-3.000,3.000,-0.471,-0.828,-1.552,-3.246,0.000,-0.954,-0.863,0.000,-2.984,0.000,-6.000,-29.613)
2. supraspinitus tear
   BEST TRANSLATION: dechirure supraspinitus [11] [total=-110.067] core=(-100.000,-4.000,2.000,-1.459,-1.485,-1.740,-3.607,0.000,0.000,-1.053,0.000,-1.363,0.000,-3.000,-22.368)
3. Non-melanin pigmentation due to exogenous substances
   BEST TRANSLATION: pigmentation Non-melanine due aux produits exogenes [111111] [total=-120.145] core=(-100.000,-8.000,4.000,-1.099,-8.367,-4.825,-12.692,-0.511,0.000,-0.732,-0.131,-0.336,0.000,-4.000,-43.672)
4. dementia
   BEST TRANSLATION: demence [1] [total=-2.950] core=(0.000,-3.000,1.000,-0.132,-1.133,-0.534,-3.482,-0.494,0.000,0.000,0.000,0.000,0.000,0.000,-9.891)
5. emphysema
   BEST TRANSLATION: emphyseme [1] [total=-2.924] core=(0.000,-3.000,1.000,-0.078,-1.194,-0.196,-3.568,-0.412,0.000,0.000,0.000,0.000,0.000,0.000,-9.986)
6. subcutaneous tissue
   BEST TRANSLATION: du tissu cellulaire souscutane [11] [total=-3.990] core=(0.000,-5.000,1.000,0.000,-1.657,-0.709,-8.081,-0.031,0.000,0.000,0.000,0.000,0.000,0.000,-14.181)
7. developmental anomalies of the circulatory system
   BEST TRANSLATION: anomalies du developpement de l' appareil circulatoire [111111] [total=-11.815] core=(0.000,-9.000,4.000,-4.141,-10.829,-2.365,-11.490,-0.111,-1.149,-2.628,-0.244,-1.551,-0.050,-4.000,-25.860)
8. mesothelial papilloma
   BEST TRANSLATION: mesotheliales papillome [11] [total=-9.464] core=(0.000,-6.000,2.000,0.000,-1.188,-0.629,-9.303,-2.434,0.000,0.000,-2.197,0.000,0.000,0.000,-24.502)

Unknown words

1. isosorbide dinitrate
   BEST TRANSLATION: isosorbide|UNK|UNK|UNK dinitrate|UNK|UNK|UNK [11] [total=-211.097] core=(-200.000,-2.000,2.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000,-26.994)
2. lymphangiectasia pulmonaire primitive
   BEST TRANSLATION: lymphangiectasia|UNK|UNK|UNK pulmonaire primitive [111] [total=-111.742] core=(-100.000,-3.000,2.000,0.000,-0.719,0.000,-2.396,0.000,-0.336,0.000,0.000,0.000,0.000,-5.000,-25.837)

The total score of each sentence translated is a combination of the following components:

1. distortion score
2. word penalty
3. unknown word penalty
4. 3-gram LM score
5. translation score

6.6 Hybrid Translation

If we want to add explicit knowledge to Moses models by injecting rules, for example translating terminology or numbers, dates etc., Moses has a few ways of making this possible.

7 The Neural Machine Translation Approach

7.1 OpenNMT

We use OpenNMT [Klein et al., 2017]. The system is successor to seq2seq-attn developed at Harvard, and has been completely rewritten for ease of efficiency, readability, and generalizability. The library requires Python 3.5 and torch>=1.2.

7.1.1 The OpenNMT architecture

The default model, which consists of a 2-layer LSTM with 500 hidden units on both the encoder and the decoder. Next we show the full model and the size of each component.

```
NMTModel(
    (encoder): RNNEncoder(
        (embeddings): Embeddings(
            (make_embedding): Sequential(
                (emb_luts): Elementwise(
                    (0): Embedding(24997, 500, padding_idx=1)
                )
            )
        )
        (rnn): LSTM(500, 500, num_layers=2, dropout=0.3)
    )
    (decoder): InputFeedRNNDecoder(
        (embeddings): Embeddings(
            (make_embedding): Sequential(
                (emb_luts): Elementwise(
                    (0): Embedding(35820, 500, padding_idx=1)
                )
            )
        )
        (dropout): Dropout(p=0.3, inplace=False)
        (rnn): StackedLSTM(
            (dropout): Dropout(p=0.3, inplace=False)
            (layers): ModuleList(
                (0): LSTMCell(1000, 500)
                (1): LSTMCell(500, 500)
            )
        )
        (attn): GlobalAttention(
            (linear_in): Linear(in_features=500, out_features=500, bias=False)
            (linear_out): Linear(in_features=1000, out_features=500, bias=False)
        )
    )
    (generator): Sequential(
        (0): Linear(in_features=500, out_features=35820, bias=True)
        (1): Cast()
        (2): LogSoftmax()
    )
)
```

7.1.2 Run OpenNMT

Assuming we have properly installed OpenNMT, we can proceed with the translation process step-by-step.

Step 1. Preprocess the data:

```
for l in en fr; do for f in data/*.$l; do if [[ "$f" != *"test"*
     ]]; then sed -i "$ d" $f; fi;  done; done

for l in en fr; do for f in ../data/*.$l; do perl tools/
    tokenizer.perl -no-escape -l $l -q  < $f > $f.atok; done;
    done

python preprocess.py -train_src ../data/parallel.en.atok -
    train_tgt ../data/parallel.fr.atok -save_data ../data/medical
    .atok.low -lower
```

Step 2. To train the model on a single CPU:

```
python train.py -data ../data/medical.atok.low -save_model
    medical/demo-medical
```

In order to run faster, we can use a GPU:

```
CUDA_VISIBLE_DEVICES=0 python train.py -data att/demo -
    save_model demo-model -gpu_ranks 0
```

Step 3. Translate the test file:

```
python translate.py -gpu 0 -model demo-model_step_100000.pt -src
     ../data/icd_11.en.tok -output pred_ALL.fr -replace_unk
```

Step 4. Evaluate:

```
perl tools/multi-bleu.perl ../data/icd_11.fr.atok < pred.txt
```

BLEU score always lies between 0 to 1, where 0 means total mismatch and 1 means a perfect match. So, a machine translation model is evaluated on its BLUE score. The better the model, the higher the score.

Train data	Test data	Steps	RNN-Attention	BiRNN	Arcan and Buitelaar [2017]
ICD10, dbpedia, ACPC	ICD11	100k	10.93	-	20.89
All	ICD11	100k	13.12	-	»
All + PubMED	ICD11	200k	-	29.11	»

Table 7: Neural machine translation results with the BLEU metric. Here we compare with the NMT methods by Arcan and Buitelaar [2017], tested on a subset of ICD10.

7.2 Fairseq

```
mkdir -p checkpoints/fconv

CUDA_VISIBLE_DEVICES=0 fairseq-train data-bin/iwslt14.tokenized.
    de-en --lr 0.25 --clip-norm 0.1 --dropout 0.2 --max-tokens
    4000 --arch fconv_iwslt_de_en --save-dir checkpoints/fconv
```

Translate with pre-trained models: MODEL_DIR=wmt14.en-fr.fconv-py

The paper results Gehring et al. [2017] are based on training with 8 GPUs for about 37 days.

7.2.1 Train with Fairseq

```
BPEROOT=subword-nmt/subword_nmt

python subword-nmt/subword_nmt/apply_bpe.py -c available_models/
   wmt14.en-fr.fconv-py/bpecodes < ../data/
   parallel_all_glossary7.en.atok > ../fairseq_tr/icd11/train.en

python subword-nmt/subword_nmt/apply_bpe.py -c available_models/
   wmt14.en-fr.fconv-py/bpecodes < ../data/
   parallel_all_glossary7.fr.atok  > ../fairseq_tr/icd11/train.
   fr

python subword-nmt/subword_nmt/apply_bpe.py -c available_models/
   wmt14.en-fr.fconv-py/bpecodes < ../data/icd_11.en.atok > ../
   fairseq_tr/icd11/test.en

python subword-nmt/subword_nmt/apply_bpe.py -c available_models/
   wmt14.en-fr.fconv-py/bpecodes < ../data/icd_11.fr.atok > ../
   fairseq_tr/icd11/test.fr

fairseq-preprocess --srcdict OpenNMT-py/available_models/wmt14.
   en-fr.fconv-py/dict.en.txt --tgtdict OpenNMT-py/
   available_models/wmt14.en-fr.fconv-py/dict.fr.txt --trainpref
    fairseq_tr/icd11/train --testpref fairseq_tr/icd11/test --
   source-lang en --target-lang fr --destdir fairseq_tr/
   icd11_bin

CUDA_VISIBLE_DEVICES=0 fairseq-train fairseq_tr/icd11_bin --
   restore-file OpenNMT-py/available_models/wmt14.en-fr.fconv-py
   /model.pt --arch fconv_wmt_en_fr -s en -t fr --save-dir
   fairseq_tr/icd11/new_model --max-tokens 5000 --valid-subset
   test --criterion label_smoothed_cross_entropy --label-
   smoothing 0.1 --lr-scheduler fixed --force-anneal 50
```

After the train process, we can now translate with either fairseq-generate or fairseq-interactive:

```
fairseq-generate fairseq_tr/icd11_bin/ --path fairseq_tr/icd11/
   new_model/checkpoint_best.pt --beam 5 --remove-bpe

fairseq-interactive --path wmt14.en-fr.fconv-py/model.pt wmt14.
   en-fr.fconv-py --beam 5 --source-lang en --target-lang fr --
   tokenizer moses --bpe subword_nmt --bpe-codes wmt14.en-fr.
   fconv-py/bpecodes < data/final_ICD11.en > data/final_ICD11.fr
   .out
```

7.2.2 Ensemble models

Preparing the data with prepare_en_fr.sh:

```
SCRIPTS=../mosesdecoder/scripts
TOKENIZER=$SCRIPTS/tokenizer/tokenizer.perl
CLEAN=$SCRIPTS/training/clean-corpus-n.perl
NORM_PUNC=$SCRIPTS/tokenizer/normalize-punctuation.perl
REM_NON_PRINT_CHAR=$SCRIPTS/tokenizer/remove-non-printing-char.
   perl
BPEROOT=../subword-nmt/subword_nmt
BPE_TOKENS=40000

src=en
tgt=fr
lang=en-fr
prep=fairseq_en_fr
tmp=$prep/tmp
orig=orig

mkdir -p $tmp $prep

test=~/Documents/icd/data/icd_11.en.atok

# cd $orig

echo "pre-processing train data..."
for l in $src $tgt; do
    #rm $tmp/train.tags.$lang.tok.$l
    cat ’parallel_all_pol’.$l | \
        perl $NORM_PUNC $l | \
        perl $REM_NON_PRINT_CHAR | \
        perl $TOKENIZER -threads 8 -a -l $l >> $tmp/train.tags.
           $lang.tok.$l
done

echo "splitting train and valid..."
for l in $src $tgt; do
    awk ’{if (NR%1333 == 0)  print $0; }’ $tmp/train.tags.$lang.
       tok.$l > $tmp/valid.$l
    awk ’{if (NR%1333 != 0)  print $0; }’ $tmp/train.tags.$lang.
       tok.$l > $tmp/train.$l
done

TRAIN=$tmp/train.fr-en
BPE_CODE=wmt14.en-fr.fconv-py/bpecodes

#rm -f $TRAIN
for l in $src $tgt; do
    cat $tmp/train.$l >> $TRAIN
done

# echo "learn_bpe.py on ${TRAIN}..."
# python $BPEROOT/learn_bpe.py -s $BPE_TOKENS < $TRAIN >
   $BPE_CODE

for L in $src $tgt; do
```

```
    for f in train.$L valid.$L; do
        echo "apply_bpe.py to ${f}..."
        python $BPEROOT/apply_bpe.py -c $BPE_CODE < $tmp/$f >
            $tmp/bpe.$f
    done
done

python $BPEROOT/apply_bpe.py -c $BPE_CODE < $test > bpe.test

perl $CLEAN -ratio 1.5 $tmp/bpe.train $src $tgt $prep/train 1
    250
perl $CLEAN -ratio 1.5 $tmp/bpe.valid $src $tgt $prep/valid 1
    250

/Users/konstantinosskianis/Documents/icd11-translation/
    mosesdecoder/scripts/generic/multi-bleu.perl -lc /Users/
    konstantinosskianis/Documents/icd11-translation/
    ICD_11_pack15avril2019/final_ICD11.fr < /Users/
    konstantinosskianis/Documents/icd11-translation/
    ICD_11_pack15avril2019/ensemble_fairseq.translated.fr
```

7.3 Back-translation

The process of Back Translation in OpenNMT can be implemented in the following way.

1. Train a reverse model.
2. Translate the monolingual data, generate synthetic pairs.
3. Pre-process the monolingual+synthetic data using preprocess.py which creates new demo.train.pt , demo.valid.pt and demo.vocab.pt.
4. Train the model using the newly created PyTorch indices, and use the pre-trained model using "train_from demo-model_xx.xx.xx.pt".

```
python preprocess.py -train_src bpe\_data/train.tgt -train_tgt
    bpe_data/train.src -save\_data reverse/medical.atok.low -
    lower
```

```
CUDA_VISIBLE_DEVICES=0 python train.py -data reverse/medical.
    atok.low -save_model reverse/demo -gpu_ranks 0
```

perl tools/tokenizer.perl -no-escape -l fr -q < snomed.fr > snomed.fr.tok

Step 3. Translate the test file:

```
python translate.py -gpu 0 -model reverse/demo_step_100000.pt -
    src snomed.fr.tok -output snomed_pred.en -replace_unk
```

BLEU = 26.25

7.4 Transfer learning

7.4.1 Preparing

```
SCRIPTS=../mosesdecoder/scripts
TOKENIZER=$SCRIPTS/tokenizer/tokenizer.perl
CLEAN=$SCRIPTS/training/clean-corpus-n.perl
```

```
NORM_PUNC=$SCRIPTS/tokenizer/normalize-punctuation.perl
REM_NON_PRINT_CHAR=$SCRIPTS/tokenizer/remove-non-printing-char.
    perl
BPEROOT=../subword-nmt/subword_nmt
BPE_TOKENS=50000

src=en
tgt=fr
lang=en-fr
tmp=tmp
orig=orig

mkdir -p $tmp $prep

#test=~/Documents/icd/final_ICD11.en
#cat $test | perl $TOKENIZER -threads 8 -a -l en >> test.$lang.
    tok.en

#cd $orig

echo "pre-processing train data..."
for l in $src $tgt; do
     #rm $tmp/train.tags.$lang.tok.$l
     #cat ’parallel_training’.$l | \
     cat ’data/UFAL_medical_shuffled/medical_UFAL’.$l | \
          perl $NORM_PUNC $l | \
          perl $REM_NON_PRINT_CHAR | \
          perl $TOKENIZER -threads 8 -a -l $l >> $tmp/train.tags.
              $lang.tok.$l
     cat ’data/icd_10’.$l | \
          perl $NORM_PUNC $l | \
          perl $REM_NON_PRINT_CHAR | \
          perl $TOKENIZER -threads 8 -a -l $l >> $tmp/valid.tags.
              $lang.tok.$l
done

echo "splitting train and valid..."
for l in $src $tgt; do
     awk ’{if (NR%1333 == 0)  print $0; }’ $tmp/train.tags.$lang.
         tok.$l > $tmp/valid.$l
     awk ’{if (NR%1333 != 0)  print $0; }’ $tmp/train.tags.$lang.
         tok.$l > $tmp/train.$l
done

TRAIN=$tmp/train.fr-en
#BPE_CODE=$prep/code
BPE_CODE=../wmt14.en-fr.fconv-py/bpecodes

#rm -f $TRAIN
for l in $src $tgt; do
     cat $tmp/train.$l >> $TRAIN
done

#echo "learn_bpe.py on ${TRAIN}..."
#python $BPEROOT/learn_bpe.py -s $BPE_TOKENS < $TRAIN >
    $BPE_CODE

for L in $src $tgt; do
     for f in train.$L valid.$L; do
```

```
        echo "apply_bpe.py to ${f}..."
        python $BPEROOT/apply_bpe.py -c $BPE_CODE < $tmp/$f >
           $tmp/bpe.$f
    done
done

#python $BPEROOT/apply_bpe.py -c $BPE_CODE < test.$lang.tok.en >
    data/bpe.test

perl $CLEAN -ratio 1.5 $tmp/bpe.train $src $tgt data/train 1 250
perl $CLEAN -ratio 1.5 $tmp/bpe.valid $src $tgt data/valid 1 250
```

7.4.2 Preprocessing & learning

```
#!/bin/bash

FAIRSEQ=~/Documents/icd/fairseq
PRETRAINED_MODEL=~/Documents/icd/wmt14.en-fr.fconv-py

SEED=1

EXP_NAME=fine-tune

SRC=en
TRG=fr

TRAIN_SRC=~/Documents/icd/french/parallel_training.$SRC
TRAIN_TRG=~/Documents/icd/french/parallel_training.$TRG

DEV_SRC=~/Documents/icd/french/icd_10.$SRC
DEV_TRG=~/Documents/icd/french/icd_10.$TRG

SRC_VOCAB=$PRETRAINED_MODEL/dict.$SRC.txt
TRG_VOCAB=$PRETRAINED_MODEL/dict.$TRG.txt

PRETRAINED_MODEL_FILE=$PRETRAINED_MODEL/model.pt

CORPUS_DIR=~/Documents/icd/french/data
DATA_DIR=~/Documents/icd/french/data-bin

TRAIN_PREFIX=$CORPUS_DIR/train
DEV_PREFIX=$CORPUS_DIR/valid

mkdir -p $CORPUS_DIR
mkdir -p $DATA_DIR

######################################
# Preprocessing
######################################
CUDA_VISIBLE_DEVICES=0 fairseq-preprocess \
    --source-lang $SRC \
    --target-lang $TRG \
    --trainpref $TRAIN_PREFIX \
    --validpref $DEV_PREFIX \
    --destdir $DATA_DIR \
    --srcdict $SRC_VOCAB \
    --tgtdict $TRG_VOCAB \
    --workers ‘nproc‘ \
######################################
# Training
######################################
CUDA_VISIBLE_DEVICES=0 fairseq-train $DATA_DIR \
    --restore-file $PRETRAINED_MODEL_FILE \
    --lr 0.5 --clip-norm 0.1 --dropout 0.1 --max-tokens 3000 \
    --criterion label_smoothed_cross_entropy --label-smoothing
       0.1 \
    --lr-scheduler fixed --force-anneal 50 \
    --arch fconv_wmt_en_fr \
    --reset-optimizer \
    --save-dir checkpoints/fconv_wmt_en_fr_medical_UFAL
```

8 Final approach

Having access to the datasets mentioned in Section 4, we first applied terminology parsing. As dictionaries and medical documents come in different formats, some of them being pdf documents, others in xml or owl format, different parsing tools were enabled. Next, we extracted the labels or descriptions, in order to form the corpus of parallel sentences.

During the pre-processing step, we need to prepare the data for training the translation systems and perform tokenisation, truecasing and cleaning. For the NMT models, the BPE process is applied.

Apart from the necessary parsing, extracting and pre-processing steps, we required a way to evaluate and validate our results. Thus, we submitted the English version of ICD-11 to Google Translate for translation. While Google Translate can not be seen as the best translation available, we can argue that it constitutes one of the state-of-the-art methods for machine translation.

The automatic translation evaluation is based on the correspondence between the output and reference translation (gold standard). For the evaluation part, we use the popular BLEU metric Papineni et al. [2002]. Last, the translation is handed over to experts for analysis, recommending additional medical resources.

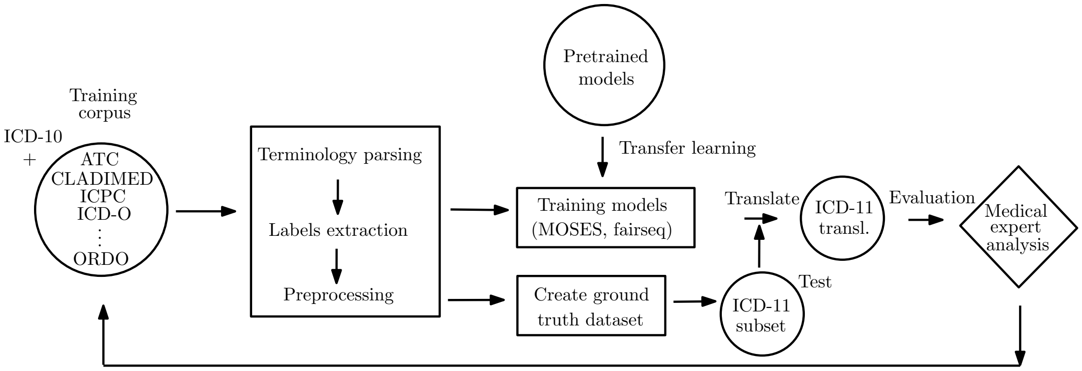


Figure 10: Our final machine translation pipeline for ICD-11.

An abstractive illustration of our proposed methodology is shown in Figure 10. Essentially, the pipeline can be split in five major parts: i) dataset & dictionary datasets' search and retrieval, ii) parsing, extraction and preprocessing, iii) model training, iv) translation and inspection, and v) evaluation and expert analysis.

8.1 Glossary v7

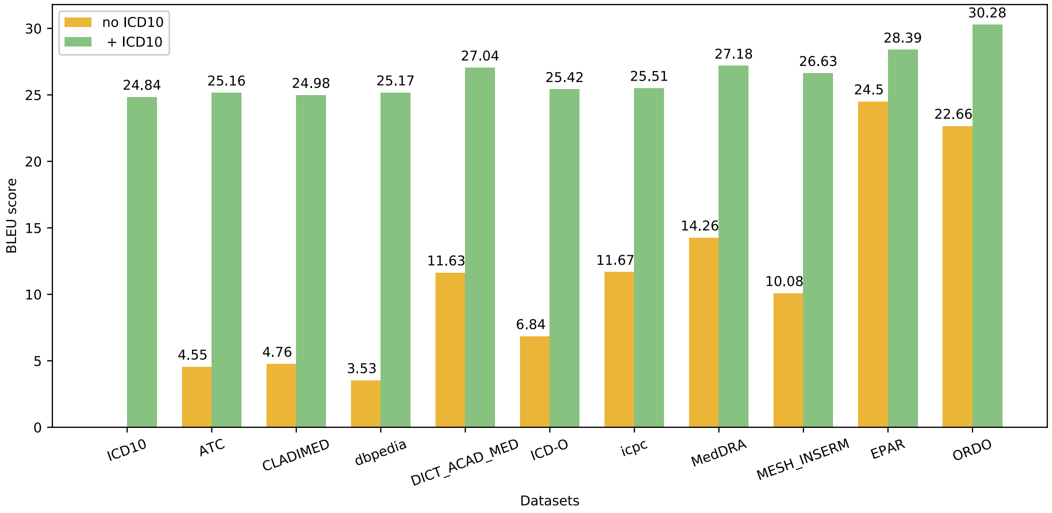


Figure 11: Bleu SMT scores for each dataset without and with ICD10, tested on Google translated ICD11.

8.2 Setup & Experiments

In this section, we present the setup and experiments conducted during this study. As multiple models and tools are available, we ended up in using three, due to their popularity, efficiency and effectiveness. From the SMT approaches we selected MOSES, and for NMT methods, OpenNMT and fairseq.

Create ground-truth dataset Our attempt offers the possibilities of speeding up the process of translating medical lexicons and documents, saving valuable human and computational resources. We evaluate our pipeline in two datasets: a sample of ICD-11 and the whole ICF terminologies. In the case of ICF terminology, we have access to both English and French medical experts validated versions. For ICD-11, since the French official version does not exist yet, we develop a method to evaluate and validate our results.

Through our studies, we discovered that a sample of the English ICD-11 terms can be found in existing French dictionaries. Thus, we can use these terms along with their French translation as already human-validated sentences. We end up having 24242 pairs in English and French that are already integrated in terminologies like ORDO, MESH_INSERM, LOINC_2.66 and others. Although, existing terms may as well require revision by a medical expert, the process indisputably accelerates the translation pipeline, compared to translating a terminology from scratch.

The automatic translation evaluation is based on the correspondence between the output and reference translation (ground truth/gold standard). We use popular metrics that cover several approaches:

- BLEU (Bilingual Evaluation Understudy) Papineni et al. [2002] is calculated for individual translated segments (n-grams) by comparing them with a dataset of reference translations. Low BLEU score means high mismatch and higher score means a better match.
- SacreBLEU Post [2018] computes scores on detokenized outputs, using WMT (Conference on Machine Translation) tokenization and it produces the same values as the official script (`mteval-v13a.pl`) used by WMT.
- METEOR (Metric for Evaluation of Translation with Explicit ORdering) by Lavie and Agarwal [2007] includes exact word, stem and synonym matching while producing a good correlation with human judgement at the sentence or segment level (unlike BLEU which seeks correlation at the corpus level).
- TER (Translation Edit Rate) Snover et al. [2006]: the metric detects the number of edits (words deletion, addition and substitution) required to make a machine translation match

exactly to the closest reference translation in fluency and semantics. High TER means high mismatch, while lower score means smaller distance from the reference text.

MOSES on all medical datasets For the MOSES method we train a 3-gram language model. The popular SMT method returned a score of 31.30 BLEU points (Table 8), when trained on the union of all the medical dictionaries. The individual results are displayed Figure 11. Apart from running MOSES on the union of all the datasets, we also ran the translation process for each dataset separately, with and without ICD-10. Using only ICD-10, the MOSES system reaches 24.84 in BLEU points. ICD-10 managed to perform better than any other dictionary/dataset alone. On the other hand, using only ATC, CLADIMED and dbpedia, resulted in poor performance, probably due to their specificity. Moreover, we observe that adding ICD-10 to all datasets individually boosts the performance dramatically, as expected since many ICD-11 concepts come from ICD-10. Finally, only by using the ORDO dataset, we manage to reach a satisfying BLEU score. ORDO's effectiveness can be attributed to the large number of rare diseases it covers, which was one of the main improvements of ICD-11.

OpenNMT We used the default OpenNMT(-py) parameters, i.e. 2 layers, 500 hidden LSTM units, global attention, batch size of 64, 0.3 dropout probability and a dynamic learning rate decay. OpenNMT reached a similar performance to MOSES, with 30.28 BLEU points. The NMT model did not manage to surpass the SMT model, probably due to the small number of pair-sentences in the training dataset. The library offers numerous architectures, and each architecture comes with a high number of parameters that can be tuned. Thus, testing all these possible combinations is prohibitive. Nevertheless, experimenting with some of them, did not yield any significant improvements.

fairseq's pre-trained model fairseq provides online pre-trained models on many language pairs, offering multiple architectures, trained on large amount of textual data. To the best of our knowledge, this work is one of the first using general-purpose pre-trained models for translating medical terminologies.

For our experiments we selected 'wmt14.en-fr.fconv-py' [Gehring et al., 2017]. The convolutional model was trained on the WMT'14 English-French dataset. The full training set consisted of 35.5M sentence pairs, where sentences longer than 175 words were removed. Last, a size of 40K BPE types was selected for the source and target vocabulary. We used the same BPE types for encoding the test dataset in both languages. The model required 8 GPUs for about 37 days for training, as stated in Gehring et al. [2017].

The fairseq pre-trained model reports a very good BLEU score, with 78.43 points, due to its massive volume of training data. Nevertheless, fairseq fails to translate all sentences in a satisfying manner. For example, the sentence "Syphilitic mitral valve stenosis" returns "- - - - - - - - - - -". The phenomenon of extraneous translations, like "HAUT DE LA PAGE" or "PEPUDU", can be confirmed by searching analogous patterns across the whole output. To address this, we combine fairseq with the MOSES SMT approach, as described in the next paragraph.

fairseq + MOSES As a last approach, the fairseq pre-trained model is used as the principal translation model and MOSES SMT as a secondary-assistant. First, translations are produced by both models separately. Then, we proceed with manual translation inspection, searching for specific patterns, which turn out to be irrelevant translations. A number of these patterns can be easily discovered, by observing the predictions of fairseq (e.g. sequences of symbols or numbers, sentences containing "HAUT DE LA PAGE" etc.). Last, a score s for each translation result is given by fairseq. Essentially the score is the average log-likelihood of the translation. Thus the probability of the hypothesis p can be taken by $\exp(s)$. For sentences with probability $p < 0.15$, and for sentences that follow specific patterns (which constitute clearly erroneous translations), the MOSES translation is selected. With this simple approach, the system outputs relevant translations, instead of returning unrelated sequences of tokens. The combination of fairseq and MOSES gives the best performance with 79.50 BLEU points.

Method	**Type**	**BLEU**
MOSES	SMT	31.30
OpenNMT (RNN-attention)	NMT	30.28
fairseq pretrained	NMT	78.43
fairseq + MOSES	SMT + NMT	79.50

Table 8: BLEU score, testing on Google translated ICD-11.

Method	Type	SacreBLEU ↑	BLEU ↑	METEOR ↑	TER ↓
MOSES no ICD10 (sys1)	SMT	39.92	35.61	33.84	50.61
MOSES only ICD10 (sys2)	SMT	45.84	39.16	35.18	45.22
MOSES dicts with ICD10 (sys3)	SMT	**65.59**	**57.50**	**46.20**	**28.62**
fairseq CNN no pretrain (sys4)	NMT	51.02	42.93	38.85	38.98
fairseq CNN only pretrained (sys5)	NMT	29.98	27.18	29.22	59.02
fairseq CNN finetuned on medical dicts (sys6)	NMT	62.32	53.40	41.41	34.92
fairseq CNN finetuned on medical UFAL (sys7)	NMT	32.57	28.78	30.45	54.19

Table 9: SacreBLEU, BLEU, METEOR and TER scores on validated sample of ICD-11. Bold indicates best performance. SacreBLEU, BLEU and METEOR need to be maximized, while TER needs to be minimized.

Method	Type	SacreBLEU ↑	BLEU ↑	METEOR ↑	TER ↓
MOSES dicts with ICD10 (sys8)	SMT	50.40	42.82	38.74	38.50
fairseq finetuned on medical term/gies (sys9)	NMT	**60.82**	**52.46**	**42.97**	**32.59**

Table 10: Results on the ICD-11 24k sample, removed by the training dataset.

The summarized results from our experiments are visualized in Table 8. We also present translation examples coming from our trained models in Table 14.

The main objective of our work is to examine how fast and effectively a translation to a newly created or updated medical terminology can be created, that could be then given to medical experts for finalization. Our attempt presents the possibilities of speeding up the process of translating medical lexicons and documents, saving human and computational resources.

fairseq's CNN finetuned on medical terminologies The finetuned model (sys6) incorporates transfer learning as it continues training the pre-trained CNN model by fairseq Gehring et al. [2017], described in the previous paragraph, on medical terminologies, presented in Section 4. The model (sys6) almost reached the performance of the SMT approach, with a performance of 62.32 SacreBLEU points and 53.40 BLEU points, while being close to sys3 in both METEOR and TER points as well. As we will also present later in our analysis paragraph, the finetuned model (sys6) is better in translating long sentences (len>50) than its MOSES rival (sys3), shown in Table 11.

fairseq's CNN finetuned on UFAL We also experimented on fine-tuning with the medical UFAL[3] dataset, a large medical domain corpus. The model (sys7) showed a performance of 28.78 BLEU points, being slightly better than using only the pre-trained CNN model. The low score can be attributed firstly to the short length nature of most ICD-11 sentences and secondly to the terminology syntax, which follows a specific structure. The medical UFAL consists mostly of long medical documents, which do not necessarily follow the typology of terminologies.

Removing the test sample from training As shown in Table **??**, the validated sample of the ICD-11, which consists of 24k terms, is also included in the training dataset. Thus, we trained our two best

[3] `https://ufal.mff.cuni.cz/ufal_medical_corpus`

freq	sys3	sys6	len	sys3	sys6
1	0.8187	0.7858	-	-	-
2	0.8139	0.7626	<10	52.66	47.79
3	0.8263	0.7830	[10,20)	63.49	63.95
4	0.8429	0.7901	[20,30)	63.19	63.37
[5,10)	0.8521	0.8075	[30,40)	62.35	62.19
[10,100)	0.8714	0.8331	[40,50)	62.34	58.81
[100,1000)	0.7754	0.7749	[50,60)	59.64	59.82
≥1000	0.7773	0.7638	≥60	52.63	60.27

Table 11: Left: ICD-11 word accuracy analysis via `fmeasure` by frequency bucket. Right: sentence analysis by length bucket with BLEU metric for scoring.

Method	Type	SacreBLEU ↑	BLEU ↑	METEOR ↑	TER ↓
MOSES dicts with ICD10 (sys3)	SMT	12.55	11.90	19.88	70.02
fairseq finetuned on medical term/gies (sys6)	NMT	**72.73**	**69.50**	**47.78**	**20.79**

Table 12: Results on translating the ICF terminology.

freq	sys3	sys6	len	sys3	sys6
1	0.3009	0.5323	-	-	-
2	0.2528	0.8251	<10	15.56	69.08
3	0.4284	0.8087	[10,20)	12.83	70.75
4	0.3315	0.8541	[20,30)	13.39	68.95
[5,10)	0.3501	0.8564	[30,40)	11.43	67.51
[10,100)	0.3812	0.8700	[40,50)	11.33	70.97
[100,1000)	0.5195	0.8761	[50,60)	6.44	69.93
≥1000	0.6644	0.8784	≥60	9.29	66.20

Table 13: Left: ICF word accuracy analysis via `fmeasure` by frequency bucket. Right: sentence analysis by length bucket with BLEU metric for scoring.

architectures (sys3 & sys6) with removing the test set from the training corpus, creating two new models (sys8 & sys9). Table 10 presents their performance, showing that the neural model is far superior from the statistical approach.

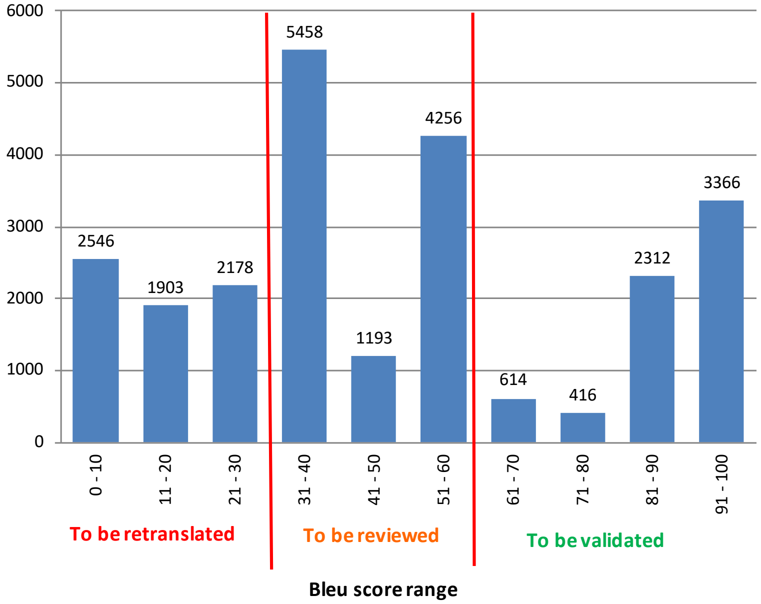


Figure 12: Sentence BLEU scoring on the 24k ICD-11 sample, categorized by a medical expert.

Testing on ICF Since the validated sample of ICD-11 was mostly known sentences of short size belonging to terminologies, we believe that the SMT approach will perform worse than NMT in generalizing to unknown terms and sentences. To confirm this hypothesis, we tested on ICF, where the average length is 10.79 and thus larger than the ICD-11 average length. We tested our two best models, MOSES trained with all the datasets (sys3) and the finetuned CNN fairseq model (sys6) toward the ICF terminology. The finetuned CNN (sys6) performs far better than MOSES (sys3), by a large difference, with 69.50 BLEU points compared to a low 11.90 BLEU points, respectively. sys6 is also far superior to sys3 in terms of METEOR and TER points. The scores are presented in Table 12.

Analysis We also challenged our best SMT and NMT methods with `compare-mt`[4] [Neubig et al., 2019] to study their output. The tool offers aggregate scoring with BLEU and other metrics, word

[4] `https://github.com/neulab/compare-mt`

Ground truth/Reference	**MOSES trained on medical terminologies (sys3)**	**fairseq CNN fine-tuned on medical terminologies (sys6)**
Ref: pied convexe congénital bilatéral	pied convexe congénital bilatéral (100)	astragale verticale congénitale bilatérale (0)
Ref: syphilis des ostia coronaires	syphilis des ostia coronaires (100)	maladie ostiale coronarienne syphilitique (0)
Ref: chute accidentelle de la personne portée	personne portée (9.56)	chute accidentelle de la personne portée (100)
Ref: maladie des inclusions microvilleuses	atrophie microvillositaire congénitale (0)	maladie des inclusions microvilleuses (100)

Table 14: Translation examples of our trained models on the verified sample of ICD-11, given by `compare-mt`. The number in parenthesis shows the sentence translation score in BLEU points compared to reference.

BLEU score range	**English label**	fairseq proposal (sys6) **Human translations**	**Comments**
0-0,2	Familial hypophosphataemic rickets	rickets hypophosphatémiques familiaux **Rachitisme familial hypophosphatémique**	Unknown word
	Adult-onset Still disease, buttock	apparition d'un adulte maladie mortelle, fessier **Maladie de Still survenant chez l'adulte, fesse**	Proper name misunderstood/not recognised
	common bile duct blunt injury	lésion de contour du canal biliaire commun **blessure contondante du canal cholédoque**	ambiguity of label (common)
0,21-0,5	Context of assault, gang rivalry	contexte de l'agression, rivalité entre gangs **Contexte d'agression, rivalité entre gangs**	Inappropriate insertion of article
	Barrett adenocarcinoma	adénocarcinome barrett **Adénocarcinome de Barrett**	proper name misunderstood missing coordination term
	Fracture of thumb bone	fracture du pouce **Fracture de l'os du pouce**	missing word
0,51-0,9	ureter cyst	cyste de l'uretère **kyste de l'uretère**	unknown word translated with editorialy very close term
	talipes equinovalgus	pied bot equinovalgus **talipes equinovalgus**	use of correct synonym
	Unintentional exposure to or harmful effects of oxazolidinediones	exposition non intentionnelle ou effets nocifs des oxazolidinediones **Effets nocifs ou exposition accidentelle à des oxazolidinediones**	word order and use of correct synonym

Table 15: Comparison of translation outputs with human translations.

accuracy via `fmeasure`[5], sentence bucket and n-gram difference analysis. Our analysis is summarized in Table 11. We see that the MOSES model (sys3) performance ranges depending the frequency of terms, while our finetuned CNN (sys6) remains stable, regardless of the frequency. Looking at the right part of Table 11, sys3 perform worse when the length of terms increases significantly (len>50), but remains better than its rival (sys6) for length<10.

Regarding the ICF terminology, the results are shown in Table 13. We clearly observe that the finetuned CNN (sys6) manages to translate well all ICF terms regardless of their frequency on words. Moreover, looking at the right part of Table 13, while sys6 provides promising results with both short and long terms, sys3 (the MOSES model) struggles to produce good translations, especially when the length of sentences increases.

We also present translation examples coming from our trained models, which come from `compare-mt`. Table 14 shows four examples of the translation systems. The first two lines present a perfect translation coming from the MOSES model (sys3), while the last two lines show a perfect translation by the finetuned CNN model (sys6), due to general knowledge, coming from transfer learning.

Next, we present a categorization of the translation BLEU scores on the 24k ICD-11 validated sample in Figure 12. The translations were studied by a medical expert, who extracted three categories using manually selected thresholds. A relatively small 27% of the translations required re-translation, a 45% needs to be reviewed and finally a 28% require to be just validated.

Last, a comparison translation outputs with human translations follows in Table 15. We present translation examples, given by the finetuned CNN model with medical terminologies (sys6), compare them with human translations, observing interesting linguistic phenomena. The comparison shows that as the BLEU score increases, the system outputs "less acceptable" translations with cases like unknown words and ambiguities, to more "acceptable" translations with cases like word order and correct synonym use.

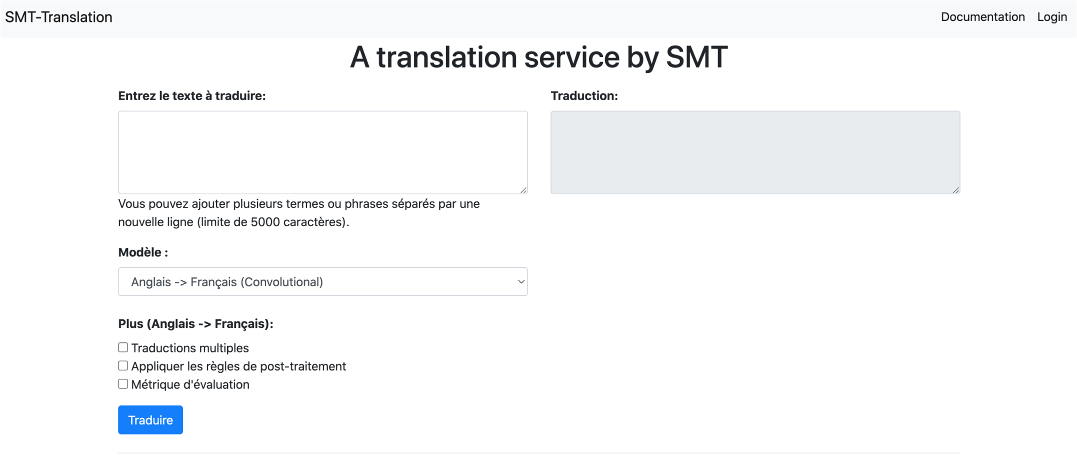
[5] `https://en.wikipedia.org/wiki/F1_score`

8.3 Discussion

The best model was given by the finetuned model by fairseq The power of the large training corpus by fairseq is obvious and beats our pure medical relatively small parallel dataset.

8.4 Demo tool

Last, we developed a live web tool that enables searching our proposed ICD-11 translations, which enables the users to search in English or in French and get the relative translation, along with the translation method. The demo is located at this link: `http://anstranslation.ddns.net:5000`.



SMT-Translation

Documentation Login

A translation service by SMT

Entrez le texte à traduire:

Traduction:

Vous pouvez ajouter plusieurs termes ou phrases séparés par une nouvelle ligne (limite de 5000 caractères).

Modèle : Anglais -> Français (Convolutional)

Plus (Anglais -> Français):

- □ Traductions multiples
- □ Appliquer les règles de post-traitement
- □ Métrique d'évaluation

Traduire

Figure 13: Our demo presents some functionalities of the methodology.

8.5 Scripts & tools

Python scripts:

- stats.py
- umls_orpha_etc.py

Shell scripts:

- glossary_v10.sh
- prepare_en_fr.sh
- fairseq-run_en2fr.sh

9 Other languages

During the project, we tried to translate the ICD11 dictionary to other languages, apart from French.

9.1 Polish

Since for Polish there are no pretrained SMT or NMT models, we need to train our own. For this purpose, we use the parallel English and Polish data from `https://paracrawl.eu/`. The Polish dataset is created by crawling 13,357 websites and the output coming from BiCleaner v5.0 is close to 1.6GB of data, providing us with 6,382,371 parallel sentences, with 145,802,939 words.

```
cd mosesdecoder

scripts/tokenizer/tokenizer.perl -l en -threads 8 -no-escape <
   ~/Documents/icd11-translation/polish/parallel_all_pol.en > ~/
   Documents/icd11-translation/polish/glossary7.tok.en

scripts/tokenizer/tokenizer.perl -l pl -threads 8 -no-escape <
   ~/Documents/icd11-translation/polish/parallel_all_pol.pl > ~/
   Documents/icd11-translation/polish/glossary7.tok.pl

scripts/recaser/train-truecaser.perl --model ~/Documents/icd11-
   translation/polish/truecase-model-glossary7.en --corpus ~/
   Documents/icd11-translation/polish/glossary7.tok.en

scripts/recaser/train-truecaser.perl --model ~/Documents/icd11-
   translation/polish/truecase-model-glossary7.pl --corpus ~/
   Documents/icd11-translation/polish/glossary7.tok.pl

scripts/recaser/truecase.perl --model ~/Documents/icd11-
   translation/polish/truecase-model-glossary7.en < ~/Documents/
   icd11-translation/polish/glossary7.tok.en > ~/Documents/icd11
   -translation/polish/glossary7.true.en

scripts/recaser/truecase.perl --model ~/Documents/icd11-
   translation/polish/truecase-model-glossary7.pl < ~/Documents/
   icd11-translation/polish/glossary7.tok.pl > ~/Documents/icd11
   -translation/polish/glossary7.true.pl

scripts/training/clean-corpus-n.perl ~/Documents/icd11-
   translation/polish/glossary7.true en pl ~/Documents/icd11-
   translation/polish/glossary7.clean 1 80

cd ..

mosesdecoder/bin/lmplz -o 3 <~/Documents/icd11-translation/
   polish/glossary7.true.pl > ~/Documents/icd11-translation/
   polish/glossary7.arpa.pl

mosesdecoder/bin/build_binary ~/Documents/icd11-translation/
   polish/glossary7.arpa.pl ~/Documents/icd11-translation/polish
   /glossary7.blm.pl

nohup nice ~/Documents/mosesdecoder/scripts/training/train-model
   .perl -cores 4 -root-dir ~/Documents/icd11-translation/polish
   /train -corpus ~/Documents/icd11-translation/polish/glossary7
   .clean -f en -e pl -alignment grow-diag-final-and -reordering
    msd-bidirectional-fe -lm 0:3:$HOME/Documents/icd11-
   translation/polish/glossary7.blm.pl:8 -external-bin-dir ~/
   Documents/mosesdecoder/mgiza/mgizapp/bin -mgiza -mgiza-cpus 4
    -parallel >& training_en_pl_medical.out

mosesdecoder/scripts/training/filter-model-given-input.pl  ~/
   Documents/icd11-translation/filtered-medical ~/Documents/
   icd11-translation/train/model/moses.ini ~/Documents/icd11-
   translation/icd_11.en.atok -Binarizer ~/Documents/
   mosesdecoder/bin/processPhraseTableMin

nohup nice ~/Documents/mosesdecoder/bin/moses -f ~/Documents/
   icd11-translation/polish/train/model/moses.ini -threads all <
    ~/Documents/icd11-translation/icd_11.en.atok > ~/Documents/
   icd11-translation/polish/medical_glossary7.translated.pl 2>
   ~/Documents/icd11-translation/polish/medical_ICD11.out
```

9.2 German

For translating from English to German, we can use large pretrained modes by fairseq and run the python script translate_fairseq_de.py

```
import torch

# List available models
torch.hub.list('pytorch/fairseq')  # [..., 'transformer.wmt16.en
    -de', ... ]

# Load a transformer trained on WMT'16 En-De
en2de = torch.hub.load('pytorch/fairseq', 'transformer.wmt16.en-
    de', tokenizer='moses', bpe='subword_nmt')
en2de.eval()  # disable dropout

# The underlying model is available under the *models* attribute
assert isinstance(en2de.models[0], fairseq.models.transformer.
    TransformerModel)

# Move model to GPU for faster translation
en2de.cuda()

f = open("../../final_ICD11.en","r", encoding="utf-8")
#f = co.open("final_SNOMED.en","r", encoding="utf-8")
lines = f.readlines()

f = co.open("pred_fairseq_conv_icd11_TRAN.de","w", encoding="utf
    -8")
for line in lines:
     f.write(en2de.translate(line)+"\n")

f.close()
```

10 Remarks

1. Better preprocessing in training sets? (J41 in the ending of definitions, ICD11 defs with 6digit numbers)
2. Tools:
   - `https://pythonhosted.org/PyMedTermino/tuto_en.html`
   - `https://github.com/chb/py-umls`
3. OWL, JSON or text format
4. Machine to use? Update resources in RosettaHub
5. Hyper-parameter searching for OpenNMT and fairseq

11 Conclusion

In this work, a comparative evaluation of machine translation methods targeting the WHO ICD-11 terminology from English to French is presented. Over ten diverse medical dictionaries along with ICD-10 have been examined. A traditional MOSES SMT approach that manages to produce a good baseline translation is shown. While a combination of MOSES with NMT architectures, and especially largely pre-trained models like fairseq, can improve the quality of the translation.

12 Future work

Many directions can be pointed as future work. First, recent papers have shown the use of BERT for neural machine translation [Imamura and Sumita, 2019, Clinchant et al., 2019]. Additional medical dictionaries and documents can be explored, following the constantly growing area of freely available language resources. Last, novel machine translation techniques have arisen, that can exploit huge monolingual corpora.

References

[1] Académie de Médecine. Dictionnaire Médical de l'Académie de Médecine, 2019.

[2] Anatomical Therapeutic Chemical. Atc, 2019.

[3] Mihael Arcan and Paul Buitelaar. Translating domain-specific expressions in knowledge bases with neural machine translation. *arXiv preprint arXiv:1709.02184*, 2017.

[4] Mihael Arcan, Mauro Dragoni, and Paul Buitelaar. Translating ontologies in real-world settings. In *International Semantic Web Conference*, pages 241–256. Springer, 2016.

[5] Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. Unsupervised neural machine translation. *ICLR*, 2018.

[6] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data, 2007. URL `http://dl.acm.org/citation.cfm?id=1785162.1785216`.

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

[8] Arianna Bisazza, Nick Ruiz, and Marcello Federico. Fill-up versus interpolation methods for phrase-based smt adaptation. In *International Workshop on Spoken Language Translation (IWSLT) 2011*, 2011.

[9] Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, et al. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, volume 2, pages 131–198, 2016.

[10] Rajen Chatterjee, Matteo Negri, Marco Turchi, Marcello Federico, Lucia Specia, and Frédéric Blain. Guiding neural machine translation decoding with external knowledge. In *Proceedings of the Second Conference on Machine Translation*, pages 157–168, 2017.

[11] Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Semi-supervised learning for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1965–1974, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1185. URL `https://www.aclweb.org/anthology/P16-1185`.

[12] CLADIMED. Classification des Dispositifs Médicaux (CLADIMED), 2019.

[13] Vincent Claveau and Pierre Zweigenbaum. Translating biomedical terms by inferring transducers. In *Conference on Artificial Intelligence in Medicine in Europe*, pages 236–240. Springer, 2005.

[14] Stéphane Clinchant, Kweon Woo Jung, and Vassilina Nikoulina. On the use of bert for neural machine translation. *arXiv preprint arXiv:1909.12744*, 2019.

[15] Hervé Déjean, Eric Gaussier, J-M Renders, and Fatiha Sadat. Automatic processing of multilingual medical terminology: applications to thesaurus enrichment and cross-language information retrieval. *Artificial Intelligence in Medicine*, 33(2):111–124, 2005.

[16] Louise Deléger, Magnus Merkel, and Pierre Zweigenbaum. Translating medical terminologies through word alignment in parallel text corpora. *Journal of Biomedical Informatics*, 42(4): 692–701, 2009.

[17] Louise Deléger, Tayeb Merabti, Thierry Lecrocq, Michel Joubert, Pierre Zweigenbaum, and Stéfan Darmoni. A twofold strategy for translating a medical terminology into french. In *AMIA Annual Symposium Proceedings*, volume 2010, page 152. American Medical Informatics Association, 2010.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL*, 2019.

[19] Mona Diab and Steve Finch. A statistical word-level translation model for comparable corpora. In *Content-Based Multimedia Information Access-Volume 2*, pages 1500–1508. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE, 2000.

[20] Ondřej Dušek, Jan Hajič, Jaroslava Hlaváčová, Michal Novák, Pavel Pecina, Rudolf Rosa, Aleš Tamchyna, Zdeňka Urešová, and Daniel Zeman. Machine translation of medical texts in the khresmoi project. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 221–228, 2014.

[21] Matthias Eck, Stephan Vogel, and Alex Waibel. Improving statistical machine translation in the medical domain using the unified medical language system. In *Proceedings of the 20th international conference on Computational Linguistics*, page 792. Association for Computational Linguistics, 2004.

[22] FR MESH. Medical Subject Headings (MESH INSERM), 2019.

[23] Pascale Fung. A statistical view on bilingual lexicon extraction: from parallel corpora to non-parallel corpora. In *Conference of the Association for Machine Translation in the Americas*, pages 1–17. Springer, 1998.

[24] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1243–1252. JMLR. org, 2017.

[25] ICH. Medical Dictionary for Regular Activities, 2019.

[26] Kenji Imamura and Eiichiro Sumita. Recycling a pre-trained bert encoder for neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 23–31, 2019.

[27] Abdul Khan, Subhadarshi Panda, Jia Xu, and Lampros Flokas. Hunter nmt system for wmt18 biomedical translation task: Transfer learning in neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 655–661, 2018.

[28] Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. OpenNMT: Open-source toolkit for neural machine translation. In *Proc. ACL*, 2017. doi: 10.18653/v1/P17-4012. URL `https://doi.org/10.18653/v1/P17-4012`.

[29] Philipp Koehn and Hieu Hoang. Factored translation models. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, 2007.

[30] Philipp Koehn and Josh Schroeder. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the second workshop on statistical machine translation*, pages 224–227, 2007.

[31] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics, 2003.

[32] Philipp Koehn, Marcello Federico, Wade Shen, Nicola Bertoldi, Ondrej Bojar, Chris Callison-Burch, Brooke Cowan, Chris Dyer, Hieu Hoang, Richard Zens, et al. Open source toolkit for statistical machine translation: Factored translation models and confusion network decoding. In *Final Report of the 2006 JHU Summer Workshop*, 2006.

[33] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180, 2007.

[34] Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*, 2019.

[35] Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. Phrase-based & neural unsupervised machine translation. *EMNLP*, 2018.

[36] Audrey Laroche and Philippe Langlais. Revisiting context-based projection methods for term-translation spotting in comparable corpora. In *Proceedings of the 23rd international conference on computational linguistics*, pages 617–625. Association for Computational Linguistics, 2010.

[37] Alon Lavie and Abhaya Agarwal. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the second workshop on statistical machine translation*, pages 228–231, 2007.

[38] Anna Lindgren. Semi-automatic translation of medical terms from english to swedish: Snomed ct in translation, 2011.

[39] Christian Lovis, Robert Baud, Anne-Marie Rassinoux, Pierre-André Michel, and Jean-Raoul Scherrer. Medical dictionaries for patient encoding systems: a methodology. *Artificial intelligence in medicine*, 14(1-2):201–214, 1998.

[40] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *EMNLP*, 2015.

[41] Daniel Marcu and Daniel Wong. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2002.

[42] Kornél Markó, Robert Baud, Pierre Zweigenbaum, Magnus Merkel, Maria Toporowska-Gronostaj, Dimitrios Kokkinakis, and Stefan Schulz. Cross-lingual alignment of medical lexicons. In *Proceedings of Language Resources and Evaluation 2006; Workshop on Acquiring and representing multilingual, specialized lexicons: the case of biomedicine*, pages 5–8, 2006.

[43] Graham Neubig, Zi-Yi Dou, Junjie Hu, Paul Michel, Danish Pruthi, Xinyi Wang, and John Wieting. compare-mt: A tool for holistic comparison of language generation systems. *CoRR*, abs/1903.07926, 2019. URL `http://arxiv.org/abs/1903.07926`.

[44] Mikael Nyström, Magnus Merkel, Lars Ahrenberg, Pierre Zweigenbaum, Håkan Petersson, and Hans Åhlfeldt. Creating a medical english-swedish dictionary using interactive word alignment. *BMC medical informatics and decision making*, 6(1):35, 2006.

[45] Jose Oncina. *Aprendizaje de lenguajes regulares y transducciones subsecuenciales*. PhD thesis, PhD thesis, Universidad Politécnica de Valencia, Valencia, Spain, 1991.

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

[47] Catherine Pease and Abdelaziz Boushaba. Towards an automatic translation of medical terminology and texts into arabic. *Proceedings of the Translation in the Arab World, King Fahd Advanced School of Translation*, pages 27–30, 1996.

[48] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *NAACL*, 2018.

[49] Matt Post. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels, October 2018. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W18-6319`.

[50] Reinhard Rapp. Automatic identification of word translations from unrelated english and german corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 519–526. Association for Computational Linguistics, 1999.

[51] Alejandro Renato, José Castaño, Maria Ávila Williams, Hernan Berinsky, Maria Gambarte, Hee Park, David Pérez, Carlos Otero, and Daniel Luna. A machine translation approach for medical terms. pages 369–378, 01 2018. doi: 10.5220/0006555003690378.

[52] Stefan Schulz, Johannes Bernhardt-Melischnig, Markus Kreuzthaler, Philipp Daumke, and Martin Boeker. Machine vs. human translation of snomed ct terms. In *Medinfo*, pages 581–584, 2013.

[53] Mario J Silva, Tiago Chaves, and Barbara Simoes. An ontology-based approach for snomed ct translation. In *ICBO*, 2015.

[54] Ivan Skorokhodov, Anton Rykachevskiy, Dmitry Emelyanenko, Sergey Slotin, and Anton Ponkratov. Semi-supervised neural machine translation with language models. In *Proceedings of the AMTA 2018 Workshop on Technologies for MT of Low Resource Languages (LoResMT 2018)*, pages 37–44, 2018.

[55] Barry Smith and Christiane Fellbaum. Medical wordnet: a new methodology for the construction and validation of information resources for consumer health. In *Proceedings of the 20th international conference on Computational Linguistics*, page 371. Association for Computational Linguistics, 2004.

[56] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas*, volume 200. Cambridge, MA, 2006.

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

[58] Drashtti Vasant, Laetitia Chanas, James Malone, Marc Hanauer, Annie Olry, Simon Jupp, Peter N Robinson, Helen Parkinson, and Ana Rath. Ordo: An ontology connecting rare disease, epidemiology and genetic data, 2014.

[59] Marc Verbeke, Diëgo Schrans, Sven Deroose, and Jan De Maeseneer. The international classification of primary care (icpc-2): an essential tool in the epr of the gp. *Studies in health technology and informatics*, 124:809, 2006.

[60] Marta Villegas, Ander Intxaurrondo, Aitor Gonzalez-Agirre, Montserrat Marimon, and Martin Krallinger. The mespen resource for english-spanish medical machine translation and terminologies: Census of parallel corpora, glossaries and term translations. *Language Resources and Evaluation*, 05 2018.

[61] Krzysztof Wołk and Krzysztof Marasek. Neural-based machine translation for medical text domain. based on european medicines agency leaflet texts. *Procedia Computer Science*, 64:2–9, 2015.

[62] World Health Organization. ICD-10 : international statistical classification of diseases and related health problems : tenth revision, 2016.

[63] World Health Organization. WHO ICD Oncology (ICDO), 2019.

[64] Hua Wu and Haifeng Wang. Improving domain-specific word alignment with a general bilingual corpus. In *Conference of the Association for Machine Translation in the Americas*, pages 262–271. Springer, 2004.

[65] Kun Yu and Junichi Tsujii. Bilingual dictionary extraction from wikipedia. *Proceedings of machine translation summit xii*, pages 379–386, 2009.