Karlsruhe Institute of Technology

AI4LT

Neural Machine Translation between the Endangered Language Western Armenian and English

Master's Thesis of

Ari Nubar Boyacıoğlu

Artificial Intelligence for Language Technologies (AI4LT) Lab
Institut for Anthropomatics and Robotics (IAR)
KIT Department of Informatics

Reviewer: Prof. Dr. Jan Niehues
Second reviewer: Prof. Dr. Alexander Waibel

May 02, 2023 – November 02, 2023

Karlsruher Institut für Technologie
Fakultät für Informatik
Postfach 6980
76128 Karlsruhe

I declare that I have developed and written the enclosed thesis completely by myself, and have not used sources or means without declaration in the text.

PLACE, DATE

. . . . . .

(Ari Nubar Boyacıoğlu)

Abstract

Western Armenian is one of the Modern Standard Armenian variants and is spoken by the Armenian diaspora residing in Europe, North and South America, and the Middle East. Due to its diasporic nature, the language mostly lives in the domestic environment. It is educated by volunteers in non-official extracurricular activities like weekend school with some exceptions. The language faces the threat of extinction and is classified as "definitively endangered" by UNESCO, as its speakers get older and the younger generation is either not able to learn the language properly or cannot use it outside home. Thanks to the socioeconomic boom in the 19th century, the language has collected a wide array of literary works that are yet to be digitized. The internet presence of Western Armenian is quite low, supported mainly by news and organizational websites and the Western Armenian Wikipedia.

We present the first neural machine translation model that supports Western Armenian and English and is built upon the state-of-the-art multilingual neural machine translation model "No Language Left Behind" by Meta researchers. Along with the model, we provide the first parallel corpus with about 150,000 sentences of a selection of domains. We investigate several research questions regarding low-resource machine translation, the performance of the models trained with Eastern Armenian data on translating to/from Western Armenian, and the potential utilization of such models and any available Eastern Armenian parallel data to boost performance.

With this work, we introduce Western Armenian to the research community, and by providing software to the people of the World. We hope this work may be a humble starting point for Western Armenian natural language processing research, its modernization, digitization, and possibly slowing down or even eliminating the threat of extinction.

Zusammenfassung

Westarmenisch ist eine der modernen Standardsprachen des Armenischen und wird von der armenischen Diaspora in Europa, Nord- und Südamerika sowie im Nahen Osten gesprochen. Aufgrund ihres Diaspora-Charakters wird die Sprache hauptsächlich im häuslichen Umfeld gesprochen. Mit wenigen Ausnahmen wird sie von Freiwilligen außerhalb des offiziellen Curriculums, z.B. in Wochenendkursen, erlernt. Die Sprache ist vom Aussterben bedroht und wird von der UNESCO als "definitiv gefährdetëingestuft, da ein großer Teil ihrer Sprecher älter wird, und die jüngere Generation entweder nicht in der Lage ist, die Sprache richtig zu erlernen oder sie außerhalb des Hauses zu verwenden. Dank des sozioökonomischen Aufschwungs im 19. Jahrhundert hat die Sprache eine Vielzahl literarischer Werke hervorgebracht, die noch digitalisiert werden müssen. Die Internetpräsenz des Westarmenischen ist gering und besteht hauptsächlich aus Nachrichten- und Organisationswebsites sowie dem westarmenischen Wikipedia.

Wir präsentieren das erste neuronale maschinelle Übersetzungsmodell, das Westarmenisch und Englisch unterstützt und auf dem neuesten Stand der Technik befindlichen Modell für mehrsprachige neuronale maschinelle Übersetzung "No Language Left Behind"von Meta-Forschern basiert. Zusammen mit dem Modell stellen wir das erste parallele Korpus mit ca. 150.000 Sätzen aus einer Auswahl von Domänen zur Verfügung. Wir untersuchen verschiedene Forschungsfragen in Bezug auf ressourcenarme maschinelle Übersetzung, die Leistung von Modellen, die mit ostarmenischen Daten trainiert wurden, bei der Übersetzung in/aus Westarmenisch und die Möglichkeit der Nutzung solcher Modelle und verfügbarer ostarmenischer Paralleldaten zur Leistungssteigerung.

Mit dieser Arbeit stellen wir die westarmenische Sprache der Forschungsgemeinschaft vor und stellen Software für Menschen auf der ganzen Welt zur Verfügung. Wir hoffen, dass diese Arbeit ein bescheidener Ausgangspunkt für die Erforschung der natürlichen Sprachverarbeitung für die westarmenische Sprache, für ihre Modernisierung, Digitalisierung und eventuell für die Verlangsamung oder sogar Beseitigung der Gefahr des Aussterbens sein kann.

Contents

List of Figures

List of Tables

Dedication

The milestones like a thesis remind you what a long and strange trip my academic career has been. Even with the doubts, the breakdowns, and the homesickness, I am glad to say "We've arrived". I especially wanted to say this to my father, who was always there even when the lights dimmed. He was a visionary and a true source of inspiration in the industry. He showed me how to be professional and cordial at the same time. I deeply regret that he is not able to see the day of graduation of his only son. Therefore I dedicate this thesis to the memory of my father, Krikor Boyacıoğlu, who had always believed in my success. May God rest His soul.

This work is also dedicated to my loving mother Arpi. She raised me with the utmost care and love and has always supported me both on sunny and cloudy days. She strives for my best and is not afraid to get her hands dirty. She is one of a kind and the most resilient person that I have ever seen.

Additionally, I dedicate this work to my maternal uncle Hagop, who taught me to be selfless and compassionate to others. To my paternal uncle Levon, who always listens before talking and gives great advice. To my confidants Berkay, Ildi, Batuhan, Yiğit, Aren, and Ozan for making the ride more enjoyable and less wobbly. To my friends back home Aksel, Deniz, Cengiz, Şant, Can, and Ayk for helping me to cope with my homesickness. To the respectable gentleman, Mr. Barbaros Demiralp, who most sincerely supported me through the tougher times, especially after the loss of my father. Last but not least, to the Armenians living in the diaspora who are trying to keep the language alive. Thank you for your dedication.

Acknowledgements

Ever since I have come across natural language processing and machine translation through lectures, I wondered why there has not been anything done for my mother tongue Western Armenian, a language with a rich history threatened with extinction. It is nothing but heartbreaking to see a language slowly fade away from life and reside only in books. Therefore I am deeply grateful to my supervisor Prof. Dr. Jan Niehues for giving me the opportunity to fight back; for his immense vision and insights when things got tricky. Additionally, this work would not be possible without the necessary resources for model training provided by the Artificial Intelligence for Language Technologies (AI4LT) lab at the Institute of Anthropomatics and Robotics (IAR) of Karlsruhe Institute of Technology. Another special thanks to Mr. Sai Koneru for his assistance during setup and for clarifying any problems regarding hardware.

Finding the necessary sources for the parallel corpus was quite a bit of a challenge itself and on that occasion, I thank Mrs. Linda Gülbağ of the Turkish Armenian Minority Schools Teachers Foundation (Թրքահայ Ուսուցչաց Հիմնարկ) and Dr. Jesse Arlen of Krikor and Clara Zohrab Information Center for their guidance during the search for printed resources; Dr. Victoria Khurshudyan of INALCO and Dr. Hossep Dolatian of Stony Brook University for giving insights about the DALiH project and online Western Armenian resources; Mr. Chahan Vidal-Gorène of Calfa for providing OCR resources supporting the Armenian script; finally Ms. Shogher Margossian and the Calouste Gulbenkian Foundation for supporting the project.

I thank once again all those who had even the slightest contribution to this work and plead for their understanding if I did not mention their name exclusively.

1. Introduction

Languages are the essential tools to convey information ranging from the most basic observation to the thoughts, emotions, and various concepts of the highest intricacy. By communicating with each other using languages, humans were able to perform actions that require planned movement, construct objects or structures that require coordination, or discuss what things are, how things are valued, etc.; each of which brings up new information in the forms of discoveries or experiences while also more questions about the concept itself. This information -with the help of writing- eventually adds up to shape a base of knowledge that is passed through a multitude of generations. This is the pedestal of building the greatest civilizations.

One might wonder why there are so many different languages in the world and whether it is more or less convenient and efficient to gather all the available knowledge of humankind in one single language. The well-known story of the Tower of Babel shows that the multiplicity of human languages was a phenomenon that has been worthy of discussion since ancient times. Evans and Levinson postulate that the diversity of languages is based on two patterns: phylogenetic (cultural-historical) and geographical patterns. Meanwhile, they mention that the human language should be modeled with a coevolutionary perspective, that is considering the biological constraints and the cultural-historical traditions [50].

Languages and cultures have a codependency. Culture is the shaper of a language. Members of diverse cultures live in various parts of the world. They have different lifestyles and their experiences with the environment are unique. Culture is the main aspect of a language's vocabulary (both on a macro and micro scale), its phrases and expressions as these usually reflect on the beliefs, customs, and traditions of a community. Language serves as the envoy of a culture, not only when members of different communities are exposed to it, but also through time, an old language or a language's old form may shed light on the culture of a community in the past.

Like cultures, each language comes with its quirks and peculiarities. These allow them to capture the reality from a unique perspective. There are 7168 different languages spoken by approximately 7.67 billion people according to Ethnologue [49]. Languages can die and go extinct if no speaker of the language is left in the world. This is not only a loss for the whole of humanity, as one of the intangible perspectives to the world ceases to exist, but it is also tragic to see the whole process, the once thriving and productive community first starts to shrink due to various reasons and becomes unable to produce sufficient content for its community, the younger generation deliberately chooses not to learn their mother tongue, and thus the language succumbs to extinction as the last speaker passes away. Although the extinction of a language has happened numerous times throughout history, languages are now facing extinction at a higher rate due to globalization and mass waves of migration within the last centuries. With exposure to

the Internet and its common language, English, young folk throughout the world tend to switch from their mother tongue to more commonly used languages to keep up with the globe. In the introductory part of the Cambridge Handbook of Endangered Languages, the editors Austin and Sallabank predict that by the year 2100 between 50% and 90% of current languages will be severely endangered or dead [12].

Western Armenian (ISO 639-3 Code: hyw) is one of the two standardized forms of the Armenian language. While the other form, Eastern Armenian is the official language of Armenia, Western Armenian is spoken by various communities in the Armenian diaspora, including the United States, France, Turkey, Lebanon, and Syria. Due to its diasporic nature and lack of an official representation, Western Armenian is facing challenges in terms of preservation and continuation. In their Atlas of the World's Languages in Danger, UNESCO classifies Western Armenian as a "definitively endangered" language [138], which means the younger generation of the language's community does not prefer to speak the language and the speakers are mainly from the parental generation or older [47]. Additionally, Western Armenian speakers in different countries interact and borrow words from the majority languages of their host countries, which brings more division and complexity when two Western Armenian speakers from different countries try to communicate with each other.

This work is an attempt to help preserve Western Armenian and establish its existence on the internet with the knowledge and resources brought by machine translation and the natural language processing (NLP) research community. The end-product is aimed to serve not only as another contact point for the younger generations of the Armenian diaspora to reconnect with Western Armenian, but also as a tool for individuals who want to learn Western Armenian language or culture, or simply immerse themselves in Western Armenian literature. English was the first choice to translate texts from/to, in order to reach the widest audience possible due to the vast English resources on the internet. Low-resource languages make up a hot research area in the machine translation community, and investigating methods for the improvement of low-resource language translation quality such as Western Armenian provides insight into both the methods and the language.

As of the present study, there are no other works in machine translation or in the natural language processing field concerning Western Armenian. The reason could be traced back to several factors which will be discussed in further chapters, however, we hope this work could wake the researchers' interest and provide some resources for research.

1.1. Problem Statement & Research Questions

Translating content between languages can be compared to building a road between two towns. The road enriches the two communities both culturally and economically. Similar results could also be observed when translating a broad array of works back and forth. Economically, both communities can understand each other much better, strengthening their ties and allowing them to build partnerships; culturally, communities start to see each other's perspective and people enrich themselves by including aspects from the new perspective. In the case of English and Western Armenian, there is no

exception. Translating English content, the lingua franca of the world, Western Armenian could be brought into the modern era, especially in the domains of technology. Besides, Western Armenian could definitely enjoy the almost endless corpus of English and enrich themselves during the process of it. English on the other hand could broaden its collection with scriptures of classic and modern Western Armenian literature.

The main goal of this work is to build up that bridge using modern technologies from the field of Linguistics and Computer Science. Stating formally, ***to build up a neural machine translation model for the language pair of English and Western Armenian, that is capable of translating in both directions with a certain level of quality***.

To introduce the research community to Western Armenian, we also aim ***to create the first (parallel) corpus of Western Armenian for natural language processing research***. In order to reach the widest amount of resources with a single step, the parallel corpus for Western Armenian will be built with an English pair. We will share the parts of the (parallel) corpus which are not subjected to any kind of copyright.

While focusing on the main goals, this work also tries to investigate some research questions on the improvement of the translation quality:

RQ1: *Which known methods used in data collection, preprocessing, or implementation of the machine translation model could be used to improve the translation quality to/from Western Armenian?*

There is a plethora of methods that aim to enhance the translation quality for neural machine translation models. The methods could impact various parts of the whole building process of the model, e.g., data collection, preprocessing, model architecture, training, postprocessing, etc. We want to try out some known methods and find out the individual impact on the translation quality.

RQ2: *How do (multilingual) machine translation models trained on Eastern Armenian data perform while translating to/from Western Armenian?*

Since there are already a few neural machine translation models trained on Eastern Armenian, we want to test their performance on Western Armenian data.

RQ3: *How can we utilize data/knowledge of Eastern Armenian when training for Western Armenian?*

Western Armenian content creation and organization for research heavily depends on voluntary work, due to the lack of official representation of Western Armenian in any country and to the slow adaptation of communal institutions in the Armenian Diaspora to recent developments in technology. Western Armenian has a relatively rich literature, however, most of it is yet to be digitized. Thus, it is nothing but reasonable to search for Eastern Armenian resources since it enjoys proper interest and care in terms of NLP research. Both languages are standardized variants of Modern Armenian and share a substantial amount of vocabulary and a set of grammatical features, therefore it is assumed that some subset of information regarding translation to/from Western Armenian can be learned from Eastern Armenian. We will investigate if this is the case and to what extent it applies.

1.2. Thesis Outline

The next chapter focuses on the historical development of Armenian and machine translation. In the former part, the schism and the individual development of both variants are described and concluded by a brief comparison of the variants. The latter part is shaped around the transformer model, the most prominent architecture for neural machine translation, whose individual modules as well as its training, inference, and evaluation processes are explained. Then the concept of a low-resource language is introduced and the problems caused by low-resource languages in machine translation and some handling techniques are investigated.

Chapter 2 analyzes the existing natural language processing research regarding Eastern and Western Armenian by individually focusing on available data and tools. Then, the baseline model of "No Language Left Behind" (NLLB) [133] is placed under the investigative light, with an extensive analysis of how the data was shaped and the whole model was built.

Chapter 3 explains the data collection/preprocessing pipeline step by step. Additionally, each individual subset of Western Armenian-English parallel, Western Armenian monolingual are showcased with their individual background information; as well as the statistics about the Eastern Armenian-English corpus of OPUS [135].

In chapter 5, each individual experimental scenario is presented along with its corresponding models. This chapter also includes the choice of default values for training parameters, which are exclusively listed in Appendix A.

The results of each scenario are presented and evaluated in chapter 6, where each scenario is investigated in different settings of test sets.

Finally in chapter 7 the findings are summarized and brought into a form that serves as the answers to the research questions which is followed by a brief critical analysis of the work to point out the shortcomings and possible improvements.

2. Background

This chapter is designated to lay up a basic framework and to provide reference to the concepts explained in the further chapters.

First, the reader is introduced in the first section to the Armenian language via its historical development and then to its modern standards by a brief comparison. The second section focuses on the concepts of (neural) machine translation, how the task is modeled, which modules are employed in various models including the most popular Transformer model architecture, and how the models are trained and evaluated.

2.1. The Armenian Language

Armenian is a language in the Indo-European language family with its isolated branch. The language captivates the linguists' attention due to the complex relationship with its neighboring languages. The origin of the language is subject to many discussions and hypotheses. Linguists have investigated Armenian's connections with other Aryan and Balto-Slavic languages as well as Greek [66, 110, 103, 111, 69, 37]. According to Martirosyan, "Armenian is usually placed between Indo-Iranian to the east and Greek to the west, and on the northern side it might neighbor Balto-Slavic (and/or Germanic and others)" [102]. Donabedian-Demopoulos mentions that Armenian emerged after the intermixture of Proto-Armenian, a dialect of Proto-Indo-European, with the extinct Urartian language [45].

Armenian has its own script (see Table 2.1), introduced by Mesrop Mashdots and Sahak Bartev (Isaac of Armenia) in 405 AD which followed by a century of high activity in the field of scripture and literature. Also called the "Golden Century of Armenian Literature", many classic-era texts, including the Bible as well as various religious, historical, ethnographical, philosophical, linguistic, and socio-geographical texts were translated either from the original language or from Greek, the lingua franca of the 5^{th} century AD, creating the Classical Armenian language or "Grabar" (literally: written, literary language).

The evolution of the Armenian language can be divided into three periods: Classical Armenian (5^{th}-10^{th} century), Middle Armenian (11^{th}- 17^{th} century), and Modern Armenian (18^{th} century onwards). Classical Armenian was used as a spoken language until the 11^{th} century and continued to be the literary language until the emergence of Modern Armenian variants. It is still used as the liturgical language of both the Armenian Apostolic and the Armenian Catholic churches. Middle Armenian emerged in the 11^{th} century in the region of Cilicia (modern-day southern Turkey) after the foundation of the Armenian Kingdom of Cilicia. As a new administrative language, Middle Armenian distances itself from the rigid Classical Armenian by employing some modernizations and adaptations from the spoken language. It also includes new words from colloquial Armenian as well as

Capital / Minuscule	Romanization	Pronunciation	Capital / Minuscule	Romanization	Pronunciation
Ա / ա	a	t**i**me	Ն / ն	n	**n**ew
Բ / բ	p	**p**en	Շ / շ	sh	**sh**op
Գ / գ	k	**c**ome	Ո / ո	o/vo	g**o** / **vo**ice
Դ / դ	t	**t**ime	Չ / չ	ch	**ch**eese
Ե / ե	e / ye	c**a**t / **ye**s	Պ / պ	b	**b**ell
Զ / զ	z	**z**one	Ջ / ջ	ch’	cat**ch**
Է / է	ē	**e**lle (Fr.)	Ռ / ռ	rr	**r**un (rolled)
Ը / ը	ə	nugg**e**t	Ս / ս	s	**s**ee
Թ / թ	t’	**t**in	Վ / վ	v	**v**et
Ժ / ժ	zh	mea**s**ure	Տ / տ	d	**d**ad
Ի / ի	i	m**ee**t	Ր / ր	r	**r**ace
Լ / լ	l	**l**emon	Ց / ց	ts’	**ts**unami or **Z**ahn (Ger.)
Խ / խ	kh	Da**ch** (Ger.)	Ւ / ւ	w	ali**v**e
Ծ / ծ	dz	han**ds** up	Փ / փ	p’	**p**aragraph
Կ / կ	g	**g**reen	Ք / ք	q	bu**ck**
Հ / հ	h	**h**ouse	Օ / օ	o	b**o**ss
Ձ / ձ	ts	ba**ts**	Ֆ / ֆ	f	**f**oot
Ղ / ղ	gh	**R**eis (Ger.)	ՈՒ / ու	u	m**oo**se
Ճ / ճ	j	**j**ump	Ւ՚ / ւ՚	ü	**ü**ber (Ger.)
Մ / մ	m	**m**om	Է՚Օ / է՚օ	ö	**Ö**l (Ger.)
Յ / յ	y/h	**h**ello / voice			

Table 2.1.: The Armenian Alphabet with Western Romanization and Approximate Pronunciation

translations or direct borrowings from neighboring languages to articulate more modern concepts of the Middle Ages. This shift also allowed new themes to appear in Middle Armenian literature. Some distinct Modern Western Armenian features could be traced back to Middle Armenian.

Unlike Classical Armenian, which was frozen in time, the influence of spoken Armenian has grown in Middle Armenian. Inspired by the ideas of the Age of Enlightenment, the intellectuals of several Armenian communities in the Ottoman and the Russian Empire agreed upon the standardization of spoken Armenian or “ashkharabar” (literally: language of the world), which had numerous dialects across Anatolia and the Armenian Highlands. In his work, Adjarian [2] identifies the dialects and the languages spoken as the mother tongue of various Armenian communities in 1909 and shows them as a map (Figure 2.1). Notice that not all Armenian communities speak Armenian as their primary language. The "gë/gə" dialects are closer to the Modern Standard Western Armenian while the "um" and "el" dialects are to the Modern Standard Eastern Armenian.

The emergence of Western and Eastern variants of Modern Armenian is a result of the socio-politic situation of the 18^{th} and 19^{th} century, as Armenians had two cultural capitals in both Ottoman and Russian Empire: Constantinople (Istanbul) and Tiflis (Tbilisi). The standardization of spoken Armenian has developed independently in both empires, having the dialect of the capital as the standard for each variant. This movement resulted in major educational campaigns during the 19th century throughout various Armenian communities of both large cities and of countryside which were led by writers, intellectuals, and other individuals of influence. A tragic end was waiting for these folk in the Ottoman Empire, who were arrested and murdered on the 24^{th} of April 1915 followed by the displacement and the destruction of Armenian communities across the Ottoman Empire. These events were recognized as the Armenian genocide by 34 countries as of 2023 and by the academic

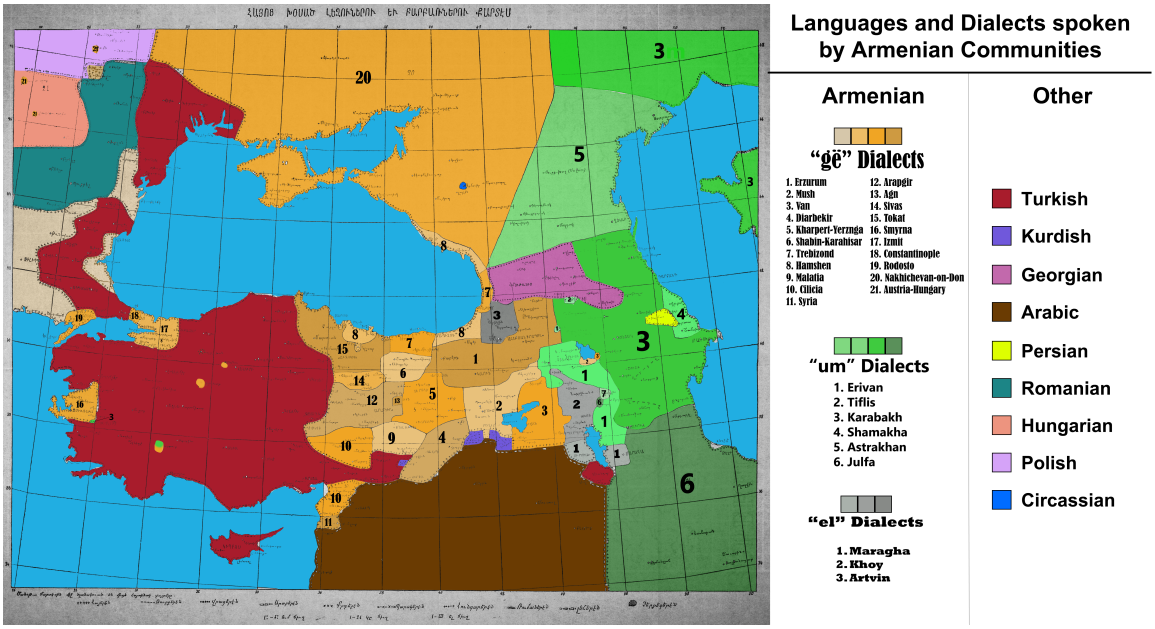

Figure 2.1.: Languages and Dialects spoken by Armenian Communities [2] (colorized & annotated)

authorities. After 1915 not only most of the Western dialects of Modern Armenian vanished, but also the continuation and development of Western Armenian became critical due to the collapse of the community and its institutions and organizations. Modern Eastern Armenian on the other hand was the official language of the First Republic of Armenia and then of the Armenian Soviet Socialist Republic and has been able to maintain its institutions and other bodies of literature which made language develop naturally. Today, the word "Armenian language" usually refers to the Eastern variant since it is recognized as the official language of Armenia.

The two standardized forms of Modern Armenian are mostly mutually intelligible [45, 29], although speakers of one variant may need to adapt themselves in both written and spoken form of the other variant, because of the difference in grammatical case suffixes, of phonetic differences, of different orthographies (Western Armenian has kept the classical orthography, while Eastern Armenian has adapted the reformed orthography of 1920s.) and of the different borrowings for the same word in each variant (most commonly from Russian in the case of Eastern Armenian and from Arabic, French, English and Turkish for the case of Western Armenian). There are also more recent cases where speakers of different variants have come across obstacles while trying to understand the other variant [78].

In 2017 the request to identify the Western and Eastern Armenian dialects as separate languages was accepted by SIL International ISO 639-3 Registration Authority [127]. ISO 639-3 is the standard to encode every human language in history with a three-letter code. According to the report [127], the languages' linguistic distance is not great, but having developed distinct vocabularies and literature is the evidence for the emergence of two separate languages. Thus, the two dialects are identified as separate languages each with

its own code: hyw for Western and hye for Eastern Armenian. Additionally, the code hyx is assigned to identify the Armenian language family.

According to Ethnologue's statistics [49], as of 2022 there are approximately 1.6 million Western Armenian speakers in the world. The speakers mostly reside in the Middle East (Lebanon, Syria, Turkey, Jordan, Iraq, Iran and others), Northern and Southern America (the US, Canada, Argentina, Brazil, Uruguay), Europe (France, Italy, Greece, Bulgaria, Romania, Poland, the UK, Germany) and Australia. Ethnologue's statistics for Eastern Armenian speakers date back to 2013 with 3.8 million speakers worldwide [49], however today it is estimated that this number has gone up to 6-7 million.

The original Armenian alphabet consists of 36 letters, during the Middle Armenian era two more letters were included as a modernization attempt (օ and ֆ). Three additional sounds are represented by two letters (ու, իւ and էօ). There is a difference in the pronunciation of the letters in each modern variant. Table 2.1 shows the modern Armenian alphabet with its Western romanization and pronunciation, which will be used further in this document.

2.1.1. A Brief Comparison of the Modern Variants

The main differences between Modern Standard Western and Eastern Armenian through some examples shown on Table 2.2. The differences can be divided into three categories: 1) Phonological differences, meaning the word with the same spelling is pronounced differently in the standards. 2) Orthographical differences, meaning the same word is spelled differently. 3) Grammatical/Syntactical differences, meaning the usage of different morphemes for the same semantical meaning. The morpheme used in one variant can have a different meaning in the other standard or can be completely unused. Additionally the standards loan words from different sources, for Eastern Armenian being Russian and for Western Armenian being English, French, Arabic, Turkish, etc. which depends on the residence of the speaker.

	English	Western Armenian (Pronunciation w/ IPA)	Eastern Armenian (Pronunciation w/ IPA)
Phonological	Hello	Բարեւ (pʰarɛv)	Բարեւ (barɛv)
	Library	Գրադուն (kʰradun)	Գրադուն (ɡradun)
Orthographical	Data	Տուեալ (dvjal)	Տվյալ (tvjal)
	Relationship	Յարաբերութիւն (harabʰɛrutʰjun)	Հարաբերություն (harabɛrutʰjun)
Grammatical / Syntactical	I do not understand where these birds came from	Ես չեմ հասկնար, թէ ուրկէ եկած են այս թռչունները: (jɛs ʧʰɛm hasɡnar tʰɛ urɡɛ (j)ɛɡadz ɛn ajs tʰɾɟʰunnɛɾə)	Ես չեմ հասկանում, թե որտեղից են եկել այս թռչունները: (jɛs ʧʰɛm haskanum, tʰɛ vortɛɣitsʰ ɛn (j)ɛkɛl ajs tʰɾɟʰunnɛɾə)
	They were going to play today.	Անոնք այսօր պիտի խաղային: (anonkʰ ajsor bidi χaʁajin)	Նրանք խաղալու էին այսօր: (nrankʰ χaʁalu ɛin ajsor)

Table 2.2.: Examples showing differences between Western and Eastern Armenian

2.2. Machine Translation

2.2.1. History

Machine translation is the task of translating input, which is mostly in the form of text or speech, from a source language to a target language using computational algorithms.

Machine translation is one of the first fields that has interested computer scientists since the mid-1940s. The scientists, mostly coming from a background of decoding military messages during World War II, believed that machine translation should not cause much trouble since languages are simply different encodings of the idea that is being conveyed. Warren Weaver addresses this in his letter in 1947: “When I look at an article in Russian, I say "This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode.” [145]

The approach for handling machine translation task has gone into some paradigm shifts during the course of history. The first approach, Rule-based Machine Translation involves analyzing the text in the source language, transforming it into an intermediate form, and generating text in the target language from the intermediate form. The methods under this approach differ mostly on the depth of analysis, transfer, and generation steps. Rule-based Machine Translation systems include source and target language parsers for reading and writing texts, morphological analyzer/generator for source and target language to convert the input to the intermediate form and generate from it, and a set of rules and dictionaries to translate from source to target language’s intermediate form. The quality of translations was mostly dependent on the granularity and coverage of the rules. Rule-based Machine Translation is mostly limited to certain domains since working with larger domains will require a substantial amount of handcrafted rules. Although it did not require large parallel corpora for training, obtaining a good set of rules was usually too challenging. Examples of Rule-Based machine translation systems include METEO [134] which specializes in translating weather forecasts between English and French, and Apertium [53] a free and open-source software that can translate between multiple language pairs.

With the rise of the internet and the availability of large text databases, or corpora, the paradigm in machine translation started to shift towards statistical methods during the 1990s. Instead of relying on rules and dictionaries for translation Statistical Machine Translation [26] extracts translation rules in the form of patterns and learns to extract patterns in source and target language texts and to find the corresponding pairs. Coarsely, Statistical Machine Translation systems calculate the probability of a sentence being a sentence that is generated by a native speaker for each language and the probability of a word sequence in the target language to be the translation of the word sequence in the source language input. The system’s aim is to find a probability distribution for each calculation that maximizes the probability of the correct translations. To be able to do that Statistical Machine Translation systems make use of parallel corpora, i.e., direct translations of the same document. Learning directly from the corpora, machine translation systems have become more adaptable to the flexible nature of languages. However, probabilities tend to approach zero when sequences get longer, and each part of the sequence has a great number of outcomes. This is the case for Machine translation since sentences and vocabulary of a language have no limit. Additionally, human languages utilize anaphora/co-references to avoid repetition, however by doing that they also bring ambiguity which is challenging for computers to detect.

Late 2000s and 2010s have brought better and stronger hardware which allowed computers to compute more complex tasks and calculations with larger numbers. Such a task was the realization of Artificial Neural Networks, which were modeled in the 1950s [85, 119]. This has boosted the size of the Artificial Intelligence research community

with the interest of finding new Deep Learning methodologies and new neural network architectures specializing in different tasks including Machine Translation. The current paradigm is called Neural Machine Translation, however, rather than being a completely new paradigm, it is an extension of Statistical Machine Translation since neural models still try to find the same probability distributions, but the utilization of specialized neural network architectures has mitigated the shortcomings of standard statistical models. Currently, Neural Machine Translation is a hot topic in the field of Natural Language Processing and has already an effect on companies and individuals across the world with software that can translate multiple languages. These include Google Translate [56], DeepL Translator [41] and OpenNMT [86].

2.2.2. Neural Machine Translation

2.2.2.1. Modelling the Task

Examples are the typical material, from which neural models learn to fulfill a specific task. In the case of machine translation, the learning material is the parallel corpora. These are sets of translated documents whose sentences or paragraphs are usually aligned with their counterparts in the paired document. Examples of such corpora include the Rosetta Stone, a stele on which a decree on behalf of Ptolemy V is written in Ancient Egyptian (in hieroglyphic and Demotic script) and in Ancient Greek; or its modern counterpart the documents of European Parliament which have translations of all languages in the European Union.

Texts can be considered as sequences. The task of translation is to change the units in the source language sequence to their target language counterparts. The unit of the sequence may depend on the granularity. Therefore, texts may be regarded as a sequence of characters, (sub)words, sentences, paragraphs, chapters, etc. Typically, in neural machine translation texts are translated on the sentence level, in other words, the given input for neural machine translation models is a sentence which is considered as a sequence of words. Thus, the translation task may be modeled as a sequence-to-sequence problem. Given an input sequence $X = (x_1, x_2, \ldots, x_M)$ find an output sequence $Y = (y_1, y_2, \ldots, y_N)$ which maximizes the conditional probability $P(Y|X)$ or: $\hat{Y} = argmax_Y P(Y|X)$. Unlike other traditional neural network models, sequence-to-sequence models accept and generate inputs and outputs of variable lengths.

The translation task fits perfectly to this formulation; as the source language sentence is a sequence of source language words as input, i.e. $S = (s_1, s_2, s_3, \ldots, s_K)$; the output is given as a sequence of target language words or a target language sentence, i. e. $T = (t_1, t_2, t_3, \ldots, t_L)$; and the conditional probability of T being the translation of S in target language; given the source language sentence S, i.e. $P(T \text{ is the translation of } S|S)$. The neural model tries to find a probability distribution that fulfills the translation task, i.e., the conditional probability for any given source language sentence. The probability distribution is typically parametrized with θ, whose values are learned by the model through exposure to example sentence pairs during the training.

2.2.2.2. Word Representations

(Deep) Neural Networks are known for their hierarchical feature extraction approach while trying to learn a task. The first layers focus on the local features in data, while the deeper ones on the global features which are aggregated from the local features. The classic example is the image recognition task, where the first layers extract easier shapes like circles or edges, while deeper layers try to find more complex shapes like a face. Since the neural network requires working directly with the data, it must be brought to a suitable representation so that the network can make the calculations during training. Again from image recognition, the images can be represented with a set of RGB values for each pixel in the image. The same also should be done to text input. Perhaps the most straightforward approach would be assigning each word appearing in the corpus an index number (i.e. building the vocabulary of the corpus) and thus representing each word as a one-hot vector, that is, a vector with a dimension size of the total number of words (or vocabulary size) whose entries are all zeroes except the dimension whose index number is equal to the index number of the word. The drawback against the simplicity of this idea lies in the scalability, as the vocabulary size grows so does the dimension of every calculation, increasing the duration of both training and inference sessions. Thus, the vocabulary of a model is usually kept in a feasible size in order to keep the model functionally desirable. In most of the cases, the words that are left out are the rare ones. If a rare word appears during training or inference, it cannot be represented and thus the translation quality of the model is negatively affected. This problem is called the Out-Of-Vocabulary Problem and it is one of the most extensively researched areas in the field of Natural Language Processing as any kind of improvement will bring better models for any kind of task in the field. Some workarounds and improvements will be presented in subsection 2.2.4.

Coming back to the word representations, one-hot embeddings are not the only way to represent words in neural networks. Unsupervised word representation [27, 107, 92] is an approach that uses unsupervised learning methods such as clustering to represent multiple words in word representative which effectively makes the vocabulary size smaller. Another unsupervised approach is to use Neural Language Models on the corpus and use the resulting embeddings as the representations of words. Language Modeling is another task from NLP in which basically a given sentence is assigned to a probability of being a sequence of words that is produced by a native speaker of the language that the model is dedicated to. This task can also be learned by Neural Networks [106]. During training the learned word representations are denser and capture the syntactic and semantic relationship of words. The words that have such a close relationship appear near to each other in vector space[105, 104]. This is an improvement to the one-hot embeddings with two aspects; first, the word representations from the language modeling task have fewer dimensions, additionally, they capture more information about the word itself and possibly about other words. Most state-of-the-art neural machine translation models and models of other NLP tasks use pretrained word representations. This allows transferring knowledge gained by one model to another, e.g. during training instead of solely relying on the vocabulary of the dedicated corpora using pretrained word representations capture all the words of the language model. Examples of such word representations or embeddings are Word2Vec [105, 104], GloVe [70], BERT [42] and fastText [24].

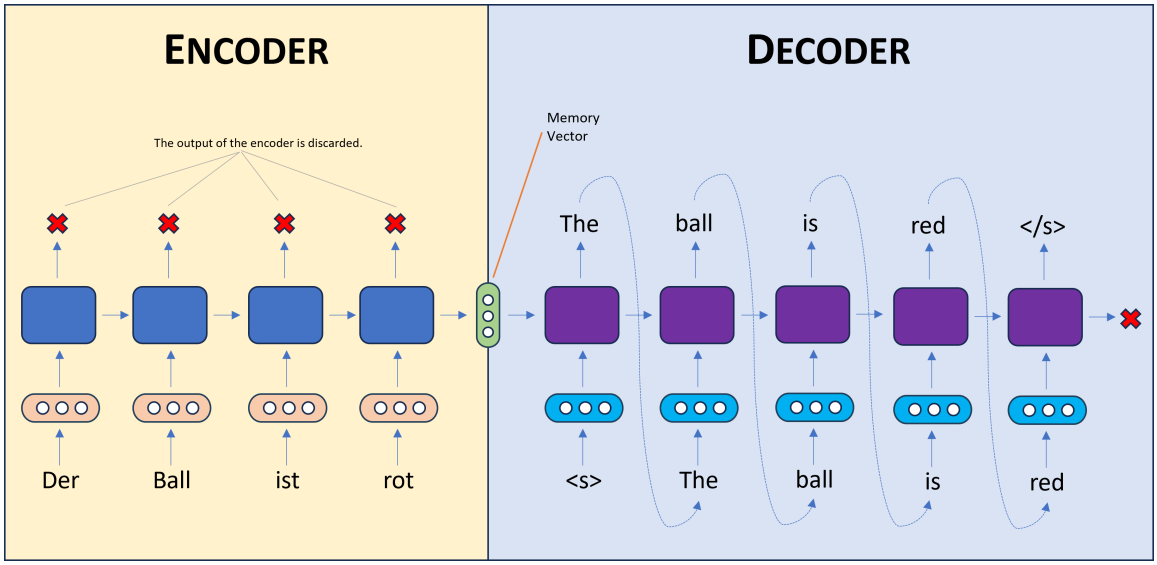

Figure 2.2.: Encoder-Decoder Architecture

2.2.2.3. Architectures and Components

Along with other sequence-to-sequence tasks, neural machine translation is modeled by the Encoder-Decoder neural architecture [32], which consists of two main parts as seen on Figure 2.2. The Encoder takes units of a sequence of variable length one after another, thereby encoding each unit's (token's) representation into a single vector with a fixed length, which is also called the memory vector. In other words, the memory vector is the vectorial representation of the input sentence. The memory vector is then fed into the decoder, where each token in the target language is generated sequentially. In each step, the memory vector and the last generated token are given to the Decoder as the input, and a new token is generated which will be used as the input for the next step. To generate the first token in the target sentence, the start-of-sentence token (`<s>`) is given along with the memory vector. The generation of tokens continues until the model generates the end-of-sentence token (`</s>`).

Recurrent Neural Networks with Long Short-Term Memory The first proposed network to implement the Encoder-Decoder architecture is the Recurrent Neural Networks (RNNs) with Long Short-Term Memory (LSTM) [64]. The RNN contains a hidden state and accepts a sequence of variable length $X = (x_1, x_2, \ldots, x_N)$. Each unit is given into the network at a time step, during which the hidden state is updated, this is formally described as:

$$h_t = f(h_{t-1}, x_t), \tag{2.1}$$

where h_{t-1} and h_t are the hidden states of the time steps t and $t-1$; x_t is the t-th component of the input sequence and f is a non-linear activation function. Using traditional activation functions like the sigmoid function has proved to be ineffective for sequence-to-sequence learning tasks since longer sequences result in more hidden states, and during training

calculating each hidden state's contribution to the error results in either too large or too small values. This is also called the exploding/vanishing gradient problem [63].

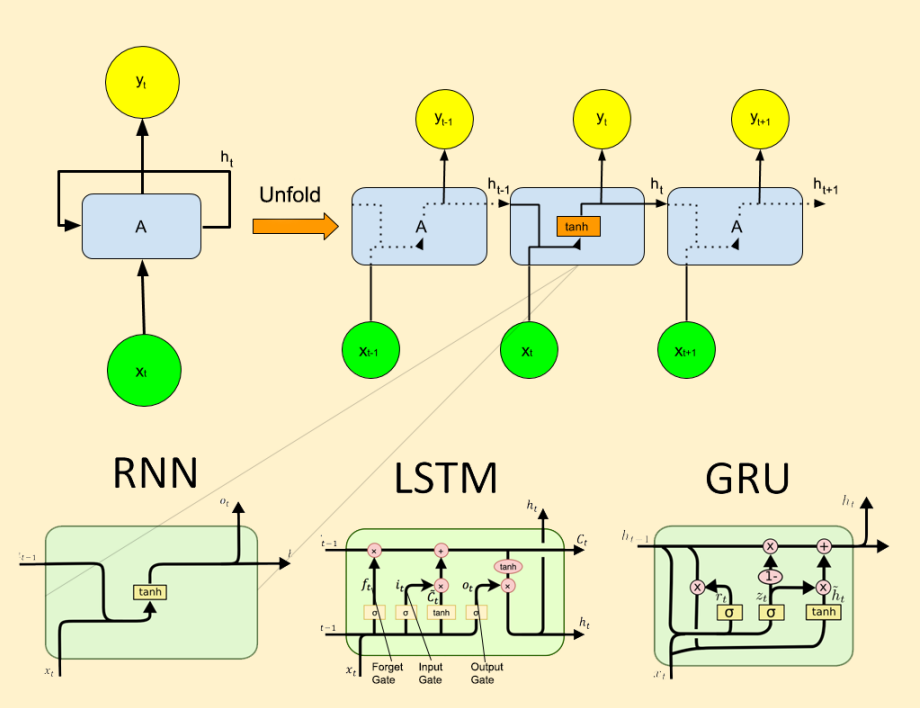

Figure 2.3.: Concept of RNN, LSTM, and GRU. From[1] [2] (Combined and modified)

The Long Short-Term Memory is one of the first proposed network units that addresses the vanishing gradient problem. Its activation function introduces three functions, or gates: the forget gate controls how much of the information of the previous hidden state to discard or to forget; the input gate controls how much new information to include in the new state; and the output gate controls how much of the information of the new state to output. The intuition behind the LSTM is to make the network able to learn which information to keep in the sequence for future use and when to forget information that is no longer needed. One downside of the LSTMs is the introduction of additional parameters for each gate in every time step. To mitigate this, Gated Recurrent Units (GRU) [32] were introduced as an alternative, which work similarly to LSTMs but do not include the output gate. Figure 2.3 shows an overview of RNNs, LSTMs, and GRUs.

Attention In Sequence-to-Sequence learning the input and output sequences can be infinitely long. Cho et al. [33] have investigated the relation between sentence length and translation quality. As seen on Figure 2.4, the quality (measured with the BLEU score, see paragraph 2.2.2.6) drops while the sentences get longer. This is due to the fact that the sentence, regardless of its length, is always represented by a single fixed-size vector, also known as the memory vector. Some of the information about the sentence gets overwritten

[1] https://arbu00.blogspot.com/2017/05/3-rnn-recurrent-neural-networks.html
[2] http://dprogrammer.org/rnn-lstm-gru

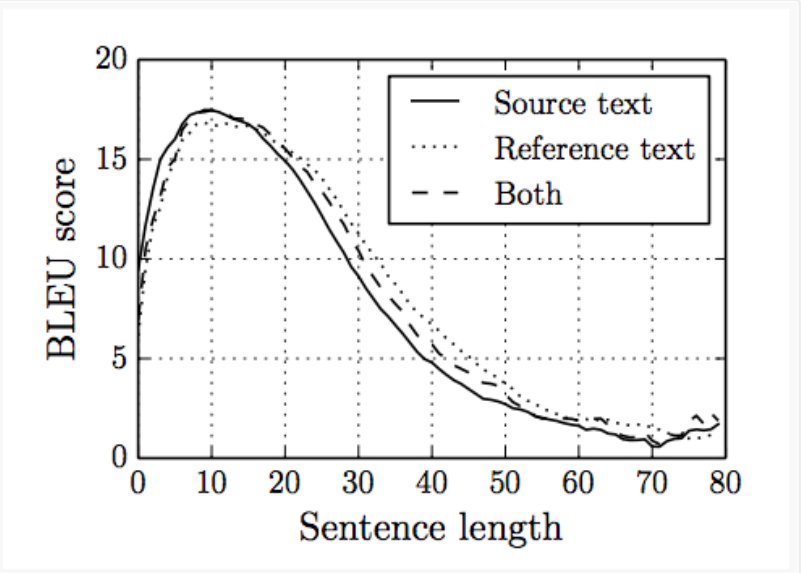

Figure 2.4.: The relation between translation quality (measured with BLEU score) of an RNN-based machine translation and input sentence length [33]

in the memory vector, effectively rendering it to the bottleneck of the Encoder-Decoder model.

The concept of Attention was proposed to address this problem. Used in several different fields of deep learning (such as Computer Vision), attention in the context of machine translation involves learning to align and translate at the same time [18]. In a nutshell, in each decoding step, the model tries to find the most suitable token to translate by comparing each hidden state of the encoder with the current state of the decoder. Formally,

$$\mathbf{e}_i = attention_{net}(y_{i-1}, \mathbf{h}) \in \mathbb{R}^n \text{ where each } e_{ij} = attention_{net}(\mathbf{y}_{i-1}, h_j), \tag{2.2}$$

There are several functions proposed for the similarity score such as Cosine, tanh, and (scaled) dot-product. To be able to learn the alignment of tokens from the data, attention must be represented by a probability distribution of tokens. Thus, the suitableness or the alignment of the states is measured by a similarity score. For each similarity score the softmax function is applied:

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{T_x} exp(e_{ik})} \tag{2.3}$$

α_{ij} is also called the attention weight of the token j in the input sequence calculated for the token that is being processed in the decoder at the timestep i. By multiplying each attention weight α_{ij} with its corresponding hidden encoder state h_j and adding them together yields a version of the memory vector containing all the important information for generating the token for the time step i, also called as the attention output. Precisely:

$$z_i = \sum_{j=1}^{T} \alpha_{ij} \mathbf{h}_j \tag{2.4}$$

Self-Attention By using RNNs the inputs and outputs are processed sequentially, an undesirable aspect for software. Self-attention aims to employ parallelization by encoding each token in the encoder sequence simultaneously while capturing the contextual information or the importance of other tokens in the sequence. Calculating self-attention involves three elements: the query represents the token that will be compared with all keys in the sequence; for each token in the sequence a key-value pair is assigned, the similarity of key and query gives the attention weight of the key for that query. Each attention weight is then multiplied by its key's value and summed together to obtain the attention output for the query. The difference from the attention mentioned above is in self-attention the queries and the keys are from the same sequence, thus each token in the sequence is encoded containing information about all other tokens in the sequence, allowing a better representation of long-range dependencies. The power of parallelization comes with the stacking of the query, key, and values into their respective matrices $Q \in \mathbb{R}^{n \times d}$, $K \in \mathbb{R}^{m \times d}$, $K \in \mathbb{R}^{m \times d}$ and the joint calculation of them. The calculation assumes the vectors have the same dimension d. The scaled dot-product (self)-attention is thereby computed as:

$$attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{2.5}$$

d_k is a scaling factor to keep the calculation numerically stable. The result contains all attention outputs of each key according to each query.

Multi-Headed Attention To be able to capture multiple different patterns in the sequences concerning both semantics and syntax, individual self-attention modules are employed in parallel. Each module or head is then concatenated together and transformed by an output matrix. This is called a multi-head attention module and is formulated as:

$$multihead\_att(Q, K, V) = concat(head_1, \ldots, head_H)W^O, \tag{2.6}$$

where $head_i = attention(QW_i^Q, KW_i^K, VW_i^V)$; and $W_{i...H}^O$, $W_{i...H}^Q$, $W_{i...H}^V$ are weight matrices output, query, key, and values respectively.

Cross-Attention The query, key, and value model can implement the first mentioned attention (see paragraph 2.2.2.3) between encoder and decoder. Called the cross-attention, its queries are generated from the last decoder state and the key-value pairs come from the outputs of the encoder.

Positional Encoding Without recurrent networks, the hidden representations lose sequential information about the input. For example, both the sentences "John loves Mary" and "Mary loves John" have the identical representation. To capture the order of the sequence each token's representation is processed by a positional encoding function. Positional encoders place relative positional information into the embeddings which is useful for attention mechanisms. There are multiple types of functions, both learned and fixed, for positional encoding [73], Transformer uses sine and cosine functions to assign each position in the sequence a unique value, i.e. for each position p two values for positional

encoding are calculated and stored in its representative matrix of size $P \times D$ (P is the sequence length), which is formulated as:

$$PE(p, 2i) = sin\left(\frac{p}{10000^{\frac{2i}{D}}}\right) \quad \text{and} \quad PE(p, 2i+1) = cos\left(\frac{p}{10000^{\frac{2i}{D}}}\right), \tag{2.7}$$

where $0 \leq i < \frac{D}{2}$ is used to map each position to D columns.

The Transformer Architecture The Transformer architecture, proposed by Vaswani et al. [140], eliminates any kind of recurrence and interstate dependency while being capable of accepting and generating inputs and outputs of variable length. Using self-attention as its main encoding and decoding scheme, each token in the sequence given into Transformer obtains a direct connection with any other token in the same sequence, and thus relationships between any two tokens can be better captured. There are three variants of multi-headed attention which are placed in different parts of the transformer: Self-attention is used in the encoder and all its queries, keys, and values come from the embeddings of the input sequence. Cross-attention is used between the encoder and decoder and the query is generated from the last state of the decoder and the key-value pairs come from the encoder. In the decoder, self-attention is used with masks, called masked self-attention, in which the queries, keys, and values of the tokens in the output sequence that are not generated yet are masked. This allows the generation of output sequences whose tokens are dependent on the previously generated output tokens. In other words, in masked attention, the connection between one already generated and one not yet generated token is masked since the output sequence must be generated sequentially.

Figure 2.5 shows the original proposed transformer architecture by Vaswani et al. [140]. The transformer architecture continues to have a separate encoder and decoder part for processing input and output sequences. The encoder consists of $N = 6$ identical layers stacked on top of each other. An encoder layer includes 2 sublayers: A multi-head self-attention with 8 heads and a feed-forward sublayer. Each sublayer's output is merged by a residual connection [61] followed by a layer normalization [15]. A residual connection adds the values of inputs before and after being processed by the sublayer, formally: $x + Sublayer(x)$. By employing residual connections, the embeddings of the first layer are kept the same within the whole network and each (sub)layer learns features by the difference of their given input and output. Layer normalization aims to keep the values of sublayer outputs within a reasonable range and thus also the same range for the probability distributions. Residual connections and layer normalization address the problems of vanishing gradients and the longer time spent in training respectively. The inputs are converted into continuous vectors of fixed dimension called input embeddings, and then positional information of inputs is encoded into the input embeddings with a positional encoding function. Then the embeddings are forwarded deeper into the encoder via multi-head attention and fully connected feed-forward network sublayers. The fully connected feed-forward network is the classic neural network architecture typically comprised of an input, a hidden, and an output layer and each node within a layer has all possible connections with a node of a neighboring layer. The output of each

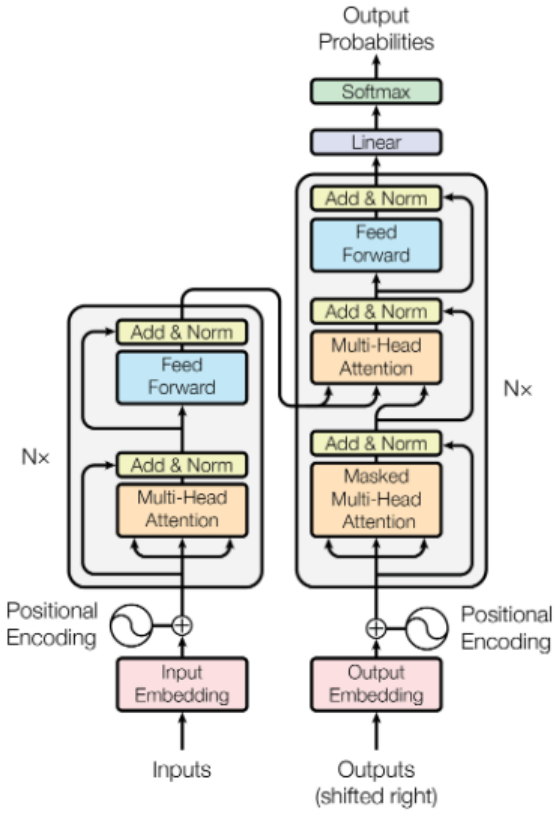

Figure 2.5.: Transformer architecture proposed by Vaswani et al. [140]

encoder layer is then fed into the cross-attention sublayer of the decoder with the same depth as the encoder.

The decoder also has a stack of $N = 6$ identical layers. A decoder layer consists of three sublayers and each layer is coupled with a residual connection and a layer normalization. The decoder generates outputs token by token in an autoregressive manner and each output is fed as an input to generate the next token in the sequence. After converting the token into a continuous representation and injecting it with positional information, the token is given into a masked multi-head attention layer, followed by a multi-head cross attention and a fully connected feed-forward network. The output is then projected to the vocabulary space by a linear layer, followed by a softmax layer to generate probabilities.

The Transformer architecture revolutionized not only machine translation along with other natural language processing tasks but also it is widely used in computer vision and other machine learning fields. Many state-of-the-art models in natural language processing are based on the transformer architecture such as GPT-based models [114] and BERT [42].

2.2.2.4. Training

In the case of supervised learning, the neural model learns the assigned task by observing a great amount of data entries, each of which is paired with a label. In each training step, the model predicts the label for the given entry and the prediction is then compared with

the real label or the ground truth. The comparison is realized by an objective or a loss function. After the calculation of the loss, the contribution for the loss is measured for each parameter within the model and they get updated according to the contribution.

In Neural Machine Translation, the loss is usually given in the form of Cross Entropy or Negative Log-Likelihood function. The model tries to adjust its parameters in such a way that the cumulative negative log-likelihood loss for each entry in data is minimized. The cumulative loss is calculated as:

$$NLL(D) = -\sum_{i=0}^{|D|}\sum_{j=0}^{n} log(P(\hat{y}_j^{(i)}|x^{(i)})), \tag{2.8}$$

where D represents the given data to the model for training, n is the length of the output sequence which depends on the input sequence and therefore is not fixed, $P(\hat{y}_j^{(i)}|x^{(i)})$ is the probability of jth token in the output sequence of the ith data entry being the target token, i.e jth token in the target language sequence of the ith data entry. The probabilities for each position are obtained from the respective softmax layer of the hidden decoder states.

To update the parameters from the loss; the gradient descent and the backpropagation [120] algorithms are employed. During backpropagation, each parameter's contribution to the loss is calculated by using the chain rule of calculus. Gradient descent finds the steepest vector of the gradient of the loss. This vector is multiplied by a learning rate or step size alpha and then subtracted from the vector which represents a parameter. The learning rate models how strictly the updates should be made and its value affects the stability and the speed of the training process. The update function for a parameter in time t is calculated by the equation:

$$\theta^{(t+1)} = \theta^{(t)} - \alpha\Delta_\theta NLL^{(t)}, \tag{2.9}$$

where θ is the updated parameter, t is the training-step, $alpha$ is the learning rate and $\Delta_\theta NLL^{(t)}$ is determined for each θ by backpropagation.

With larger datasets, calculating the cumulative loss of the whole dataset in each training step takes a lot of time. To counter this Stochastic Gradient Descent is mainly used in deep learning in which a subset or a minibatch of the dataset is used for calculating the loss in a training step, instead of the whole dataset. The optimal value for the learning rate is essential for faster training and reaching better local optima, therefore modern models typically include learning rate optimizers such as schedule functions or Adam [120].

2.2.2.5. Inference

In Neural Machine Translation, inference involves the generation of a target language sequence given the source language sequence. This time however the source language sequence does not come with a target language sequence as its label. The model simply generates the most probable output based on the given input sequence and the already generated target language tokens until the end-of-sentence token is output.

The sequential nature of decoding brings some nuisances concerning generating the best output. Taking the most probable token in each decoding step, also known as the

greedy search, is one of the fast and straightforward methods yet yielding suboptimal results, since many optimal sequences do not have the token with the highest probability for the first places in the sequence. On the other end of the scale, calculating all possible sequences is not computationally possible as the number of choices grows with $|V|^{|d|}$ (V: vocabulary size, d: sequence length). Beam search makes a compromise between the two aforementioned methods by keeping the best n possibilities (beams) and continuing by considering every single possibility of the kept beams and discarding everything but the n best again.

2.2.2.6. Evaluation

Due to the flexible and creative nature of the languages, finding a quantitative metric for the quality of a translation has proved to be quite a challenge. Human evaluation methods are still the gold standard both in research and the industry. However, human evaluation not only suffers from subjectiveness (both between persons and within the same person at different times) but also the considered aspects in the translation are often not quantifiable or comparable. Examples of human evaluation metrics include a pair of 5-point [88] or 1-100 continuous [59] scales on the adequacy and fluency of the translation; or a ranking-based system in which the translations of the same text are ranked [142]. Assessing the quality of human evaluation and investing in its aspects is a hot research topic in the community.

Although automatic metrics lack delicateness, they are still desirable not only since they provide a quantifiable metric that is comparable, but also relative to human assessment, automatic evaluation is cheaper in terms of both money and time. There are several evaluation metrics used in machine translation, including BLEU [109], TER [130], METEOR [19], COMET [116] and others.

BLEU BiLingual Evaluation Understudy or BLEU [109] is the most widely used automatic evaluation metric in machine translation. It assesses both the adequacy and fluency of a translation. The BLEU score is calculated by the formula:

$$BLEU = BP \cdot exp\left(\sum_{n=1}^{N} \frac{log(p_n)}{N}\right), \quad \text{where } BP = min\left(exp\left(1 - \frac{r}{h}\right), 1\right) \tag{2.10}$$

and $p_1, \ldots, p_N$ stands for n-gram overlaps for $N = 4$ of the machine translation (hypothesis) and the real translation (reference). BP stands for the brevity penalty where the hypotheses of length h that are shorter than the reference of length r are penalized by this value.

BLEU is usually calculated on the document level, meaning the BLEU scores of every hypothesis-reference pair in the document are calculated separately and summed together. The choice for BLEU is due to its simplicity, correlation with human evaluation, and independence of language [113]. BLEU has also received many criticisms over the years [132, 28, 129]. Ananthakrishnan et al. [5] list several problems:

1. **Intrinsically meaningless score:** The BLEU score only measures the quality of a machine translation system in a relative manner. It can only be used to compare multiple machine translation systems' performance on the same input.

2. **Admits too much variation:** Measuring the score by n-gram overlaps allows many syntactically and semantically incorrect permutations of a correct sequence to get the same score as the correct sequence.

3. **Admits too little variation:** Measuring exact matches in n-grams penalizes the usage of synonyms or similar phrases. Additionally, machine translation systems are penalized too severely while translating to/from heavily inflected languages.

4. **Problems with writing systems with no word boundaries:** To measure n-grams BLEU requires a separation marker between words, which is typically a space in many writing systems. However, several Asian writing systems (e.g. Chinese, Japanese, Thai scripts) do not separate words.

It is possible to measure the BLEU score on tokenized or detokenized sentences (see section Subword-Tokenization). Post explains how measuring tokenized hypotheses and references brings unclarity and inability to recreate results [113] since each work uses its separate pre/post-processing schemes on both hypotheses and references. He additionally proposes SacreBLEU [113], a tool for measuring BLEU without processing the reference, which became standard in the machine translation community.

chrF chrF [112] is an alternative metric used in machine translation that measures character n-gram-based F-score. It is calculated with the formula:

$$chrF\beta = (1 + \beta^2)\frac{chrP \cdot chrR}{\beta^2 \cdot chrP + chrR}, \tag{2.11}$$

where $chrP$ stands for character n-gram precision, i.e. percentage of character n-grams in the hypothesis which also appears in the reference, averaged over all n-grams; $chrR$ stands for character n-gram recall, i.e. the percentage of character n-grams which are also present in the hypothesis. β is a parameter to adjust the weight of precision and recall, where a value of $\beta > 1$ gives more importance to recall, of $0 \leq \beta \leq 1$ to precision. If $\beta = 1$, then both precision and recall are equally weighted. chrF3 scores, i.e. with $\beta = 3$ is commonly used alongside BLEU is presented especially for low-resource language machine translation works because many low-resource languages lack a standard tokenization [149] and chrF3 has a high correlation with human evaluation when translating from English [131].

2.2.3. Low Resource Languages

Neural models excel at learning a certain task only when suitable training data is provided with a great multitude. When training data is limited, models tend to overfit the features within the data, thereby losing generalizability, meaning its performance wanes when any kind of valid input that shares only a few features with examples in the training data is given.

Based on the size and cultural activity of their communities, various amounts of content are created for each language. In the digital age, a language's vitality, and its potential to develop further can be partially associated with its presence on the internet. The online presence not only helps the language reach a broader audience, but also any online content of a language can be used as training data for natural language processing tasks after preprocessing it accordingly. Digital parallel corpora, i.e., the translations of documents shaped similarly to Rosetta Stone, are the main training material for neural machine translation models. These are considered as labeled data since each text has a translation provided as its label (for both languages in the pair). There are a multitude of open-sourced parallel corpora for many language pairs available on OPUS[3] [135].

Joshi et al. [74] classify the world languages by their availability of resources suitable for natural language processing tasks, assigning a number based on available data online and activity in research:

- **Class 0 (The Left-Behinds):** With no available labeled and very limited unlabeled data online, these languages are mostly neglected in the community.
- **Class 1 (The Scraping-Bys):** These languages have some unlabeled data online, having the potential to get support from natural language processing researchers after processing the unlabeled data.
- **Class 2 (The Hopefuls):** Languages with a very limited amount of labeled data, which enjoy the small amount of signs of interest by the natural language processing community.
- **Class 3 (The Rising Stars):** Languages with online presence but lack the according levels of labeled data.
- **Class 4 (The Underdogs):** Languages with support from the natural language processing community but have relatively lesser labeled data.
- **Class 5 (The Winners):** Languages with significant online presence, having multiple industrial and governmental investments for natural language processing research.

Another way to investigate a language's vitality and available data for natural language processing research is to assess the amount of Wikipedia articles on that language since these can be used as training data. In the case of machine translation, the translations of

[3] https://opus.nlpl.eu/

the articles can be used as parallel corpora, although the pre-processing concerning the alignment of sentences is required.

Until 2017, Western Armenian did not have its separate ISO 639-3 language code, which is the main requirement to create a Wikipedia domain dedicated to a language. Until then Eastern Armenian Wikipedia included both Western and Eastern variants of an article in separate pages. As of August 2023, Western Armenian Wikipedia[4] includes 11210 articles [152]. The maintenance is currently undertaken by several foundations and volunteers.

According to the classification by Joshi et al. [74], Western Armenian can be assigned to class 1 (the Scraping-Bys) since there is a relative online presence of Western Armenian in the form of media outlets, a dedicated Wikipedia, websites of several organizations and foundations of diasporan Armenian communities throughout the world and individually created content in various social media platforms. However, this content is not suitable for natural language processing research for Western Armenian as tools for processing the data are currently lacking.

The machine translation of low-resource languages is a heavily researched topic and throughout the years many methods were proposed to include machine translation of low-resource languages. Ranathunga et al. [115] provides an overview of the methods and techniques regarding low-resource language translation. The main methods and techniques for low-resource machine translation include:

1. **Data Augmentation:** The main goal in data augmentation is to generate synthetic data from existing sources. In the case of NMT, this is realized by generating synthetic parallel texts from monolingual corpora. There are three main approaches to data augmentation:
   a) **Replacement Based Data Augmentation:** Synthetic data is generated from monolingual or parallel corpora by replacing words with their uncommon synonyms [51] or by replacing phrases with other similarly meaning phrases [95].
   b) **Backtranslation Based Data Augmentation:** An already existing machine translation model is used to generate synthetic translations of monolingual corpora, i.e., synthetic parallel corpora [124].

2. **Unsupervised NMT:** As low-resource languages lack the necessary parallel corpora for supervised training, unsupervised NMT makes use of monolingual corpora of the languages in translation. Before learning to translate, the models are initialized by learning the language space of each language from its monolingual corpus. More modern approaches use the embeddings of pre-trained language models [35]. Then the translation model learns a transformation between the language spaces. This results in cross-lingual embeddings in which both languages are represented. Using cross-lingual embeddings as input, Artetxe et al. [11] propose an architecture with 3 main parts: a shared encoder that accepts cross-lingual embeddings, and two decoders for each language in the translation pair. The training involves the processes of denoising, where a noised version in language 1 (L1) is generated by

[4] https://hyw.wikipedia.org/

the shared encoder and given to the L1 decoder whose task is to reconstruct the noiseless version of the L1 sentence; and on-the-fly backtranslation, where the noised L1 sentence is translated by the shared encoder into L2 and decoded by the L2 decoder. The model optimizes to probability obtain the noiseless L1 sentence. The denoising and on-the-fly backtranslation of L2 input is also done analogously.

3. **Multilingual NMT:** Multilingual NMT models are models that can translate between many language combinations. Multilingual NMT models learn a shared representation of languages from both multilingual and/or bilingual parallel corpora, which can be compared to interlingua [72]. The shared representation allows the translation of language pairs that do not have a parallel corpus for training. When the number of included languages is small and the included languages are related, multilingual systems outperform their bilingual counterparts [91]. Multilingual NMT models require at least one parallel corpus for a language to be included in the model. Additionally, the training data should include similar amounts of examples for each language to have balanced translation performance across all included languages [7].

4. **Transfer Learning:** Transfer Learning involves the utilization and transfer of knowledge gained by a neural model in another model. In NMT, Transfer Learning is employed by training or using a pretrained parent model, typically a multilingual machine translation model with high-resource language pairs. This model is then fine-tuned with the smaller training material of the low-resource language, yielding the child model. The fine-tuning can be done with different levels of freezing of the parent model parameters, i.e. not updating the values of frozen parameters during training. To maximize the knowledge transfer, the language(s) within the parent model and the child model must be closely related [40, 108], meanwhile some other factors such as the corpora sizes, domains, overlap in vocabulary, and language scripts should be considered to improve the transfer between the parent and the child model [4, 87, 39, 93].

A selection of methods mentioned here was used to mitigate the low-resource problem of Western Armenian as explained in chapter 5.

2.2.4. Rare Word / Out-of-Vocabulary Problem

The vocabulary of a neural machine translation model is a set of units (e.g. words, tokens, characters) that build the input and output sequence. The vocabulary can be constructed by the model during training, or a predetermined vocabulary can also be provided to the model. While training the model learns to represent the units in the vocabulary themselves and their semantic and syntactic relations with each other. The quality of a unit's representation improves when more examples of that unit are provided in the training data. Neural machine translation models require a large vocabulary in order to capture a broad set of domains within the languages of translation. A larger vocabulary however not only requires a larger number of examples which scales very fast, but also the requirements of the model get bigger in terms of training time and memory.

According to Zipf's law, some words appear more frequently than others in any kind of corpus. Thus, some words in the training data appear less often and therefore it is represented more poorly in the vocabulary. This is called the rare word problem. Since there are fewer examples of a rare word, the model struggles to generalize and becomes able to generate only the sequences that are similar to the training example. To boost the representation of rare words methods like replacement-based data augmentation are utilized [51].

Language has an unlimited field in terms of words and expressions. New words and ways of expression emerge every day in every human language. Representing each word in a limited-size vocabulary is an impossible task and therefore some words will stay out of the model's vocabulary. When the model comes across a word that is out of its vocabulary, it cannot calculate the probabilities when no proper handling method is employed. This is called the out-of-vocabulary problem and the more severe case of the rare word problem.

The simplest way to handle out-of-vocabulary problem is to represent any out-of-vocabulary word with a special "unknown" token or `<unk>`. With the inclusion of an unknown token, the calculations can continue. Unknown tokens can also appear during training, especially when predetermined vocabularies are used. In this case, the unknown token's hidden representation will include the information of all out-of-vocabulary words. Luong et al. [97] suggest outputting the unknown tokens and their correspondence in the source language and handling them in post-processing either by copying the source word (in the case of named entities, e.g., personal, locational, organizational names) or replacing the unknown token by the dictionary translation of source side word.

The vocabulary of earlier machine translation models was mainly comprised of words, which was the most intuitive way to think about a vocabulary, but due to the limited size and endless number of possible words in language this results in the out-of-vocabulary problem. Most languages have on the other hand a certain number of characters and using characters as the units of sequences allows the representation of any sequence, but as the units get smaller the sequence gets longer and thus the models require even larger memories and longer training and inference time. A character-based model will have to spell the word "the", which requires 3 time units, while a word-based model will output "the" in a single time step.

Character n-gram-based models make a compromise between character and word-based models. Their unit of sequence is characters, using them allows more plausible training and inference times while still avoiding the out-of-vocabulary problem. However, an arbitrary choice of n results in tokens that do not correctly split a word into morphemes as not all morphemes share the same amount of letters. Thus, Sennrich et al. [125] propose using Byte-Pair Encoding, a compressing technique, to tokenize the data and build the vocabulary from the tokens. Figure 2.6 illustrates a step-by-step tokenization example according to the Byte-Pair Encoding algorithm. In essence, Byte-Pair Encoding splits the given input by the most frequently appearing tokens. This way, words like "the" get represented by a single token while rare words are split into sub-word tokens which although not always but usually are morphemes. Using morphemes allows better translation especially in highly compounded or highly inflected languages, since usually morphemes have correspondences across languages, but it also reintroduces the out-of-vocabulary problem, as there can be an infinite number of morphemes. Byte-pair encoding

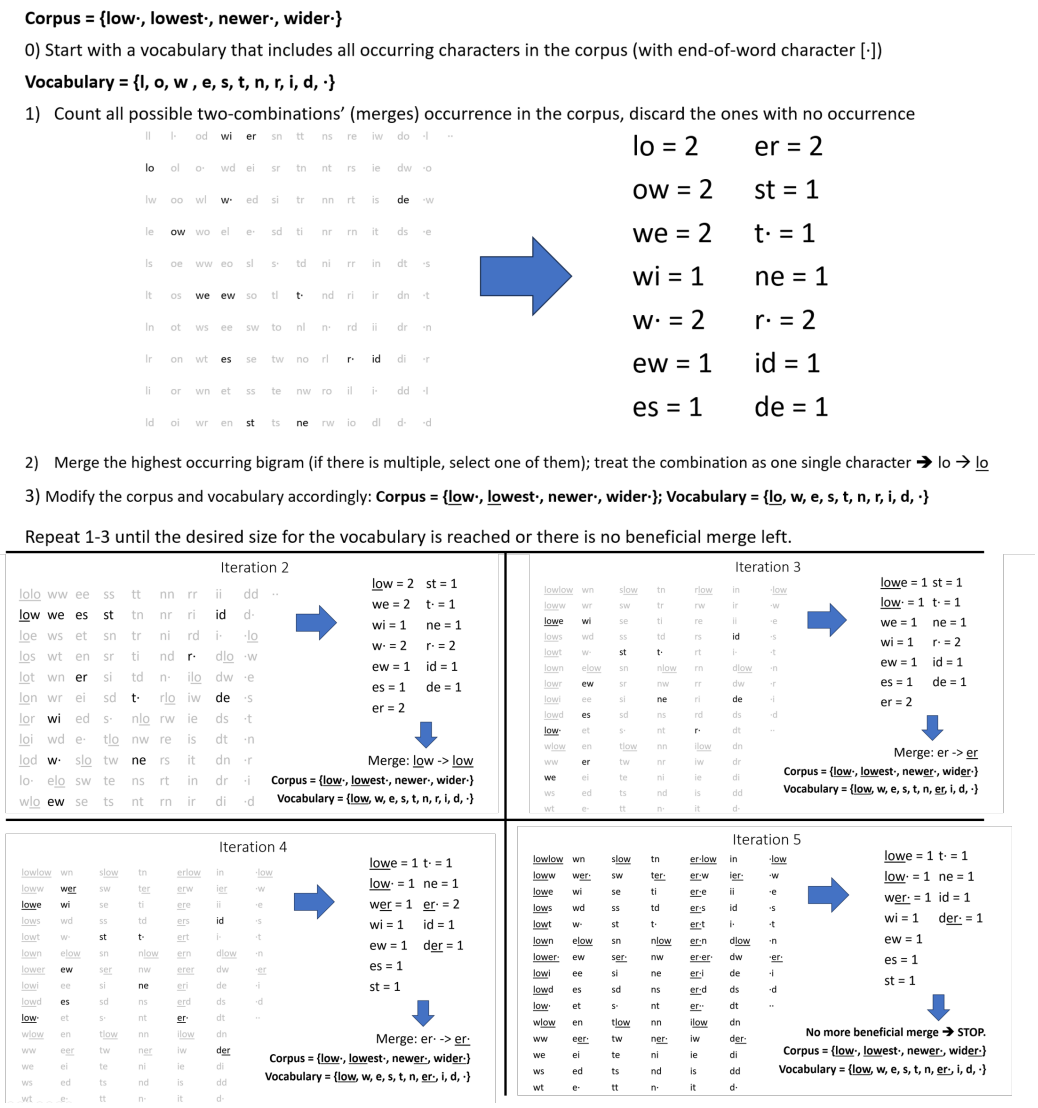

Figure 2.6.: Tokenization of an example corpus according to the Byte-Pair Encoding algorithm

is a great approximation since it allows the vocabularies of models to hold a subset of morphemes while keeping the flexibility high by including other non-morpheme tokens (simple character sequences) to represent any kind of input, and thus it is currently the standard technique in research to handle the out-of-vocabulary problem.

3. Related Work

This chapter serves as a comprehensive overview of works, resources, and tools regarding both Armenian standards as well as the base neural machine translation model "No Language Left Behind" [133].

By showcasing available data, tools, models, and other work regarding Armenian we aim to provide a head start in further research and assess the current status of both modern Armenian variants in terms of natural language processing research.

No Language Left Behind [133] is an open-source multilingual neural machine translation model that supports more than 200 languages and generates translations of state-of-the-art quality. We dive into its creation process to provide information and context as all the models within this work are finetuned upon this model.

3.1. Natural Language Processing Research for Armenian

In many papers, Armenian is classified as a low-resource language [139, 58, 77, 68, 148, 144, 62], but the distinction between the Western and Eastern standards is usually neglected and the papers almost always consider Eastern Armenian. The cause might originate from the fact that Western Armenian is not represented as an official language of a country. Both standards and Classical Armenian have a wide heritage of literature that is yet to be digitized. The “Digitizing Armenian Linguistic Heritage: Armenian Multivariational Corpus and Data Processing” (DALiH) project was initiated in April 2021 by INALCO with the aim of creating annotated corpora for Armenian natural language processing research [67]. Particularly the project aims to create separate corpora for Classical, Middle, Modern Standard Western, and Modern Standard Eastern Armenian as well as a separate corpus for the smaller dialects. The project will be launched in 2025 [82].

This section focuses on listing the data, works, and tools regarding the natural language processing of both variants.

For a comprehensive overview please refer to Dolatian’s list of Armenian resources [65].

3.1.1. Eastern Armenian

3.1.1.1. Data

The Eastern Armenian National Corpus[1] [81] is currently the largest open-source corpus of Eastern Armenian with morphological, semantic and metatext annotations of 109 million tokens across almost 10000 documents of written and oral discourse [82]. The

[1] http://www.eanc.net/

morphological analysis and annotation are performed by a rule-based system. The English translations of the frequent tokens are also provided. The morphological analyzer was updated to comply with the UniParser format [8] and is available online[2]. The corpus is planned to be extended both in terms of tokens and of neural network models for morphological analysis including a recurrent neural network based and a transformer-based model [82].

There is a WordNet for Eastern Armenian created by Bond and Foster [25]. A WordNet is a database of "synsets", the groups of words which have similar meanings. Other relationships of words like hypernyms, hyponyms, and meronyms are also included in WordNets to properly represent the associative structure of languages.

The Universal Dependencies project includes 2 treebanks for Eastern Armenian[3] [4]. A treebank is a database of sentences that are annotated with syntactical information. The syntactic structure of the sentences is represented in the form of trees, hence the name treebanks. The two Eastern Armenian treebanks have a combined number of 4800 annotated sentences including a total of 94162 tokens.

ArmSpeech-POS [17] is a Part-of-Speech tagged corpus for Eastern Armenian that includes 57160 tokens over 6180 sentences. Both treebanks and Part-of-Speech tagged corpora mainly represent syntactical information in the sentences. The main difference is the complexity of information as treebanks show the Part-of-Speech tags assigned to each individual token as well as the hierarchical dependencies between tokens, thanks to its structure, whereas in Part-of-Speech tagged corpora only the assigned tags are included for each token. Part-of-Speech tagged corpora require however less preprocessing and thus they are preferable as training data.

There are several available sets of pretrained monolingual word embeddings for Eastern Armenian, including fastText [24, 60], YerevaNN[5], pioNER [55] and embeddings of Avetisyan and Ghukasyan [14]. Eastern Armenian is often included in multilingual natural language processing models and in their multilingual word embeddings [52, 3, 38, 147, 10, 16, 22].

Parallel corpora of Eastern Armenian with various other languages are available in OPUS[6].

3.1.1.2. Models / Tools

Lemmatization and Part-of-Speech Tagging are important natural language processing tasks, and their models are typically used in preprocessing or as an individual module of more complex natural language processing tasks regarding the extraction of information. 2 neural models with different architectures were proposed for the lemmatization/Part-of-Speech tagging for Eastern Armenian: 1) the model by Ghukasyan and Avetisyan [54] based on the COMBO-Architecture [121] (a stack of biLSTM layer to obtain a reduced size representation of character embeddings 3 dilated convolutional layers and a softmax

[2] https://github.com/timarkh/uniparser-grammar-eastern-armenian
[3] https://github.com/UniversalDependencies/UD_Armenian-ArmTDP/tree/master
[4] https://github.com/UniversalDependencies/UD_Armenian-BSUT/tree/master
[5] https://github.com/YerevaNN/word2vec-armenian-wiki
[6] https://opus.nlpl.eu/

layer that outputs the probabilities). 2) ArmParser [6], an extension of sequence tagging network [118] with multiple decoders realized by Gated Recurrent Units.

Armenian is written with its own script and therefore popular OCR tools like Tesseract OCR produce many errors. Tigranyan and Ghukasyan [136] propose a two-step post-processing method to reduce the erroneous OCR output: 1) a multilayer perceptron to detect the errors in OCR, 2) a COMBO network [121] to correct the detected errors.

Sentiment analysis is a popular natural language processing task that involves assessing the general tone of emotion (positive, negative, neutral) of a given text. Emotion recognition is a similar task in which the given text is classified into one of the emotions (e.g. happiness, sadness, anger, fear, surprise, etc.) Kalayjian [76] offers a BERT-based model to handle both tasks for Eastern Armenian. The annotated data for both sentiment analysis and emotion recognition is also publicly available[7].

The National Center of Communication and Artificial Intelligence Technologies [151] has built various natural language processing models for the tasks of automatic speech recognition[8], speech synthesizer[9] and machine translation[10] in Eastern Armenian with the contributions of voluntaries who read predetermined sentences to populate the speech corpus.

As of August 2023, there are 15 machine translation APIs that support Eastern Armenian, including Google [56], Yandex [43] and Baidu Translate [36]. 2 companies (Happy Scribe [123] and VocalMatic [137]) provide audio transcription services using proprietary Speech-to-Text software for Eastern Armenian.

3.1.2. Western Armenian

3.1.2.1. Data

As for annotated corpora, there are a few small corpora [46, 80, 79] with hierarchical syntactic annotations using the linguistic annotation software NooJ [128].

There is one Western Armenian treebank, Western Armenian ArmTDP [100], included in the Universal Dependencies project. It contains a total of 121583 tokens over 6656 sentences.

Collecting, preprocessing, and maintaining monolingual and parallel corpora is a cornerstone for natural language processing research and therefore it is of great importance to list as many possible sources for Western Armenian monolingual/parallel corpora as possible, although scraping and preprocessing scripts for each source will be required individually. Table 3.1 illustrates a list of various online Western Armenian unprocessed resources.

3.1.2.2. Models / Tools

Dolatian et al. [44] provide an open-source morphological transducer for the highly inflected Western Armenian language, which has a 90.6% precision and 74.82% recall over

[7] https://github.com/nigkal/ArmenianNLP
[8] https://aws.ican24.net/asr/index.php
[9] https://aws.ican24.net/tts/index.php
[10] https://aws.ican24.net/nmt/index.php

Category	Name (Add. Description)	Website
News Websites	Aztag	https://www.aztagdaily.com/
	Asbarez	https://asbarez.am/
	Hairenik Weekly	https://hairenikweekly.com/
	Jamanak	http://www.jamanak.com/
	Agos	https://www.agos.com.tr/am/mudki-ech
	Nor Marmara	https://www.normarmara.com/
	Hye Tert	https://hyetert.org/
	Massis Weekly	http://massisweekly.com/archives.html
	Nor Or	https://noror.org/
	Oragark	https://www.oragark.com/hy/
	Hayern Aysor	https://hayernaysor.am/wa/
	Massis Post	https://massispost.com/am/
	Archives of Avedik	https://web.archive.org/web/20111204204934/http://www.armeniancatholic.org/inside.php?lang=en&page_id=71
	Hask	https://www.armenianorthodoxchurch.org/hask-am
	Diario Armenia	https://www.diarioarmenia.org.ar/armenian/
	Sardarabad	https://www.sardarabad.com.ar/idioma/armenio
	Armenia Media (includes audio/video recordings)	http://www.armenia.com.au/
	Horizon Weekly	https://horizonweekly.ca/en/
	Torontohye	https://torontohye.ca/home-arm/
	Nor Haratch	https://norharatch.com/main-hy
	Azat Or	https://azator.gr/
	Ararad Daily	https://araraddaily.com/
Organizational Websites	Hamazkayin	https://hamazkayin.com/
	Homenetmen	https://www.homenetmen.org/hy/
	Armenian Community Centre of Toronto	https://www.acctoronto.ca/
	Calouste Gulbenkian Foundation - Armenian Communities	https://gulbenkian.pt/armenian-communities/hy/
	Hrant Dink Foundation	https://hrantdink.org/hyw/
Archives *(Of scanned documents which probably need OCR)*	ARAM Repository of Historical Newspapers	https://webaram.com/hy/biblio/presse
	Databases of National Library of Armenia *(Classical, Western, Eastern)*	https://nla.am/en/resourses-en?page=1
	Digital Library of American University of Armenia *(Classical, Western, Eastern)*	https://digilib.aua.am/
	Grahavak *(Classical, Western, Eastern)*	https://grahavak.blogspot.com/p/blog-page.html
	Mekhitarist Congregation Library *(Classical, Western)*	https://mechitaristlibrary.org/
Audio	Rerooted *(Audio recordings with Western Armenian and English transcriptions)*	https://www.rerooted.org/archive
Miscellaneous	Houshamadyan	https://www.houshamadyan.org/arm/home.html
	Zarmanazan	https://zarmanazan.com/
	Zndoog	https://zndoog.com/

Table 3.1.: List of Online Western Armenian Resources to gather textual/audio data from

1225 tokens in testset. They also release a web-scraped Western Armenian news corpus of Kantsasar News[11].

Vidal-Gorène et al. [141] build a Recurrent Neural Network Encoder-Decoder model for lemmatization and Part-of-Speech tagging tasks. The model uses a joint loss function that is comprised of a standard word-level classification loss and a bidirectional word-level language modeling loss. This is proposed by Manjavacas et al. [99] and they claim including language modeling loss allows the model to learn representations in such a way that it can easily disambiguate lemmas. Using this loss achieves state-of-the-art lemmatization scores for non-standard and historical languages. Returning to the original paper, Vidal-Gorène et al. [141] observe using a model trained on Eastern Armenian data obtains high rates of correct classifications on both lemmatization (88.79%) and Part-of-Speech tagging (87.33%) tasks and mention that the model "could be a baseline for a massive corpus annotation in [Western Armenian] standard." [141]

The project of the National Center of Communication and Artificial Intelligence Technologies [151] aims to collectively build a speech corpus for Western Armenian for Automatic Speech recognition and similar tasks are currently underway. It is requested by the participants to read a selected passage via the project's platform[12].

[11] https://github.com/mr-martian/hyw-corpus/tree/master/Newspaper
[12] https://aws.ican24.net/hywrec/index.php

3.1.3. Both Languages

3.1.3.1. Data

Currently, there are no parallel or comparative corpora of Western-Eastern Armenian, however, it is possible to create corpora from books or news websites.

3.1.3.2. Models / Tools

Avetisyan [13] compares statistical and neural models for the language identification task between the languages Classical, Western, and Eastern Armenian. The neural model based on the fastText [75] achieves a 98% accuracy at the sentence and a 100% accuracy on the document level, where statistical models peak at 67%. The data was scraped from Eastern and Western Armenian Wikipedia articles. Another interesting point from this work is that limiting the sentences to 200 characters each is enough to reach 100% sentence accuracy.

Chakmakjian and Wang [31] investigate the requirements, available data, and challenges of building a unified Automatic Speech Recognition system for both Eastern and Western standards and offer a simple methodology to tackle this problem.

Transliteration software with several standards for Classical, Western, and Eastern Armenian is available online[13]. To convert between classical and modern orthographies an online[14] and an offline[15] tool are provided.

There is a translation software provided by ISMA[16], which offers translation between 15 languages, including Classical, Western, and Eastern Armenian as well as Hamshen dialect, high resource languages like English, French, German and Russian, low-resource languages like Latin, Talysh, Kurdish and Yezidi. The software seems to be incomplete since some translation directions (e.g. English to Yezidi) do not work and most importantly the translation lacks the quality to/from Western Armenian as it falls short in terms of fluency, the sentences are translated word by word without considering context. E.g.

ENG: *I don't know why I'm a little worried now.*

HYW: *Ես գիտնալ ինչո՞ւ ես փոքր հիմա անհանգստացայ:*

Rom.: *Yes kidnal inchu yes p'okr hima anhanksdats'a*

Pron.: (j)ɛs kʰidnɑl intʃu (j)ɛs pʰokʰər himɑ ɑnhɑnkʰəsdatsʰɑ

The verb գիտնալ (to know) is not conjugated and the translation of the word "little", "փոքր" does not make sense, since little in this sentence does not refer to the size of a shape but the amount/degree of worry, which should be translated into "քիչ մը". Both phrases "a little" and "քիչ մը" are quite common in their respective languages, and a translator based on a statistical/neural model should capture this relationship, but ISMA does not provide any insight on their software, leaving only room for speculation.

[13] https://www.translitteration.com/transliteration/en/armenian-eastern-classical/iso-9985/
[14] https://arak29.org/on-line-armenian-orthography-converter/
[15] https://github.com/instigatetcf/armenian-orthography-converter
[16] http://translator.am/en/index.html

3.2. No Language Left Behind

Finding online parallel corpora for low-resource languages is hard and when such a corpus is found, they often lack the required size for both training and evaluation. Bitext mining, i.e. comparing the representations of a candidate translation pair by similarity and accepting it if the pair exceeds a certain threshold, is a cheap and effective method to build a parallel corpus from comparable corpora, but it produces noisy translation pairs which are not desirable for evaluation of a model's translation quality. Thus, Meta (Facebook) researchers released the FLORES-101 evaluation benchmark [58], a small multilingual corpus sampled from English Wikipedia articles about topics from various domains and translated into 101 languages by professional translators. This corpus is extended to cover 204 languages as a part of the No Language Left Behind (NLLB) [133] project, in which along with the FLORES-200 evaluation dataset, several additional human translation datasets for training (NLLB-Seed, NLLB-MD, Toxicity-200); tools for bitext mining (LASER3 encoder models for 148 languages to obtain representations for comparison, and stopes a bitext mining library to prepare monolingual data, process via LASER3 models, compare and obtain aligned bitexts); as well as multiple translation models covering 202 languages with different sizes: A Mixture-of-Expert model with 54.5B parameters, 3.3B and 1.3B Dense models and distilled models of 54.5B MoE model with 1.3B and 600M parameters.

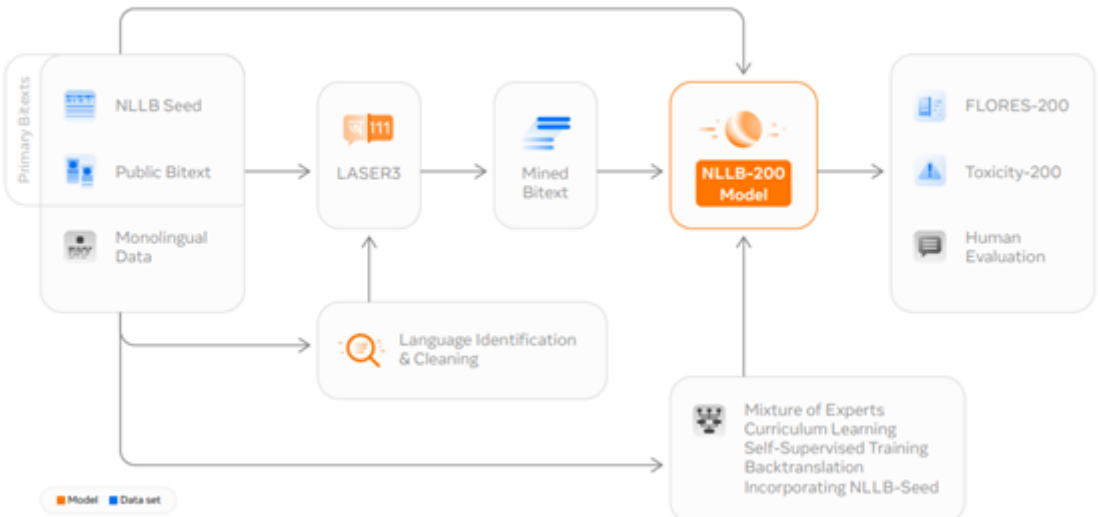

Figure 3.1.: NLLB-200 Model Creation Pipeline [133]

In Figure 3.1 the NLLB-200 model creation pipeline is shown. The training data for the model originates from the available online parallel corpora for high-resource languages, which require only preprocessing steps such as cleaning, deduplication, and tokenization; and low-resource parallel corpora are built from mined bitexts using LASER3 embeddings.

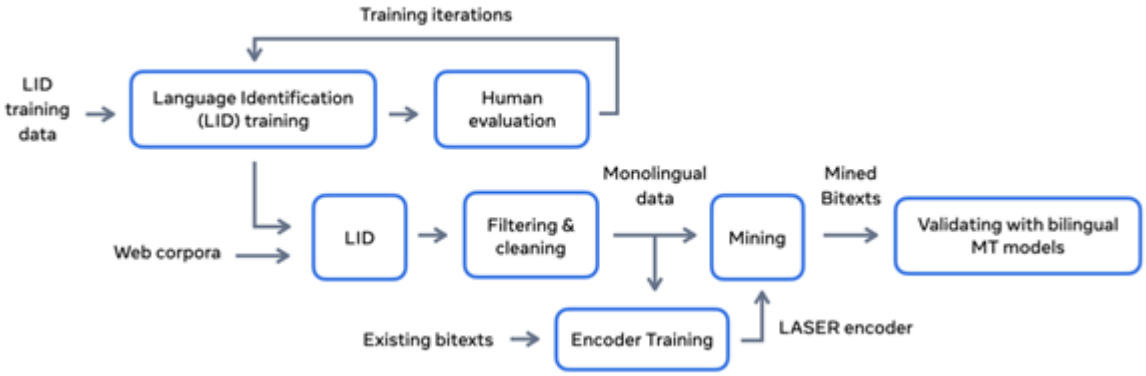

Figure 3.2.: Bitext Mining Pipeline of NLLB to obtain multilingual parallel data [133]

3.2.1. Bitext Mining

Figure 3.2 shows the overview of the bitext mining process. The training data for each LASER encoder is collected in 2 main ways. The first method involves training a language identification model to classify each given sentence to a language, as online documents tend to include parts in multiple languages. For this, monolingual data labeled with the corresponding language code is given for training. The language identification model is based on fastText [75] which accepts character n-gram embeddings of input text and returns a probability distribution of the languages that the input can belong to. This model is then used to identify monolingual web corpora, whose process follows a hierarchical manner. A monolingual document is first given in paragraphs then for each paragraph's identified language a sentence splitting technique is employed, and sentences are once again identified. The non-matching sentences with the paragraphs are filtered. Several other filters are utilized including a threshold for the identification score, the number of digits, punctuation marks, and emojis. Filtered sentences are then evaluated by a language model for quality when a language model for that language exists, and duplicate sentences are discarded. The cleaned and filtered monolingual data includes an average of 26.8 million sentences across low-resource and an average of 1.3 billion sentences across high-resource languages. Data from existing parallel corpora is added to the cleaned monolingual corpora and the combined corpus is then used to train LASER3 encoders.

The mining model is realized by a teacher-student approach, it involves specifically training a massively multilingual sentence encoder first as the teacher. The teacher is an extension of the LASER encoder [62], called LASER2, which employs a slightly different training process than the original proposition [10]: a custom SentencePiece tokenizer to tokenize texts in each included language's script and changed values for upsampling the representations of the low-resource languages. Each student model is assigned to encode a single or a group of similar low-resource languages. After training the teacher model, each student model trains to minimize the cosine similarity loss of its output with the output of the teacher model. Some student models also minimize their respective monolingual language modeling loss to utilize the monolingual data.

To evaluate the similarity of sentences a margin-based score proposed originally by Artetxe and Schwenk [9], called in this work as xsim, is used, which considers both the cosine similarity of the candidate pair and the average cosine similarity of each combination in the nearest neighborhood, better formulated as:

$$score(x, y) = margin\left(cos(x, y), \sum_{x \in NN_k(x)} \frac{cos(x, z)}{2k} + \sum_{v \in NN_k(y)} \frac{cos(y, v)}{2k}\right), \tag{3.1}$$

where x and y are source and target sentences, $NN_k(x)$ is the k nearest neighboring sentences in the opposing language, and $margin(a, b)$ is defined in this case as the ratio of a and b ($\frac{a}{b}$). The scores are only calculated for x $\rightarrow$ English direction. English sentences are calculated by the teacher and the other language by the student model.

Several datasets of mined bitexts with different thresholds for the lowest xsim score for inclusion were created. An individual translation model was trained and the quality of each was tested on the FLORES-101 benchmark. The experiments showed the threshold for the lowest score should be 1.06 for a mined bitext to be included. Using this threshold, 761 million sentence pairs including English as well as 302 million sentence pairs not including English were mined.

3.2.2. Model Architecture

The NLLB-200 translation model is based on the traditional transformer encoder-decoder architecture [140]. The only difference in dense translation models is the choice of applying layer normalization inside the sublayers at the beginning (Pre-LN) instead of applying it between layers after the residual addition (Post-LN), which was shown to improve stability during training [146].

Traditional massively multilingual translation models based on dense architectures activate all parameters in each step of the training whose batch includes sentence pairs of different translation directions, effectively causing interference between unrelated languages [52]. As a solution, Mixture of expert sublayers [48, 150, 126] were proposed. They include E Feed-Forward-Networks, i.e. experts, and a gating mechanism that activates an expert according to the input. The experts and the gating mechanism together realize learning how much to share information for cross-lingual transfer and how much to mutually exclude information to mitigate interference. We refer to the NLLB paper [133] again for further implementation details, including the additional gating for activating between the shared expert and the specialized expert. In the NLLB Mixture-of-Experts model every f_{MoE} = 4th feed-forward sublayer with a gated mixture-of-experts sublayer.

3.2.3. Used Datasets

The types of datasets used in NLLB models can be divided into 3 categories:

1. **Primary Bitext:** Data from available parallel corpora.

2. **Mined Bitext:** Data mined from monolingual corpora pairs.

3. **Monolingual Data:** Data utilized in synthetic generational/mining methods like bitext mining, backtranslation, and self-supervised learning.

Several filtering strategies were employed to provide high-quality training data to the models:

1. **LASER Filtering:** Any mined sentence pair with a similarity score lower than 1.06 was excluded.

2. **Length Filtering:** The length is a heuristic to assess the alignment of candidate sentence pairs, however employing a single value for the ratio between sentences would not work. To make the lengths of the sentences comparable, the character lengths of each language's FLORES-200 dataset are calculated and for each, the ratio to the English dataset is computed. The value obtained can then be used for computing the sentence lengths' of any language into the "English" unit. After conversion, the ratio between the lengths of sentence pairs can be calculated. Sentence pairs exceeding the threshold value of 9 are excluded. Additionally, for backtranslated data, sentences with a converted length value below 15 are excluded.

3. **Toxicity Filtering:** Sentence pairs with high toxicity imbalance are excluded. For the detection/implementation details concerning toxicity please refer to the original paper [133] as it does not bear adequate relevancy for this work.

4. **Deduplication:** To determine duplicate sentences, the sentences are first normalized by removing punctuations, non-printing characters, and replacing numbers. There are three types of deduplication including the deduplication of pairs, i.e. the occurrence of multiple identical pairs in both source and target side; the source deduplication, i.e. different source side sentences are aligned with the same target sentence; and the target deduplication, i.e. different target sentences are aligned with the same source sentence. Aside from pair deduplication for all datasets, source deduplication is employed for backtranslated datasets and target deduplication for mined datasets.

3.2.3.1. Methods for Leveraging Monolingual Data

Self-Supervised Learning Self-supervised learning generally aims to learn general patterns in the sequences of a language using only monolingual corpora. When included in the training process along with the translation task, it can help improve the quality of translations [20]. Experiments over including self-supervised learning showed that a combined task of denoising auto-encoding [96] and multilingual translation yields better results in translation than only learning the task of multilingual translation. The combined task can be described in a nutshell as learning to predict the target sequence given a noisy version of the input sequence.

Backtranslation The researchers include two large backtranslated datasets: one generated with a multilingual neural model and one with a series of bilingual MOSES [89] statistical models. Combining both backtranslated datasets with the primary and mined bitexts

results in higher scores than giving each dataset individually. To avoid overfitting the synthetically generated data, the inputs are provided with an extra tag indicating the dataset that they originate from. This is called tagged backtranslation [30] and previous work has shown improvements [101] in the performance of translating low-resource languages.

3.2.4. Vocabulary

For tokenization, a SentencePiece model is trained with a combined bitext dataset including all languages. For a balanced representation of each language in the vocabulary, the temperature sampling value of 5 is included as a hyperparameter, effectively upsampling the low-resource language tokens while downsampling high-resource ones. The vocabulary size is 256000 tokens.

3.2.5. Training

The naïve approach to include the data for all languages jointly causes overfitting for low-resource languages. The model requires many training steps to see all the examples of a high-resource language since they usually have a large number of examples; this means the few examples of low-resource languages are consequently seen too many times, causing the loss of generalizability for these languages. Kuwanto et al. [90] proposed a curriculum design for training multilingual models with training data of various sizes. In a nutshell curriculum learning in machine translation involves introducing high-resource training examples first during training and after a while low-resource examples. The NLLB team experimented with 2 strategies to find optimal points during training to introduce low-resource examples gradually. The first approach involves splitting the language pairs into two groups: the ones with over than 9 million sentence pairs and the ones with less than 9 million sentence pairs. The training only includes the language pairs with over 9 million sentences and after 200000 steps the rest were added as well. The second approach involves training a model without a curriculum and observing each language pair's step number where the model starts to overfit for that pair. Then the pairs are divided into 4 buckets in which languages have similar training steps where the model starts overfitting for them. Both approaches have similar performances on average, but the latter strategy is more beneficial for mitigating overfitting of low-resource pairs therefore it is the chosen curriculum learning strategy for the final model. Nonetheless, the former approach is cheaper since it does not require training another model to obtain the optimal buckets.

3.2.5.1. Knowledge Distillation

Knowledge Distillation is a technique for smaller models to learn the representations of a larger model to perform the same task. This makes not only the models more accessible and portable but also with fewer parameters the inference times get shorter with a small cost of performance. Training a small model with knowledge distillation yields better performance than training a model with the same size on the same task from scratch [57].

Two knowledge distillation techniques covering online and offline settings were investigated. The online knowledge distillation was employed on the word level where the smaller model is trained on the same training data with an additional objective to minimize the cross-entropy loss of the larger models' vocabulary probability distribution. The offline knowledge distillation is done on the sentence level. The larger model generates the translation of an input, and the smaller model learns to imitate the larger model's output. Compared by the performance, the latter method yields better results, however, it is temporally expensive since the training data must be generated online by a larger model and it requires a longer time for generation. The former instead does not require any additional time for generating training data.

3.2.6. Performance

NLLB-200 was compared with several state-of-the-art multilingual machine translation models from the research and industry. NLLB-200 outperforms both in English-centric and non-English-centric pairs the state-of-the-art research models of M2M-100 [52], DeepNet [143] and DeltaLM [98] models when compared over the FLORES-101 test dataset. For the translations to English NLLB-200 also outperforms the model provided by Google's Translation API[17].

All the models and scripts for data preparation and evaluation datasets are provided on the FAIRSEQ's dedicated branch for NLLB[18]. Additionally, several demos of NLLB models are showcased in HuggingFace[19].

The choice for the baseline model was the distilled 600M NLLB-200 model, the setup process for this work will be discussed thoroughly in chapter 5.

[17] https://cloud.google.com/translate/docs/languages

[18] https://github.com/facebookresearch/fairseq/tree/nllb

[19] https://huggingface.co/models?search=nllb

4. Data Collection / Preprocessing

Data is the bread and butter of machine learning. This is reflected in machine translation research in such a way that languages are classified based on their available training data as seen in subsection 2.2.3. Therefore we place great importance on sharing the first data collection process for a language. In this chapter, we focus on the building and process of Western Armenian-English parallel corpus as it is one of the main goals of this work. Specifically, we share our data collection process and the data preparation pipeline.

As mentioned previously, this work only focuses on building a parallel corpus for the Western Armenian-English pair, because of the English-centric nature of the natural language processing research community. First, we assess the situation of Western Armenian in terms of potential resources. There is a plethora of written work in Western Armenian coming from its long history, but finding Western Armenian resources for building a parallel corpus with English is quite a challenge. There are two main problems:

1. **Many Western Armenian books and other important documents are yet to be digitized.** There are some ongoing projects about digitizing Western Armenian documents initiated by DALiH and the universities of Armenia, however currently many Western Armenian transferred to the digital realm are in the form of scans, requiring optical character recognition or a similar method to obtain texts. Optical character recognition is the conversion of typed or hand-written text to digital machine-encoded text, which is processable by computers. Currently, many optical character recognition software utilize neural image recognition models. One of the most popular and open-source optical character recognition engines used in research and industry is the Tesseract OCR [71], which also supports the Armenian script. However, possibly due to the lesser amount of data for the Armenian script and its font types, many mistakes have been observed in the digitized Armenian texts.

2. **The rareness of Western Armenian-English bitext.** The divergence of Western Armenian from Classical Armenian began in the 18^{th} century. The main initiative was to standardize the language of worldly matters, precisely translating many works of science, literature, education, etc. to Armenian, however at that time English was not as prevalent as today and many translated works then originated from Latin, French, Italian, and German. After the events of the Armenian Genocide, Ottoman Armenians scattered around the world and consequently, each community in the Diaspora continued translating mainly the works in their local official language. In this regard, there are translations of English-origin books or bilingual Western Armenian-English books written by individuals of the Armenian community in the United States, Canada, Australia, and the United Kingdom. Yet the domain of these

works remains quite limited. During the emergence of the Internet and the dot-com bubble, the bodies of media of various Armenian communities have established a website to share updates within the community on the Internet. However, possibly due to the inability to adapt to new technologies and the issues of presenting Armenian text on websites have led these bodies of media to write the news on their websites only in the local official language for quite a long time. Media in Western Armenian meanwhile continued only in print. After the adaptation of UTF-8 character encoding in the web, both Western and Eastern Armenian have raised their prevalence and nowadays many news, organizational, and foundational websites provide articles in at least two languages, many being English.

These two problems lead to the two-fold processing of data collection. As much of Western Armenian content is not available digitally, it is important to identify and collect the most suitable printed documents for building/extending the parallel corpus, as this content has a larger domain than online content. Digitizing along with aligning is however not a trivial task and requires additional temporal resources or manpower, hence rendering them to be handled in the long term. Instead, utilizing the resources from the world-wide web yields comparable resources more quickly, however, the content suffers from the limited domain, as mentioned above.

The general search for resource candidates was undertaken in various online and physical libraries in Istanbul, Turkey, and Yerevan, Armenia as well as via extensive research throughout the internet. After collecting and choosing the candidates, we preprocess the data with a unified pipeline that handles both printed and digital documents.

4.1. Collecting Printed Documents

As mentioned previously, Western Armenian is a diasporic language that is not officially adopted by any country. This effectively results in the lack of official documents, e.g. constitutions, agreements, and additional reports/protocols of official meetings, which are written in multiple languages, making them a prime resource for building a parallel corpus. This is oftentimes the case for low-resource languages and perhaps one of the reasons why it is identified as such and the alternative resource for parallel sentences are usually found in religious documents such as the Christian Bible as it is one of the most widely translated documents in the world. Although the Bible has a very specific domain that lacks more modern concepts and words, it is a prime resource for low-resource languages as it requires a minimal amount of alignment work, since the content is shaped in verses and all translations can be directly aligned with the verse numbers. The original Bible is a printed document, however, there is a digital Western Armenian version available on the Internet in rich text format, requiring additional shaping for the alignment. Additionally, we have come across once printed but now online issued religious magazine of the Watchtower[1] of the Jehovah's Witnesses, another body of media which has translation in many different low-resource languages.

[1] `https://www.jw.org/en/library/magazines/`

#	Author	Name	#	Author	Name
1	Albert Lavignac	Musical Education	20	Mary Wood Allen	What a Woman Ought to Know
2	Aleksey Tolstoy	Tsar Fyodor Ivanovitch	21	Maurice Maeterlinck	Mary Magdalene
3	H. H. Chakmakjian	Armeno-American Letter Writer	22	Moliere	The Forced Marriage
4	Carlo Goldoni	La Locandiera	23	Moliere	The Impostures of Scapin
5	Emerson E. White	School Management	24	Moliere	The Jealousy of Barbouille
6	Florence Kinsley	Titus, a Comrade of the Cross	25	Moliere	The Love Doctor
7	Franz Lehar	The Merry Widow	26	Mrs. Howard Taylor	One of China's Scholars
8	Franz Schiller	Don Carlos	27	Oscar Wilde	A Woman of No Importance
9	George du Maurier	Trilby	28	Oscar Wilde	Salome
10	Giuseppe Mazzini	An Essay on the Duties of Man	29	Robert Ingersoll	What is Religion
11	Harry Emerson Fosdick	The Meaning of Prayer	30	Samuel Smiles	Self-Help
12	Henrik Ibsen	A Doll's House	31	Victor Hugo	Ruy Blas
13	Henrik Ibsen	Ghosts	32	Voltaire	Azire
14	Jean Anouilh	Antigone	33	Winfield Hall	Sexual Knowledge
15	Jean Racine	Esther	34	Bedros Torossian	How to Become an American Citizen
16	Johann Wolfgang von Goethe	Faust	35	No Author (Berberian Book Store)	Suggestions and Instructions for Those Preparing for American Citizenship
17	Krikor Zohrab	The Voice of Conscience	36	Assyrian Ladies' Church-Loving Association of Worcester	Constitution & By-Laws of the Assyrian Ladies' Church-Loving Association of Worcester
18	Lev Tolstoy	What Men Live By	37	A.R.F. Tzeghagrons	Constitution & By-Laws of the A.R.F. Tzeghagrons
19	Lord Byron	Manfred	38	M. S. Gabriel	Christian Armenia & the Christian Powers

Table 4.1.: List of Works by Foreign Authors whose Western Armenian Translation Exists

The search for further suitable printed documents focuses on mainly finding books of foreign authors in the public domain which also has a translation in Western Armenian. Although finding English-translated works of a Western Armenian author guarantees that the Western Armenian version exists, these are harder to find in common libraries as they do not particularly include Western Armenian literature. During the initial search, the original language of a foreign book was unconstrained, assuming an English translation already exists. After many visits to physical as well as digital libraries, Western Armenian books were identified and scanned, whose English translations exist in an open-source platform like Project Gutenberg. We share a non-complete list of identified English books with an existing Western Armenian translation in Table 4.1. The books originate from different domains such as fiction, non-fiction, education, religion, and philosophy.

Since a great amount of work is required for correcting the output of optical character recognition and for aligning the sentence pairs, we only continued with 2 books from this list to include in the parallel corpus: Armeno-American Letter Writer by Chakmakjian and The Voice of Conscience by Zohrab. The reason for these books as the first choice is due to the fact that they include either bitexts or their original language is Western Armenian and translated to English and thus have a more natural style of Western Armenian. To include more unique domains into the parallel corpus, the focus of further projects about extending the Western Armenian-English parallel corpus should be on digitizing and aligning printed documents.

To sum up, the printed documents of the **Bible**, the **Watchtower** magazine, and the books **Armeno-American Letter Writer** and **The Voice of Conscience** were selected to be further processed and be included in the parallel corpus. Detailed information about each dataset in the corpus will be presented in section 4.4.

4.2. Collecting Web Documents

To identify suitable web documents for the parallel corpus, first potential Western Armenian websites that also provide English content were identified. Wikipedia has a dedicated Western Armenian section that consists of translations or originally written articles. It is a prime candidate, although the parallel contents are usually not direct translations and hence require additional alignment work.

In order to find further websites, an extensive search was conducted which was initiated from the Wikipedia article of "Armenian Newspapers"[2]. We recommend this list as a first step in finding available online Western Armenian textual resources. The search was extended to include organizational websites, in order to widen the domain coverage, as the news sites mainly cover the news domain. This is realized by a forward search in which each external link provided within the "Partners" section of an already collected website gets added iteratively. After getting an extensive list of websites, any website that did not provide parallel content was removed. These include also websites that do provide English and Western Armenian content, but the content is not a direct translation of each other. The identification and filtering were undertaken by manual inspection.

The following websites were identified as candidates: Gulbenkian Armenian Communities[3], a small newsletter about the new projects regarding particularly Western Armenian and its communities; Hamazkayin[4], a multi-seated cultural and educational organization that includes a newsletter and biographies; Hayern Aysor[5], a Western Armenian news website with parallel English content established by the government of Armenia; Houshamadyan[6], a non-profit organization that prepares articles showcasing the daily lives of Ottoman Armenians; and Western Armenian Wikipedia[7], an online encyclopedia extended with the work of voluntaries.

4.3. Data Preparation

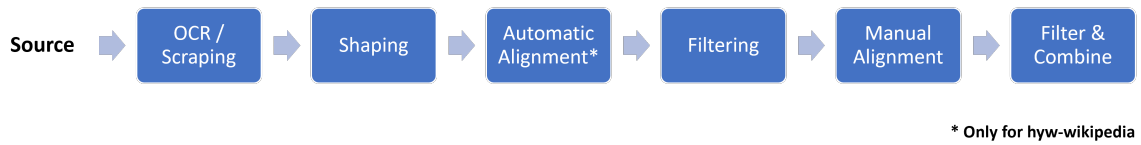

Figure 4.1.: Data Preparation Pipeline

The pipeline for data preparation is shown on Figure 4.1. We continue to explain what has been done in each step specifically.

[2] https://en.wikipedia.org/wiki/Armenian_newspapers
[3] https://gulbenkian.pt/armenian-communities/
[4] https://hamazkayin.com/
[5] https://hayernaysor.am/en
[6] https://www.houshamadyan.org/home.html
[7] https://hyw.wikipedia.org/

4.3.1. OCR / Scraping

To obtain a digital textual form of printed documents we used Tesseract OCR Engine, during installation the Armenian module is not installed as default. To interact with Tesseract via Python scripts we used the pytesseract[8] library. The scans of books are kept in PDFs, which then were converted into a set of images of pages. Each page was identified manually to contain only English, Armenian, or both and given to the Tesseract with the according parameter. The outputs were saved based on their page number.

For the collection of website documents, an individual scraping script was written for each candidate website. The scripts utilize BeautifulSoup4[9], trafilatura [21] and Selenium[10] libraries to collect relevant information from each page and to navigate between pages. Each script specializes in its assigned website but follows the same two-step logic. The script takes the website link of the first page of the list containing every Western Armenian document's link. It starts traversing each Western Armenian document by identifying and matching the corresponding English document. Then for each matched link pair the contents are collected, irrelevant content such as navigational links are removed and the content of each pair is saved by the Western Armenian title of the document followed by the language code (Ex. Լուր.hyw / Լուր.en).

4.3.2. Shaping

Shaping includes the normalization and segmentation within each document. For printed documents, each page's content was restructured to include only a single chapter of the document. During this step, any mistakes generated by the OCR were identified by side-to-side comparison and corrected.

The data for machine translation is shaped in such a way that each sentence within the data is written on a separate row. To obtain such a shape, the sentences within each document should be identified and written separately. For this, we used the NLTK [23] library for the English sentence segmentation which utilizes a neural model, whereas for the Western Armenian sentence segmentation the pySBD [122] library which has a rule-based approach. The list of characters to find the Armenian sentence boundaries was extended by the colon (":", Armenian has a separate character for sentence boundary ":", which resembles the colon, but in many websites, both characters are used), and the ellipsis "...".

4.3.3. Automatic Alignment

This step is performed only for the Wikipedia dataset as its documents are not direct translations of each other. For such texts, bitext mining methods are usually utilized. As explained in subsection 3.2.1 these methods employ word embeddings which Western Armenian does not have. Therefore, we created a workaround by translating Western

[8] https://github.com/madmaze/pytesseract
[9] https://www.crummy.com/software/BeautifulSoup/
[10] https://github.com/SeleniumHQ/Selenium

Armenian sentences into English and then by comparing each translated Western Armenian sentence with each English sentence appearing in the same document. We used Google Translate API[11] and gave the Western Armenian sentences as if they were Eastern Armenian. Since the languages are similar the general meaning of the sentence is translated correctly, the issues appear mainly in the tense/mood of the verbs. This is already known and used within the Western Armenian community to generate English translations of Western Armenian content. After obtaining the English translations of each Western Armenian sentence, each translation is compared with every English sentence in the document for similarity using NLTK's [23] cosine similarity function over the sentence embeddings and any English sentence with a similarity score over 0.75 was included as the candidates for the Western Armenian sentence. The Western Armenian sentences with no candidates were discarded.

4.3.4. Filtering

This step includes the removal of content that is not beneficial for the learning of the translation task, such as URLs, Emojis, and long sequences of numbers.

4.3.5. Manual Alignment

Since Western Armenian does not have any kind of corpus for natural language processing research, it is important to build one with higher quality. To ensure this quality the corpus must be crafted by an individual who has a certain level of mastery in that language. We think the first corpus for a language must be crafted manually to ensure a baseline and afterwards, this corpus must be extended with the modern tools. Although it is a substantial amount of work, the direct alignment of sentences will ensure a small but high-quality parallel corpus. This is the reason why the chosen candidates for resources are made of documents that are direct translations of each other. Otherwise, the manual aligning phase would last implausibly long.

The manual alignment of the 5430 Wikipedia documents (ca. 450k parallel lines) took about 2,5 months of intensive work. During which sentences were compared side-by-side. Any one-to-many, many-to-one, and many-to-many alignments were restructured to be within one row in the document. For the Wikipedia dataset, the alignment was done for each Armenian sentence from the set of candidates.

4.3.6. Filtering and Combination

The sentence pairs with a significant difference in their lengths were filtered. Upon inspection the ratio of $\frac{1}{2}$ was used for both sides. Afterwards, all documents within a dataset were combined into one file with the name of the dataset followed by the language code (e.g. wikipedia.en / wikipedia.hyw).

[11] `translate.googleapis.com`

4.4. Western Armenian Datasets

Each dataset was subdivided into training, validation, and testing parts. The training subset is the data that the model sees and trains upon. The validation set is to assess the performance of the model after each epoch during training. The testing set is used to measure the model's general performance and therefore will not be seen by the model during training. The sizes for validation and test sets were defined to be 2000 sentences. Any dataset that does not have 2000 sentences each for test and validation and has a unique domain instead has a test and validation set size of 10% of the whole dataset size. If the domain of a dataset is covered by another dataset, it will not be included in the training data, but rather be used only in testing. There are a total of 9 datasets that make the first Western Armenian-English parallel corpus. Table 4.2 illustrates the statistics of each Western Armenian dataset within the parallel corpus and Table 4.3 showcases typical sentences from each dataset revealing their styles.

4.4.1. Armeno-American Letter Writer (AALW)

Armeno-American Letter is a guide by Haroutioun Hovannes Chakmakjian on how to write letters for different situations, published 1913. The book is a typical example of a bitext as the left-hand side pages of the book are written in English and the right-hand side in Western Armenian. It teaches by example about how to write business, familiar, love letters as well as job applications; providing a unique domain of formal and informal correspondences for the translation model. Also, it includes some uncommon phrases both in Western Armenian and English. Although this dataset does not have adequate examples to build test and validation sets, we still include it because of its uniqueness. The AALW dataset has 1730 examples for training, 192 for validation, and 213 for test datasets.

4.4.2. Bible

In multilingual machine translation training datasets, the Bible is often included since not only it is the most translated text in the world, but also its verse numbers make the alignment work trivial. For the Western Armenian side, the 1853 version was used whose online version includes both the Old and the New Testament. For English, the Modern English version is used. The religious domain that the Bible covers is though limited, it captures many names of people and places. The Bible dataset has a total of 30604 examples.

4.4.3. Gulbenkian Armenian Communities

Calouste Gulbenkian Foundation is a non-profit foundation that promotes various art, science, and educational projects. It also has a dedicated branch called "Armenian Communities" which supports students of Armenian origin with scholarships and sponsors various projects for the preservation and development of (Western) Armenian language. The dataset was prepared from the direct translations of the news articles on the webpage of Gulbenkian Armenian Communities. Along with personal names and their English transliterations, the dataset includes also modern words for many technological concepts

like “app”, “download” etc. Since the news domain will be covered by a larger dataset, this dataset will be only used in test sets. This dataset has a total of 598 examples.

4.4.4. Hamazkayin

Hamazkayin Armenian Educational & Cultural Society is a major cultural organization with multiple seats across the Armenian Diaspora. Hamazkayin organizes and supports many cultural events, such as art exhibitions, music festivals, seminars, lectures, book/film review nights, etc., and additionally involves in the publishing of a monthly literary magazine and various books. The Hamazkayin dataset was prepared from the news articles reporting the events hosted or sponsored by Hamazkayin as well as reviews about many books and films. Hamazkayin’s website also includes a subsection of profiles, where short biographies of persons who had an impact in the Armenian Diaspora are presented. The dataset covers the domains of art, literature, biographies, and news. Names of countries and cities are very prominent in the dataset which has a total of 10739 examples.

4.4.5. Hayern Aysor

Hayern Aysor (Armenians Today) is a news website established by the government of the Republic of Armenia. The news articles are released in 4 different languages: Eastern and Western Armenian, English, and Russian. Upon inspection, the Western Armenian articles seem that they were translated from Eastern Armenian rather than being written on their own. This dataset has unique style with phrases from Western and Eastern Armenian included in the texts. The dataset also covers the official and governmental domains since most of the articles are the official statements of various ministers of the Republic of Armenia. The dataset includes a total of 5422 examples.

4.4.6. Houshamadyan

Houshamadyan is a Berlin-originated non-profit association dedicated to preserving and showcasing the everyday lives of the Armenian communities within various cities and the countryside of the Ottoman Empire. The website includes pages, each of which is either dedicated to a city, a village, or a subdistrict in the Ottoman Empire. In these pages, articles about various themes like local characteristics, education, economy, literature, traditions, clothing styles, recipes of local dishes, etc. are presented in 3 languages: Western Armenian, English, and Turkish. Aside from its wide coverage of domains, the dataset also includes a subsection of image captions in which many historical names of places as well as many Armenian personal names both with their English transliterations can be found. The dataset has a total of 38267 sentences.

4.4.7. The Watchtower Magazine of Jehovah’s Witnesses

This is another massively translated body of media that has included Western Armenian for many years. The magazine includes articles about biblical prophesies, morals and values, the history of religion, the Bible, and Jehovah’s Witnesses. The magazine is published

both in print and online. Although the Watchtower is a religious magazine, it also includes contemporary topics like internet usage, etc. as well as personal stories, rendering it a multidomain resource. With its reach to worldwide communities, many names of persons of different nationalities and languages can be found with their respective Western Armenian transliteration. Also, its large size makes this dataset a desirable resource. The dataset is comprised of the articles of the Watchtower magazines issued between the years 2018-2023 and has a total of 54323 examples.

4.4.8. The Voice of Conscience

This is a collection of several short stories written by Krikor Zohrab, an influential Armenian writer and politician of the late 19th and early 20th century Ottoman Empire. It includes stories that depict the daily lives of people of many different social classes in Istanbul and is written in a realist manner, another unique domain covering the fictional story domain. The Western Armenian version is available in the digital library of the American University of Armenia, for the English version, the book translated by Jack Antreassian and published by the Krikor and Clara Zohrab Information Center was used. The translation heavily focuses on maintaining a literary aesthetic and therefore significantly differs stylistically from the texts in the other datasets. Its size, 889 examples, is not enough to build the subsets for training, test, and validation; therefore this dataset will completely be used for testing.

4.4.9. Western Armenian Wikipedia

In machine translation research, Wikipedia articles are perhaps the most utilized resources, probably due to their open-source nature and wide coverage of topics (e.g. biographies, science, popular culture, history, art, etc.) in many different languages. The translations of an article are written individually rather than being translated from each other, thus a bitext mining method is required to find and align sentences. For the Wikipedia dataset therefore the additional step of automatic alignment described in the previous subchapter was performed. In the manual alignment step, the English pair for each Western Armenian sentence was selected from its individual candidate pool. Wikipedia articles are often used as training data for multilingual machine translation models since many articles are written in multiple languages. This is also the case in Western Armenian and for relatability we include this dataset. Wikipedia dataset has the widest coverage in relation to the other datasets and includes a total of 7979 examples.

4.4.10. Western Armenian Monolingual Dataset

There is also a Western Armenian monolingual dataset built from multiple Western Armenian news websites. Two of which originate from Istanbul, Turkey: Jamanak and Agos; one from Beirut, Lebanon: Aztag, and one from Yerevan, Armenia: Arevelk. These datasets will be utilized after the first machine translation model for Western Armenian-English is obtained and the English sentences will be synthetically generated. Since there are already parallel datasets covering the news domain, the generated sentences will

keep the style of news articles. This dataset is included to extend the Western Armenian examples. The combined monolingual dataset is called HYW-Mono and includes 1.437.035 examples.

Dataset Name	Domain	# Examples
AALW	Correspondences (Formal & Informal)	2135
Bible	Religious Texts	30604
Gulbenkian	News, Technology	598
Hamazkayin	News, Culture, Art, Literature, Education, Biographies	10739
Hayern Aysor	News, Governmental, Official	5422
Houshamadyan	Sociology, Culture, Education, Food Recipes, Captions, Personal Stories	38267
Watchtower	Religion, Culture, Personal Stories, Philosophy	54323
VoC	Literature, Fictional Stories	889
hyw-Wikipedia	Biographies, Art, Science, Education, Literature, Geography, History, Popular Culture	7979
	TOTAL Parallel Corpus	**150956**
HYW-Mono	News, Literature, Philosophy, Religion, Sports	1437035
	TOTAL	**1587991**

Table 4.2.: Western Armenian Datasets

Dataset	Example Sentence
AALW	Your letter of April 24 reached me two days ago, but, not wishing to answer it hastily, I have delayed writing until today.
	I am much grieved that you should think me capable of wavering in my affection toward you or wilfully inflicting a slight upon one in whom my whole hope of earthly happiness is centered.
Bible	God blessed them, saying, "Be fruitful and multiply and fill the waters in the seas, and let birds multiply on the earth."
	Embarking on a ship of Adramyttium that was about to set sail to the ports along the coast of Asia, we put to sea, accompanied by Aristarchus, a Macedonian from Thessalonica.
Gulbenkian	The Western Armenian Treebank, and the Natural Language Processing solutions developed on its basis, are decisive in bringing state of the art language technologies to Armenian, ensuring the vitality of the language in the modern digital era.
	Children acquire new vocabulary, learn or improve their knowledge of Western Armenian and, importantly, find a new and exciting world in the language that speaks to their daily lives in the Diaspora.
Hamazkayin	The 98-year-old patriarch of literature of the Armenian Diaspora Jack S. Hagopian that has tirelessly worked and served almost eight decades and is always full of youthful vigor and enthusiasm, declared these words on his tribute.
	Tankian is not only a member of the band System of a Down, but also has his own record label which is called Serjical Strike Records, a subdivision of Columbia Records.
Hayern Aysor	In response, the Deputy Prime Minister said that the Prime Minister's statement did not remain unanswered: some RPA representatives made statements in this regard which the Deputy Prime Minister offered to familiarize with, and added that, if necessary, the Prime Minister will provide details in the future.
	The Consulate General of the Republic of Armenia in Erbil informs that the Consulate General of the Republic of Armenia opened in Erbil on March 1, and therefore the Embassy of the Republic of Armenia in Baghdad will no longer accept visa applications from Iraqi Kurdistan.
Houshamadyan	The belt was made in Van, and sports two maker's marks, one being the image of a lion, which means the belt was the work of master craftsman Arslanian, and the engraved letters "G" and "T," which attest to the fact that a second craftsman also worked on it.
	First row, left to right: Hayg; Mardiros (the father of the family); Daniel, sitting right in front of his father; Vartouhi, Shushan (nee Jamgochian, Mardiros's wife).
Watchtower	You have repented of your sins; you feel deeply sorry for the wrongs you have committed, and you have asked Jehovah for his forgiveness.
	For example, keep away from Internet websites and entertainment that feature wrong conduct. (Eph. 5:3, 4)
VoC	Husep Agha was turning his problems over in his mind, trying to find a solution for them; as he walked through the streets, feeling the emptiness of the bag, stroking it with his fingers, he felt himself transported to the side of his beloved children, and there transformed for a moment, he managed to forget his beggarly circumstances, picturing himself as rich and omnipotent.
	Khazar Agha's son could not resist this attraction for very long. He was a self-indulgent young man, but he had learned like his father to keep his longings well hidden, and for that reason perhaps was less able to control them whenever he happened to be alone with the maid.
hyw-Wikipedia	Zabel Yesayan (4 February 1878 – 1943) was an Armenian writer and a prominent figure in the Armenian academic and political community during the late nineteenth and early twentieth centuries.
	Brass, an alloy of copper and zinc in various proportions, was used as early as the third millennium BC in the Aegean area and the region which currently includes Iraq, the United Arab Emirates, Kalmykia, Turkmenistan and Georgia.

Table 4.3.: Typical sentences from each Western Armenian dataset

4.5. Eastern Armenian Datasets

To experiment with the impact of knowledge transfer from Eastern Armenian models to the quality of Western Armenian-English translation as well as the zero-shot performance of Eastern Armenian models we trained translation models for Eastern Armenian-English pair for which we used several parallel corpora obtained from OPUS. Table 4.4 shows an overview of the Eastern Armenian-English parallel corpora.

Dataset Name	Domain	# Examples
Bible-Uedin [34]	Religious Texts	13023
CCAligned [83]	Informal, Web Documents	1012653
hye-Wikipedia	Biographies, Art, Science, Education, Literature, Geography, History, Popular Culture	20480
Neulab-Tedtalks	Formal, Informal, Education, Technology	22116
Opensubtitles [94]	Informal, Popular Culture	3287
QED [1]	Education, Science	36447
Tatoeba	Informal, Education	2157
TED2020 [117]	Formal, Informal, Education, Technology	36252
XLent [84]	Web Documents	208902
	TOTAL	**1355317**

Table 4.4.: Eastern Armenian Datasets

5. Experiment Setup

To answer the research questions and have an assessment of Western Armenian machine translation we conduct several experiments. We design our experiments by first building up scenarios, that resemble real-life situations and try to answer more practical questions. The scenarios aim to create a general picture Western Armenian's low-resource setting. To gain more technical knowledge we extend our experiments by in-depth analyses. These are experiments that investigate a special aspect within datasets or a training scheme. Having a two-fold analysis allows us to scout through the uncharted territory of Western Armenian translation and make an overarching map, meanwhile the in-depth analyses sharpen the specific regions within this map. We train a large number of models with different configurations and each model is assigned to a single or multiple scenarios or to an in-depth analysis experiment. The training configurations mostly regard to the chosen training data or training scheme. Each scenario and in-depth analysis experiment is introduced in their dedicated section.

As mentioned previously, we make the translation models by finetuning the pretrained NLLB200-600M-Distilled multilingual neural machine translation model with our data instead of starting from scratch. This way we leverage the multilingual knowledge of the model provided by Team NLLB [133]. We chose the smallest model as it requires less memory. The pretrained model is also used as a baseline in most of the scenarios, allowing us to assess the performance on Western and Eastern Armenian with custom test datasets. Team NLLB [133] reports state-of-the-art level performances of both X $\rightarrow$ EN and EN $\rightarrow$ X in a low-resource language setting. The model provided by Team NLLB [133] supports Eastern Armenian and we think it is beneficial to utilize the gained multilingual knowledge, because not only the model will see more examples of Eastern Armenian which we hypothesize that it is beneficial for Western Armenian, but also the pretrained model has a larger knowledge of the English language since it was trained with a dataset that has more English examples than the combined datasets that we provide. In order to realize the knowledge transfer from the pretrained model about Eastern Armenian, we mark all the (Western and Eastern) Armenian sides in the datasets as Eastern Armenian (`hye_Armn`). This ensures the learned parameters for Eastern Armenian tokens are utilized directly instead of having to learn the mapping of Eastern Armenian tokens to the Western Armenian language. This effectively causes the model to lose the support for translating Eastern Armenian, however, this is out of this work's scope and this could be solved easily by introducing the language identification token for Western Armenian (`hyw_Armn`) to the model.

The evaluation is performed on the best version of each individual model. The best version of a model is determined by comparing the negative log-likelihood losses of the model after every epoch and choosing the version with the best score.

5.1. Default Parameters for Models

The models use the default training parameters listed in Table A.1 of section A.1 unless exclusively mentioned in a scenario or in-depth analysis experiment.

5.2. Scenario 1: No Parallel Western Armenian Data During Training

In this scenario we want to capture the situation before introducing our parallel corpus. Comparable to a field survey, we want to check and analyze what the models can already do in terms of Western Armenian machine translation with their multilingual knowledge that also includes Eastern Armenian. Western Armenian parallel data is quite hard to find and we want to analyze the models' Western Armenian translation performance when they are trained with relatively easier-to-find data such as Eastern Armenian parallel data and monolingual Western Armenian data. This scenario also involves in answering RQ2 as some models are trained exclusively with Eastern Armenian data and puts the similarity of the languages under the lens.

The training datasets of the models included in this scenario are designed in such a way that they become more similar to the Western Armenian parallel corpus. We start with the pretrained NLLB model with no additional data to see how the state-of-the-art model performs on Western Armenian, then the model trained with the available Eastern Armenian parallel corpus, the model trained with a synthetic Western Armenian parallel corpus whose Armenian side is the Western Armenian monolingual dataset and the English sides are generated by the model that has been trained with Eastern Armenian data; and finally the model with a synthetic Western Armenian parallel corpus, but this time the Armenian sides originate from the genuine Western Armenian parallel corpus and the English sides are generated by the same model mentioned previously. Each model's training dataset effectively represents a quick and easy solution to support Western Armenian translation as they are either already available (NLLB and Eastern Armenian parallel corpus) or do not require additional alignment work and relies on the Eastern Armenian knowledge to generate a parallel corpus (monolingual corpora). With this gradual approach we aim to obtain a clearer picture about the impact of each dataset.

The evaluation within this scenario is performed under 3 subcategories. 1) The general performance on the whole Western Armenian-English testset; 2) On each individual supervised testset from the Western Armenian-English parallel dataset, i.e. the sub-testsets whose corresponding trainsets are included in training; 3) On each individual unsupervised testset, i.e whose testsets are not included in the training. Evaluation in this manner allows a better assessment of the models' performance; the first subcategory shapes the overview, the second focuses more on the domains represented by the subsets, and the last one gives insight into the general performance where the input will not be seen previously by the models during training.

3 models are subjected to comparison in this scenario, 2 additional models are included to serve as baselines:

1. **NLLB:** This is the model built by the NLLB team [133] without any further finetuning. It will be used as a baseline to illustrate what is the performance without any additional data and what it can already do in terms of Western Armenian machine translation with its multilingual knowledge.

2. **NLLB + HYE:** The baseline NLLB model is finetuned with the Eastern Armenian-English parallel trainset. It represents how the Eastern Armenian additional knowledge affects the Western Armenian translation quality.

3. **NLLB + DA-HYE$_{\text{hyw-mono}}$:** The baseline model is finetuned with a synthetic Western Armenian-English parallel dataset, whose Armenian side originates from the Western Armenian monolingual dataset, and the English side sentences are generated by the NLLB + HYE model. This is especially interesting since it simulates the impact of a quickly and easily generated training set requiring only monolingual sentences and a pretrained model that supports Eastern Armenian.

4. **NLLB + DA-HYE$_{\text{hyw}}$:** The baseline model is finetuned with a synthetic Western Armenian-English parallel dataset that uses the Western Armenian side of the genuine parallel dataset and the English sentences are generated by the NLLB + HYE model. As previously mentioned, this model is intended to assess the performance of the model that is trained with Eastern Armenian translating Western Armenian and generating English sentences. Since the domains within the test and trainset will be identical, we predict the effect on domain mismatch will be less than the 3rd model in this scenario.

5. **NLLB + HYW:** The baseline model is finetuned with the genuine Western Armenian-English parallel dataset and serves solely as an upper baseline (oracle) in this scenario to illustrate how close the models in this scenario get to the performance of this model.

5.3. Scenario 2: Parallel Western Armenian Data During Training

After performing the field survey in Scenario 1, we continue on by introducing the Western Armenian parallel corpus to the models. This scenario serves to inspect the improvement of Western Armenian translation quality by the inclusion of the parallel corpus and other additional corpora such as Eastern Armenian parallel and Western Armenian monolingual corpus with synthetic English sides. The scenario focuses on the different training schemes and choices of training data to identify what is effective in the case of Western Armenian, meanwhile finding answers for RQ1 as we employ data augmentation and several training schemes such as joint and double finetuning; as well as for RQ3 with the doubly finetuned model that first finetunes upon Eastern Armenian examples and then extends this knowledge with Western Armenian examples in the second finetuning session.

Both scenarios are linked and designed to serve the narrative of what was already being done, how relatively easily accessible Eastern Armenian data serve in Western Armenian

machine translation and what gets improved and how much the translation quality is improved when a parallel corpus dedicated to the Western Armenian language is built. The evaluation follows the same categorization described in the previous scenario. A total of six models are included in this scenario for the main comparison, one serves as the baseline:

1. **NLLB:** Baseline pretrained model made by the NLLB team.
2. **NLLB + HYE:** Baseline model finetuned with Eastern Armenian-English parallel trainset, this is included to illustrate a better comparison with the relevant models.
3. **NLLB + HYW:** Baseline model finetuned with Western Armenian-English genuine parallel examples. This model serves to illustrate the general effect of the Western Armenian parallel corpus on the translation quality.
4. **NLLB + {HYE, HYW} (Joint Finetune):** Baseline model finetuned with a balanced combination of Eastern Armenian-English and Western Armenian-English genuine parallel examples. With this model, we investigate the impact of Eastern Armenian knowledge gained by the joint learning which is primarily compared to the double finetuning.
5. **NLLB + DA-HYW$_{\text{hyw-mono}}$:** Baseline model finetuned with Western Armenian-English synthetic parallel trainset whose English sides are generated by the NLLB+HYW model. This serves to show both the quality of the generated English sides as well as the impact of additional monolingual Western Armenian data.
6. **NLLB + {HYW, DA-HYW$_{\text{hyw-mono}}$}:** Baseline model finetuned with a balanced combination of genuine and synthetic Western Armenian-English examples. This model serves to make a clearer picture comparing the cases of sole inclusion of synthetic parallel data and the joint inclusion of it with the genuine examples.
7. **NLLB + HYE + HYW (Double Finetune):** Baseline model finetuned with Eastern Armenian-English examples and subsequently with Western Armenian-English genuine examples. With this model, we investigate how the gained Eastern Armenian knowledge in the first finetuning session affects the Western Armenian translation quality in contrast to joint finetuning.

5.4. In-Depth Analysis 1: Impact of Domain vs. Language

Texts of different domains are written in different literary styles using unique vocabularies. In order to have an acceptable translation quality of a text from a certain domain, the machine translation system should usually see examples from the same domain. For example, if a model is trained only on geographical texts, the model will not be able to disambiguate the meaning of the word "bank" in a text about economics and will output the wrong translation.

Since domains have different themes and topics, they can be more or less similar to each other. The more similar the two domains are, the shared vocabulary and concepts

increase among these domains. In this sense the domain of mathematics is more similar to the domain of physics than for example the domain of dancing. Languages also show similarities with each other and as the Modern Standard Armenian variants have the same origin and they share many aspects, one can conclude that these languages are similar.

In this experiment, we want to assess the sensitivity of a data subset towards linguistic and domain similarity. This brings insight into which information is more important for improving the translation of a sentence from this subset and can be a guide for future projects' data collection process. The information yielded by the results of this experiment is valuable while investigating RQ3 since we gain insight about how inter-language knowledge fares against inter-domain knowledge. We are able to realize this experiment with datasets of the same origin (Bible and Wikipedia) in both languages.

For each domain of the Bible and Wikipedia, a total of 6 models were trained. Each model represents a (mis)match in language or in domain or a combination of both. The models will be evaluated by the examples of their assigned domain. Table 5.1 illustrates the representations of models for each domain in evaluation, where the models NLLB + (HYW-Bible/HYW-Wiki) represent the models that are finetuned on the baseline NLLB model with the Western Armenian Bible dataset and Western Armenian Wikipedia dataset respectively, NLLB + (HYE-Bible/HYE-Wiki) are the analogous models that are finetuned with Eastern Armenian counterparts; NLLB + HYE, HYW-X represent the models that are trained with the combined dataset of the whole Eastern Armenian-English dataset and Western Armenian X subset.

#	Representation	Corresponding model for evaluating Western Armenian Bible	Corresponding model for evaluating Western Armenian Wikipedia
1	Matching Domain & Language	NLLB + HYW-Bible	NLLB + HYW-Wiki
2	Matching Domain Mismatching Language	NLLB + HYE-Bible	NLLB + HYE-Wiki
3	Mismatching Domain Matching Language	NLLB + HYW-Wiki	NLLB + HYW-Bible
4	Mismatching Domain & Language	NLLB + HYE-Wiki	NLLB + HYE-Bible
5	Combination of 2 & 3	NLLB + {HYE-Bible, HYW-Wiki}	NLLB + {HYE-Wiki, HYW-Bible}
6	Whole mismatching language data & 3	NLLB + {HYE, HYW-Wiki}	NLLB + {HYE, HYW-Bible}

Table 5.1.: Various subscenarios with their corresponding models in each domain in evaluation.

As there are more or less favorable models for each domain, we expect a comparable pattern to emerge in both cases, which will indicate the importance of language or the domain for each domain.

5.5. In-Depth Analysis 2: Impact of Segmentation in Knowledge Transfer

As RQ3 suggests, we want to utilize Eastern Armenian knowledge while translating Western Armenian texts. As an adequate amount of resources lacks for Western Armenian we want to maximize the transferrable knowledge from Eastern Armenian. To be able to do that any knowledge prior to learning from Western Armenian-English resources

must be in a form that does not get subjected to too much reshaping. A modern neural machine translation model "learns" to translate by looking at sequences of tokens and optimizing its parameters according to the seen tokens in the example. To be able to transfer knowledge in a double finetune training session the knowledge within the first session must be acquired upon the tokens that are also available in the examples of the second session. In order to maximize the knowledge transfer the examples of both sessions must be tokenized in such a way that the token vocabularies of the training data of the first and the second session overlap as much as possible. For example, if the word apple is tokenized as "a pp le" in every occurrence within the training data of the first session and as "app le" in the second one, the knowledge gained for the tokens "a" and "pp" will not be considered in the second session. However if they were tokenized identically, every example the knowledge about the tokens gained in the first session will be reused in the second session.

The method can be realized by encoding the datasets of both sessions with the same vocabulary. Some rare tokens of one dataset may not be in the other dataset, however, if the datasets share the same script, then each single rare token can be replaced with smaller more common token sequences, effectively increasing the overlap. The overlap can converge to a certain value and the complete overlap of two datasets may not be achieved if there are tokens that do not appear in both datasets cannot be swapped with any other more common tokens.

Example: Որտեղ ե՞ն բոլորը: (Where is everyone?)

Threshold	Tokenized Sent.	Notes
0	▁Ո րտ ե՞ ղ ▁են ▁բոլորը ։	Any token within the vocabulary can be used.
1000	▁Ո րտ ե ՞ ղ ▁են ▁բոլորը ։	Any token with an occurrence less than 1000 is replaced.
50000	▁Ո ր տ ե ՞ ղ ▁են ▁բ ո լ ո ր ը ։	
1000000	▁Ո ր տ ե ՞ ղ ▁ե ն ▁բ ո լ ո ր ը ։	There are no tokens with an occurrence more than 1 million, the sentence is effectively tokenized by characters.

Table 5.2.: Tokenization of an example Eastern Armenian sentence with different vocabulary threshold values

In our case, we want to maximize the knowledge transfer from Eastern Armenian to Western Armenian within the double finetuning session. To achieve that we utilize several custom-encoded versions of both parallel datasets and train models on only Eastern Armenian-English data and on first Eastern Armenian- then Western Armenian-English data. The former models will be used in the evaluation of translation quality of Eastern Armenian-English texts (to see how the custom encoding affects) and Western Armenian-English texts (to see the impact of custom encoding in zero-shot case); whereas the latter models will be evaluated only on the Western Armenian-English testset (to see the impact of custom encoding in double-finetuned case). The custom encoding is realized by sentencepiece[1] and performed only on the Armenian sides since this is not needed in the English side. We chose the Eastern Armenian-English trainset to generate the common vocabulary as it is the largest dataset among the datasets that will be seen by the models during

[1] https://github.com/google/sentencepiece

the training. To determine the overlaps, the Eastern Armenian- and Western Armenian-English testsets and Western Armenian-English trainsets were encoded using the common vocabulary. Then for each subset, the total number of tokens, types, the number of tokens, and types in and out of the intersection with the Eastern Armenian-English trainset were calculated. In order to increase the overlap, all the subsets were encoded with different `vocabulary_threshold` values, and the aforementioned occurrences were calculated again for each encoding with a different threshold value. `vocabulary_threshold` forces any token that occurs less frequently than the threshold to be replaced with more common tokens. If the threshold gets too large, the examples will be encoded as characters. Table 5.2 shows an example sentence encoded with different threshold values.

Evaluated on:	HYE-train		HYE-test					HYW-train					HYW-test						
Thresholds	Tokens	Types	Tokens	Types	Type Int. w/ HYE-train	OOV Types	OOV Tokens	Tokens	Types	Type Int. w/ HYE-train	OOV Types	OOV Tokens	Tokens	Types	Type Int. w/ HYE-train	OOV Types	OOV Tokens	Type Int. w/ HYE- ∪ HYW-train	OOV Tokens	OOV Types
0	35464240	54976	365170	7545	7537	8	9	4952154	9731	9657	74	318	545748	5234	5201	33	82	5222	12	29
50	35831612	20791	366601	6751	6743	8	9	4981782	7147	7107	40	189	547965	4649	4635	14	31	4642	7	22
100	36139845	17451	367700	6072	6064	8	9	5005419	6034	5994	40	189	550074	4255	4241	14	31	4248	7	22
500	37645722	12920	373822	3941	3933	8	9	5107199	3652	3612	40	189	559238	3153	3139	14	31	3146	7	22
1000	39010068	11967	382973	3093	3085	8	9	5292187	2875	2835	40	189	575722	25880	2574	14	31	2581	7	22
5000	45746032	10460	452212	1606	1598	8	9	6447531	1568	1528	40	189	699930	1393	1379	14	31	1386	7	22
10000	53206875	9876	524900	1025	1017	8	9	7156796	1036	996	40	189	784914	884	870	14	31	877	7	22
50000	74741940	9399	771308	548	540	8	9	9531926	578	538	40	189	1056571	447	433	14	31	440	7	22
100000	88032605	9343	926201	492	484	8	9	11435436	522	482	40	189	1271830	392	378	14	31	385	7	22
500000	1.07E+08	9318	1116170	467	459	8	9	13141592	497	457	40	189	1478713	367	353	14	31	360	7	22
1000000	1.08E+08	9317	1125176	466	458	8	9	13185205	496	456	40	189	1483915	366	352	14	31	359	7	22

Table 5.3.: Token-Type Statistics of Custom Encoded Datasets; the statistics of the chosen `vocabulary_threshold` values for further training are underlined

Table 5.3 illustrates the statistics of the process described in the previous paragraph. Each column is designated to represent the training or test set of Eastern Armenian-English or Western Armenian-English parallel corpus. For Eastern Armenian training set only the token and type amounts are listed according to the threshold values. The threshold of 0 is the case where any token within the Eastern Armenian dataset is allowed. For the other sets the statistics for the type intersection, i.e. the amount of shared tokens with the Eastern Armenian training set, out-of-vocabulary types and tokens, i.e. the amount of tokens and types that do not appear in the intersection, are listed. For Western Armenian test set, the intersection statistics are listed for both Eastern Armenian training set only and for the union of Eastern and Western Armenian training sets to represent the zero-shot case, i.e. evaluating Western Armenian test sentences on the models that are trained only on Eastern Armenian training data and double finetuning case, i.e. evaluating the test sentences on the models that have seen both Eastern and Western Armenian training sets. The statistics show that the tokens and types have already a high overlap (lowest overlap 99.2% of HYE-train ∩ HYW-train) with the threshold value 0 and the overlap seems to converge and not change after the threshold value 50. This is caused by the tokens that do not have an alternative representation and therefore left out of the intersection. However,in order to clearly investigate the effect of custom tokenization, we choose to train the models for the threshold values of 50, 1000, and 50000. The statistics for these values are indicated with an underline in Table 5.3.

6. Evaluation

The models are evaluated with the BLEU and chrF3 metrics, which were described in paragraph 2.2.2.6 and paragraph 2.2.2.6 respectively. We also provide chrF3 scores, since BLEU only regards exact matches between hypothesis and reference, making a too strict of a metric for inflective languages such as Armenian. chrF3 in this regard is more flexible because the comparison is done within the character level, allowing incomplete matches to receive some partial marks.

The evaluation is performed under the scenarios described in the previous chapter.

6.1. Scenario 1: No Parallel Western Armenian Data

6.1.1. Evaluation on the Combined Testset

Evaluated on:	**HYW-test COMBINED**			
Direction	**EN → HYW**		**HYW → EN**	
Model \ **Score**	**chrF3**	**BLEU**	**chrF3**	**BLEU**
NLLB	*34.9*	*2.2*	*47.8*	*20*
+ HYE	36.4	2.2	**50.1**	20.3
+ DA-HYE$_{\text{hyw-mono}}$	45.6	7.8	49.8	**20.7**
+ DA-HYE$_{\text{hyw-par}}$	**51.5**	**13.5**	49.8	20.5
+ HYW (Oracle)	*54*	*17*	*57.2*	*29.4*

Table 6.1.: Evaluation scores of the models within the scenario "No parallel Western Armenian data" on the combined Western Armenian testset.

To provide an overview of the zero-shot scenario, Table 6.1 illustrates the results on the combined Western Armenian testset. In EN → HYW direction, the baseline model has a chrF3 score of 34.9 and a BLEU score of 2.2, the lowest scores among other models. This is mainly caused by the divergence in outputs being in Eastern Armenian while the references being in Western Armenian. Qualitative investigations have shown that the similarity of the languages does not seem to have much effect on the improvement of the translation quality. A qualitative example is given below:

English Reference: Hagop Haroutiun Ohanian (born in Adana circa 1881 - died in Baghdad in 1923)
Western Armenian Reference: Յակոբ Յարութիւն Օհանեան (ծնած Ատանա մօտ 1881-ին - մահացած Պաղտատ 1923-ին)
Eastern Armenian Hypothesis: Հակոբ Հարություն Օհանյան (ծնվել է Ադանայում 1881 թ. - մահացել է Բաղդադում 1923 թ.)

Figure 6.1.: Illustrative Example 1 - Orthographical Mismatch

Although the reference in Figure 6.1 is semantically translated almost perfectly (the hypothesis lacks the translation of "circa"), the hypothesis will receive a sentence BLEU score of 0, as there is not even a single unigram overlap besides the character "-". The chrF3 scores are thought to counteract this exact phenomenon.

Focusing on the models trained only with Eastern Armenian data (NLLB and NLLB + HYE), there is only a slight increase in the chrF3 score by 1.5 points. In the NLLB paper it is mentioned that an increase of 1 point in chrF3 score is always detectable by human evaluators [133]. However, there is no improvement in terms of BLEU score. This is once again a natural outcome caused by the difference in the language of the hypothesis and the reference.

Introducing monolingual data with synthetic English sides does show some improvements in EN $\rightarrow$ HYW for both metrics. Focusing on NLLB + DA-HYE$_{\text{hyw-mono}}$, i.e. the model trained with the synthetic dataset whose Armenian side originates from the monolingual Western Armenian dataset and the English sides are generated by the NLLB + HYE model, which is trained with Eastern Armenian data; there is an increase of 9.2 points in chrF3 and 5.6 points in BLEU score. By introducing Western Armenian sentences, the models now generate proper Western Armenian outputs instead of Eastern Armenian ones hence the increase. When compared with the oracle, NLLB + DA-HYE$_{\text{hyw-mono}}$ lacks almost 10 points both in chrF3 and BLEU metrics. The qualitative investigation has shown that this model struggles to translate inputs that are outside of its training domain, specifically it fails to translate words from the religious domain since its training data, the synthetic Western Armenian monolingual dataset, does not particularly include the religious domain. We conduct a qualitative comparison between the outputs of NLLB + DA-HYE$_{\text{hyw-mono}}$ and NLLB + DA-HYE$_{\text{hyw-par}}$ to see where the models excel and fail. It is important to note that in these examples the English references are procedurally generated by the NLLB + HYE model to have a synthetic parallel corpus and therefore should be considered as genuine sentences.

In Figure 6.2 we observe fairly similar hypotheses generated by the models with only a few disparities. First, for the translation of the word "alone", NLLB + DA-HYE$_{\text{hyw-mono}}$ used the incorrect word, "ինքզինքս", which means "myself"; whereas the model NLLB + DA-HYE$_{\text{hyw-par}}$ uses a different word Մինակ but this is a synonym of Առանձին meaning alone. This kind of erroneous word choices happens by observation more on NLLB + DA-HYE$_{\text{hyw-mono}}$ especially when it encounters a word outside of its training domain. Second, there is a discrepancy in the mood of the verb "would understand" in the hypotheses. Note that the verb in the Western Armenian reference is not written with any mood.

ENG Ref.: I felt alone and thought that no one would understand my feelings.
HYW Ref.: Առանձին կը զգայի եւ կը մտածէի, թէ ոչ մէկը պիտի հասկնայ զգացումներս:
Hyp. of NLLB + DA-HYE$_{\text{hyw-mono}}$: Ես ինքզինքս կը զգայի եւ կը մտածէի, որ ոչ ոք կը հասկնայ զգացումներս:
Hyp. of NLLB + DA-HYE$_{\text{hyw-par}}$: Մինակ կը զգայի եւ կը մտածէի, որ ոչ մէկը զգացումներս պիտի հասկնար:

Figure 6.2.: Qualitative Example 1 - Choice of Word and Time Agreement

However, the synthetic English reference contains a mood that should have simply been "will understand". This is likely caused by the difference in the meaning of the word պիտի in Western and Eastern Armenian. The word can be only used in conjunction with a verb and in Western Armenian it adds the future time information and in Eastern Armenian, it adds the necessitative mood. This shows that the translations for the word պիտի were generated according to Eastern Armenian and thus lead to an incorrect way of learning. We provide the following example from the AALW subset, with the original English sentence and the synthetic English sentence:

Original HYW Sentence: Ուրիշները այդպէս պիտի մտածե՞ն երբեք:
Original ENG Sentence: Will ever others think so?
ENG Sentence generated by NLLB + HYE: Should others think that way?

Figure 6.3.: Illustrative Example 2 - The translation of պիտի

As the example in Figure 6.3 shows the word պիտի is translated as "should", as it is considered as the necessitative mood. This is an expected behavior since NLLB + HYE is trained with Eastern Armenian data.

Returning to the hypotheses, the translation by NLLB + DA-HYE$_{\text{hyw-mono}}$ outputs կը հասկնայ which is in simple present tense with no mood and the translation by NLLB + DA-HYE$_{\text{hyw-par}}$ correctly translates "would understand" with պիտի հասկնար. Further qualitative investigation has shown that this type of error happens quite often in both models with other words that have different meanings in both languages and one of the reasons why there is a difference in scores with the oracle model.

We observe that there is a difference of 5.9 points in the chrF3 score and of 5.7 points in the BLEU score between the models trained with synthetic parallel data in favor of the model that uses the training set that corresponds to the test set in the evaluation. This is a natural outcome since the domains and sentence styles are more similar between the training and the test subset of the same dataset than the training set of another dataset. But what exactly does NLLB + DA-HYE$_{\text{hyw-par}}$ better than NLLB + DA-HYE$_{\text{hyw-mono}}$ that there is this amount of difference in scores? Qualitative example 2 showed the choices of verb moods and words, we continue an additional example for stylistic choice:

ENG Ref.: For example, we become "God's fellow workers" by preaching the good news of his Kingdom and making disciples. (1 Cor. 3:5-9)
HYW Ref: Օրինակ, "Աստուծոյ գործակից"ները կ'ըլլանք, երբ Թագաւորութեան բարի լուրը կը քարոզենք եւ կ'աշակերտենք (Ա. Կոր. 3:5-9):
Hyp. of DA-HYE$_{\text{hyw-mono}}$: Օրինակի համար, մենք կը դառնանք "Աստուծոյ գործակիցները"՝ քարոզելով անոր թագաւորութեան բարի լուրը եւ աշակերտներ դարձնելով (1 Կորն. 3.5-9):
Hyp. of DA-HYE$_{\text{hyw-par}}$: Օրինակ, "Աստուծոյ գործակից" կը դառնանք՝ իր Թագաւորութեան բարի լուրը քարոզելով եւ աշակերտելով (Ա. Կոր. 3:5-9):

Figure 6.4.: Qualitative Example 2 - Stylistic Choice

The example in Figure 6.4 shows how exposure to a certain subset affects the style of sentences. The chosen example is from the Watchtower subset whose writing style is very distinct with its phrasing of words. For example, the word "Kingdom" / Թագաւորութեան is capitalized both in English and Western Armenian reference, a stylistic aspect identified by the NLLB + DA-HYE$_{\text{hyw-par}}$ since it has been exposed to other examples from the Watchtower subset. Another example is the Bible reference style which is properly captured by the model that has been exposed to the Watchtower subset and the other model fails to do so (indicated in blue). Keeping this stylistic choice is naturally debatable for the general case, however, in this environment, it affects positively since the HYW-test contains examples from the Watchtower subset as well. NLLB + DA-HYE$_{\text{hyw-mono}}$'s training data contains only news articles and therefore lacks the style of specific datasets within Western Armenian-English parallel corpus such as the Bible, the Watchtower, Houshamadyan, and Wikipedia. We will continue on this lead in the evaluation on supervised testsets to see if the scores within these subsets have the same pattern and if there exists another pattern within the subsets of the news domain within the Western Armenian-English parallel corpus.

The exposure to the proper dataset has also an effect on the correct word choice, as shown in Figure 6.5:

Although both translations within Figure 6.5 are not fully satisfactory, the one generated by NLLB + DA-HYE$_{\text{hyw-par}}$ makes more sense. The choice for the word "blasphemer" by the NLLB + DA-HYE$_{\text{hyw-mono}}$ model was "Համբերողը" which means "the person who endures / patiently waits", making no sense. The word "blasphemer" does not appear within the synthetic parallel dataset that originates from the Western Armenian monolingual dataset and thus the peculiar word choice.

The final observation regards the proper backtransliteration of Armenian personal and geographical names. Although the synthetic data generation yielded the English transliterations of Armenian names according to the Eastern Armenian transliteration rules, having been exposed to the same names on the Armenian side allows better backtransliterations into Western Armenian. An example illustrating this is shown below:

ENG Ref.: Take the blasphemer outside the camp; and let all who were within hearing lay their hands on his head, and let the whole congregation stone him.
HYW Ref: Այն հայհոյող մարդը բանակէն դուրս հանէ եւ բոլոր լսողները ձեռքերնին անոր գլխուն վրայ թող դնեն ու բոլոր ժողովուրդը թող քարկոծեն զանիկա:
Hyp. of DA-HYE$_{\text{hyw-mono}}$: Համբերողը դուրս բերէք ճամբարէն, եւ բոլոր անոնք, որոնք ունկնդրութենէն ներս էին, թող ձեռքերը դնեն անոր գլուխին վրայ, եւ ամբողջ հաւաքականութիւնը թող քարերով քարկոծէ զայն:
Hyp. of DA-HYE$_{\text{hyw-par}}$: Ուսպի հայհոյողը ճամբայէն դուրս հանեցէք եւ լսելու մէջ եղողները թող իրենց ձեռքերը անոր գլխուն վրայ դնեն եւ ամբողջ ժողովը թող քարկոծէ զայն:

Figure 6.5.: Qualitative Example 3 - Word Choice

ENG Ref.: A view from Chmshgadzak (Source: Kasbarian family collection. Courtesy of Vazken Andréassian, Paris)
HYW Ref: Չմշկածագէն տեսարան մը (Աղբիւր՝ Գասպարեան ընտանիքի հաւաքածոյ. շնորհակալութիւններ՝ Վազգէն Անդրէասեանին)
Hyp. of DA-HYE$_{\text{hyw-mono}}$: Նկարում մը Չմշկածակէն (Աղբիւր. Քասբարեան ընտանեկան հաւաքածոն. Վազգէն Անդրէասեանի կողմէ, Փարիզ)
Hyp. of DA-HYE$_{\text{hyw-par}}$: Չմշկածագէն տեսարան մը (Աղբիւր՝ Քասբարեան ընտանիքի հաւաքածոյ. գեղանկար՝ Վազգէն Անդրէասեանի, Փարիզ)

Figure 6.6.: Qualitative Example 4 - (Back)transliteration of Armenian geographical names

The example in Figure 6.6 indicates that the word Չմշկածագ is contained in many different examples, the qualitative investigation into the synthetic dataset confirms this, and the word is transliterated into different versions:

- Ան ծնած էր Չմշկածագ 1884-ին եւ այստեղէն Ռուսիա անցնելով երկար ճամբայ կը կտրէ մինչեւ որ կը հասնի Միացեալ Նահանգներ:
  He was born in Chmshkag in 1884 and from here to Russia he would travel a long way until he arrived in the United States.

- Տերսիմ գաւառը [սանճաք] կ՚ընդգրկէր հետեւեալ գաւառակները [քազա]՝ Խոզաթ (կեդրոնական գաւառ), Չարսանճաք, Չմշկածագ, Քիզիքիլիսէ, Մազկերտ եւ Օվաճիք:

  The province of Tersim [Sanchak] included the following provinces [Kaza]: Khozat (central province), Charsanjak, Chmshkatag, Kizikilis, Mazkert, and Ovajik.

- Չմշկածագի թեմը արձանագրած է 26 հայ բնակիչով 5 հայ ծուխ 1902-ին:
  The Diocese of Tsmshkakanag was established in 1902 with 26 Armenian inhabitants and 5 Armenian parishes.

Nevertheless the model is able to map all the erroneous transliterations to the original Western Armenian word and could backtransliterate correctly, while the other model could not since it has not seen any examples containing this word.

Coming to HYW → EN direction, the baseline model already shows a decent performance, this is already known by the community, as Western Armenian texts are regularly translated in casual conversation with available Eastern Armenian machine translation services like Google Translate. This was the main inspiration for the method of using an Eastern Armenian model for Western Armenian data augmentation and it yielded positive results in the opposite direction. However, in this direction, using more Eastern Armenian data results only in a slight increase in both metrics (a maximum increase of 2.3 chrF3 points and 0.7 BLEU points). Upon qualitative inspection, this increase is identified to be caused by better transliteration of named entities. This is shown by means of the qualitative examples shown in Figure 6.7:

The examples within Figure 6.7 clearly show that the NLLB model is not specialized to handle Armenian names and it tries to match with the closest name possible from its knowledge. Notice the NLLB model actually understands the names that appear in multiple languages and outputs the English cognate, in the first example this corresponds to Gregory, and in the second one, Stephan, whereas all other models resort to the Eastern Armenian transliteration method. When NLLB stumbles upon an Armenian name that has no direct translation in another language it fails to transliterate, this happens with Araxi, Haiganoush, Dikranouhi, Nevshehirlian, and Kayseri although the last two are in Turkish. Relying on the Eastern Armenian transliterations yields a better result relative to the NLLB model, however the transliterations are not correct and almost all of them will have no improvement in terms of BLEU score and only partial improvement in terms of chrF3 score, hence the slight increases. Other than that, all models have approximately the same performance, indicating that exposure to Western Armenian examples does not make much of a difference. The given Western Armenian sentences though are considered as they are in Eastern Armenian and the English sides are made accordingly, therefore the translations have almost the same quality as the baseline since it also does the same. The score increases in this direction are so small that it cannot be considered as a proper improvement.

The results point out that a model trained solely on Eastern Armenian data cannot produce satisfactory Western Armenian outputs. However, when they are used for synthetic data generation with Western Armenian monolingual input, any model trained on this data yields tolerable outputs in EN → HYW direction. This is a quick and easy way to support Western Armenian machine translation without any genuine parallel corpus. This method however only has a negligible impact in the other direction with the only improvements on the transliteration of names.

HYW Ref.: Գրիգորի մայրը՝ Արաքսի Գալուստեան, ծնած է Օրդու, 1912-ին:
ENG Ref.: Krikor's mother, Araxi Kalousdian was born in Ordou in 1912.
Hyp. of NLLB: Gregory's mother, Galveston of Aragon, was born in Ortu in 1912.
Hyp. of NLLB + HYE: Grigor's mother, Araks Galustyan, was born in Ortu in 1912.
Hyp. of NLLB + DA-HYE$_{\text{hyw-mono}}$: Grigor's mother, Araks Galustyan, was born in Ortu in 1912.
Hyp. of NLLB + DA-HYE$_{\text{hyw-par}}$: Grigor's mother, Araks Galustyan, was born in Ortu, 1912.

HYW Ref.: Հայկանուշը կեսարացի Սթեփան եւ Տիգրանուհի Նեւշէհիրլեաններու դուստրն էր:
ENG Ref.: Haiganoush was the daughter of Stepan and Dikranouhi Nevshehirlian of Kayseri.
Hyp. of NLLB: Achanus was the daughter of Stephan the Caesarian and the Tigranite Nebuchadnezzar.
Hyp. of NLLB + HYE: Haykanush was the daughter of Stepan the Caesar and Tigranuhi Neveshehirlians.
Hyp. of NLLB + DA-HYE$_{\text{hyw-mono}}$: Haykanush was the daughter of Stepan of Caesarea and Tigranouhi Neveshehirlian.
Hyp. of NLLB + DA-HYE$_{\text{hyw-par}}$: Haykanush was the daughter of Stepan the Caesarean and Tigranouhi Nevshehirlian.

Figure 6.7.: Qualitative Examples 5-6 - (Back)transliteration of Armenian personal and geographical names

6.1.2. Evaluation on Supervised Testsets

Table 6.2 shows individual results on each dataset whose trainset is either used for creating the synthetic parallel trainset (in the case of NLLB + DA-HYE$_{\text{hyw-par}}$) or is seen during training (in the case of NLLB + HYW).

In EN → HYW direction a similar pattern emerges as in the previous table of the combined results, with a few exceptions. We see subsets like Hayern Aysor where NLLB + DA-HYE$_{\text{hyw-mono}}$ performs slightly better than NLLB + DA-HYE$_{\text{hyw-par}}$ with a 0.7 increase in chrF3 score and a tie in terms of BLEU score, whereas in the general picture, NLLB + DA-HYE$_{\text{hyw-par}}$ performs better. This is caused by the heavy composition of the news domain within the monolingual Western Armenian dataset, as the Hayern Aysor subset is also from this domain. Another interesting point is that the difference in scores between NLLB + DA-HYE$_{\text{hyw-mono}}$ and NLLB + DA-HYE$_{\text{hyw-par}}$ is only apparent in the Bible, the Watchtower and Houshamadyan subsets. This is evidence to our claims that the stylistic

Evaluated on:	HYW-test (Supervised)															
Subset	AALW				Bible				Hamazkayin				Hayernaysor			
Direction	EN → HYW		HYW → EN		EN → HYW		HYW → EN		EN → HYW		HYW → EN		EN → HYW		HYW → EN	
Model \ Score	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU
NLLB	*29.9*	*0.7*	*39.7*	*13.3*	*34.4*	*2.6*	*50.2*	*23.7*	*33*	*1.1*	*42.9*	*11.5*	*44.3*	*3.4*	*58.1*	*28.6*
+ HYE	30.5	1.2	42.4	15.4	34.2	1.8	47.7	20.1	36.6	1.9	**48.7**	14.5	45.9	4	62.9	33.2
+ DA-HYE$_{\text{hyw-mono}}$	38.9	4.7	**43.2**	**15.7**	36	3.6	44.9	18	47.7	7.8	47.6	**15.1**	**58.4**	**14.9**	**64.3**	**34.9**
+ DA-HYE$_{\text{hyw-par}}$	**42.7**	**6.8**	42.9	**15.7**	**53.4**	**16.5**	**48**	**21**	**49.4**	**8.9**	47.6	14.4	57.7	**14.9**	62.3	32.3
+ HYW (Oracle)	*44.2*	*9.4*	*49.3*	*22.5*	*58.7*	*22.3*	*61.5*	*37.7*	*52.2*	*11.8*	*53.2*	*18.9*	*58.5*	*16.3*	*65*	*36.5*

Evaluated on:	HYW-test (Supervised)											
Subset	Houshamadyan				Watchtower				Wikipedia			
Direction	EN → HYW		HYW → EN		EN → HYW		HYW → EN		EN → HYW		HYW → EN	
Model \ Score	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU
NLLB	*32.4*	*2.6*	*41.9*	*13.6*	*30.1*	*1.7*	*51.8*	*28.2*	*33.6*	*1.9*	*45.1*	*16.3*
+ HYE	34.9	2.8	47.5	16.7	31	0.8	47.4	21.4	33.8	1.7	**45.8**	**15.9**
+ DA-HYE$_{\text{hyw-mono}}$	45.6	7.4	**47.7**	**17.2**	41.7	7	**47.6**	**22.5**	40.7	4.7	45.1	15.2
+ DA-HYE$_{\text{hyw-par}}$	**51.9**	**13.8**	**47.7**	**17.2**	**57.7**	**23.8**	46.4	20.5	**42.1**	**5.5**	44.9	15.5
+ HYW (Oracle)	*58.8*	*22.3*	*61.2*	*31*	*60.9*	*29.2*	*61.6*	*39.5*	*42*	*6.2*	*46.6*	*17.5*

Table 6.2.: Evaluation scores of the models within the scenario "No parallel Western Armenian data" on each Western Armenian supervised test subsets.

exposure is appreciated by these datasets except Wikipedia. The Watchtower enjoys the inclusion of its training data at most, indicating it has a unique style that is captured by the Western Armenian-English parallel corpus. Another argument may be the sizes of these subsets. The Bible, the Watchtower, and Houshamadyan subsets are the largest three subsets within the Western Armenian parallel corpus, therefore including their training data is most beneficial to these subsets.

Coming to the opposite translation direction, we also see the same pattern of slight performance increases. There are two outliers of this pattern: first the Bible the winner is NLLB + DA-HYE$_{\text{hyw-par}}$ instead of NLLB + DA-HYE$_{\text{hyw-mono}}$ with a 3-point difference in BLEU and 3.1 point difference in chrF3. This is probably caused by the domain mismatch of NLLB + DA-HYE$_{\text{hyw-mono}}$ however we do not see this result in the Watchtower subset, where its domain composition is heavily built by the religious domain, although it is more flexible than the Bible subset as it also includes more domains than religion. The second outlier is the Wikipedia subset where NLLB + HYE is the winner in both BLEU and chrF3 scores. This is an interesting result since it indicates that including Western Armenian content including its Wikipedia subset is less beneficial than its Eastern counterpart. We want to investigate this in the in-depth analysis experiment. Another interesting point is that in the outlier subsets and the Watchtower subset, the models perform worse than the baseline model. This indicates that the baseline model has already seen examples from the Bible other religious content and Wikipedia.

6.1.3. Evaluation on Unsupervised Testsets

As seen on Table 6.3, there is a drop in performance in both unseen datasets for both directions. This outcome is expected, but the performance drop in especially EN → HYW direction shows that a substantial amount of additional parallel Western Armenian data is needed to bring the general performance to an acceptable level. Translating in EN → HYW

Evaluated on:	HYW-test (Unsupervised)							
Subset	Gulbenkian				VoC			
Direction	EN → HYW		HYW → EN		EN → HYW		HYW → EN	
Score / Model	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU
NLLB	*35.6*	*1.2*	*45.9*	*14.4*	*25.8*	*0.2*	*28.9*	*3.9*
+ HYE	36.8	1.3	**48.4**	**16.7**	26.9	0.2	**35**	4.7
+ DA-HYE$_{\text{hyw-mono}}$	**46.2**	**6**	48.2	16.1	31.7	1.3	34.5	**5.6**
+ DA-HYE$_{\text{hyw-par}}$	45.8	5.8	47.6	15.4	**32.8**	**1.7**	34.5	5.3
+ HYW (Oracle)	*46.6*	*6.4*	*49.7*	*17.2*	*32.3*	*0.9*	*35.2*	*6.1*

Table 6.3.: Evaluation scores of the models within the scenario "No parallel Western Armenian data" on the Western Armenian each Western Armenian unsupervised test subsets.

direction NLLB + DA-HYE$_{\text{hyw-mono}}$ performs best in the Gulbenkian subset, although the difference with the second best NLLB + DA-HYE$_{\text{hyw-par}}$ is too near that we cannot declare a decisive winner. The included news domain has increased the overall performance at almost the same rate. In the VoC dataset, we see a similar picture, where both models that use synthetic data perform better than the model that is trained upon genuine parallel data, which shows the slight impact of Western Armenian examples, however, the scores across every model are so low in this section, that it is not possible to make a meaningful deduction.

Coming to HYW → EN direction, we see a diverging result from the previous tables where NLLB + HYE is the winner. This is possibly caused by the genuine example pairs faring better than procedurally generated ones. In VoC, the classic result reigns where synthetic Western Armenian parallel datasets perform better than Eastern Armenian genuine parallel data, however the chrF3 and BLEU scores are too low that even the best model struggles to translate texts from this dataset. This is expected as this dataset has artistically complex and longer sentences.

6.2. Scenario 2: Parallel Western Armenian Data

6.2.1. Evaluation on the Combined Testset

Table 6.4 provides an overview of the performance of the models when they are trained with genuine Western Armenian parallel data, and as expected it brought an improvement to the translation quality in both directions, when compared with the models that relied on solely Eastern Armenian parallel data. The improvement is larger in EN → HYW altogether, since the baseline and Eastern Armenian-trained models could already translate from Western Armenian to English with a certain level of quality as mentioned in the previous scenario.

For EN → HYW direction the model NLLB + HYW and the double finetuned model NLLB + HYE + HYW have the best performances in terms of BLEU score; and in terms of chrF3

Evaluated on:	HYW-test Combined			
Direction	**EN → HYW**		**HYW → EN**	
Model \ **Score**	**chrF3**	**BLEU**	**chrF3**	**BLEU**
NLLB	*34.9*	*2.2*	*47.8*	*20*
+ HYE	*36.4*	*2.2*	*50.1*	*20.3*
+ HYW	54	17	57.2	29.4
+ DA-HYW$_{\text{hyw-mono}}$	47.8	9	54.6	25.5
+ {HYW, DA-HYW$_{\text{hyw-mono}}$}	**54.2**	16.6	**57.7**	**29.8**
+ {HYE, HYW}	52.3	15.3	57.5	29.5
+ HYE + HYW (Double Finetune)	**54.2**	**17.1**	57.4	29.3

Table 6.4.: Evaluation scores of the models within the scenario "Parallel Western Armenian data" on the Western Armenian combined testset.

score the doubly finetuned model share the throne with data augmented NLLB + {HYW, DA-HYW$_{\text{hyw-mono}}$} model. This shows Western Armenian-English genuine parallel data (HYW) has specific information/style that is necessary for HYW-test and when it is not included as in the case of NLLB + DA-HYW$_{\text{hyw-mono}}$ the scores become significantly lower than the cases where it is included. This is possibly due to the lack of domain coverage of the Western Armenian monolingual dataset as it is solely comprised of news articles. Note that the English sides of the monolingual Western Armenian data are generated by the NLLB + HYW model in this scenario. Additionally, any other data inclusion to the genuine parallel dataset such as synthetic parallel dataset as in the case of NLLB + {HYW, DA-HYW$_{\text{hyw-mono}}$} or Eastern Armenian parallel data as in joint finetuning (NLLB + {HYE, HYW}); results in slight drops in performance, this indicates that the inclusion of extra data is not beneficial for the HYW-test set as it leads to confusions due to conflicting examples through the different languages (e.g. in the case of joint finetuning) and the different styles (e.g. in the case of NLLB + {HYE, HYW}). The sole inclusion of Western Armenian parallel corpus in a training session seems to yield the best results in EN → HYW direction. The extra knowledge gained from the Eastern Armenian data seems to have a negligible effect, presumably, the final learned parametrization of the distribution is similar due to a local extremum, indicating the learned Eastern Armenian knowledge is either forgotten or the model overfits to the Western Armenian examples. We will continue to compare the NLLB + HYW and NLLB + HYE + HYW models in this regard and see if that holds in each supervised subset. We continue with qualitative investigation to clearly identify what the models do right or wrong.

The example in Figure 6.8 shows how exposure to a certain dataset and its domain can change the correct word choice. Palu is a city in the Ottoman Empire and today's Turkey. The correct Armenian pronunciation Բալու is captured by the models who have been exposed to the genuine Western Armenian-English parallel dataset and the

ENG Ref.: The Palu villagers have two ways of winnowing.
HYW Ref.: Բալուի գիւղացիները հոսելու երկու ձեւեր ունին:
Hyp. of NLLB + HYW: Բալուի գիւղացիները հոսելու երկու ձեւ ունին:
Hyp. of NLLB + DA-HYW$_{\text{hyw-mono}}$: Փալու գիւղացիները երկու երկիրներով կ'ունենան աշխապանք:
Hyp. of NLLB + {HYW, DA-HYW$_{\text{hyw-mono}}$}: Բալուի գիւղացիները հանքագործութեան երկու ձեւ ունին:
Hyp. of NLLB + {HYE, HYW}: Բալուի գիւղացիները երկու ձեւ ունին հոսելու:
Hyp. of Double FT: Բալուի գիւղացիները հոսելու երկու ձեւեր ունին:

Figure 6.8.: Qualitative Example 7 - Choice of Backtransliteration and Technical Word Translation

model that has not spells it incorrectly. Another incorrect translation by the NLLB + DA-HYW$_{\text{hyw-mono}}$ occurs on the word "winnowing" which originates from the agricultural domain of Houshamadyan. Since this model does not have access to this subset it generated a nonsensical translation of this word. An interesting point is that when the synthetic and genuine parallel data is introduced (NLLB + {HYW, DA-HYW$_{\text{hyw-mono}}$) the translation results in հանքագործութիւն, which means "mining", an incorrect translation but relative to NLLB + DA-HYW$_{\text{hyw-mono}}$ a translation headed towards the correct direction. This shows the inclusion amount of the synthetic dataset must be optimized in order to boost the translation performance and not lead to additional confusions. The NLLB + {HYW, DA-HYW$_{\text{hyw-mono}}$ model's training data is composed of 50% genuine and 50% parallel data, tipping the balance towards genuine examples could have a positive impact.

Coming to HYW $\rightarrow$ EN direction, we encounter a similar picture where all the models except NLLB +DA-HYW$_{\text{hyw-mono}}$ perform similarly. The drop in score for this model could be once again traced back to the inadequate domain coverage. The differences in scores are tighter in this direction, indicating the different training schemes do not have a particular effect and the additional data aside from the Western Armenian-English parallel corpus does not seem to confuse the models' translation knowledge in this direction.

To summarize, the results from the combined evaluation of the whole Western Armenian testset show that it favors the inclusion of its training set very strongly, indicating this dataset has a great number of unique styles and vocabulary that is not covered by the monolingual dataset. The inclusion of additional data directly into the same training session, i.e. enlarging the training data of models seems to slightly worsen the performances in relation to only including Western Armenian-English parallel corpus in EN $\rightarrow$ HYW direction. However, this is not the case in the opposite direction as the models with additional data within a single finetuning session as the models NLLB + HYW and NLLB + HYE + HYW take up the lower ranks. Since the models perform similarly in HYW $\rightarrow$ EN direction, the most confident argument we can make in this case is that the additional data does not worsen the translation quality level achieved by NLLB + HYW.

6.2.2. Evaluation on Supervised Testsets

Evaluated on:	HYW-test (Supervised)															
Subset	AALW				Bible				Hamazkayin				Hayernaysor			
Direction	EN → HYW		HYW → EN		EN → HYW		HYW → EN		EN → HYW		HYW → EN		EN → HYW		HYW → EN	
Score / Model	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU
NLLB	*29.9*	*0.7*	*39.7*	*13.3*	*34.4*	*2.6*	*50.2*	*23.7*	*33*	*1.1*	*42.9*	*11.5*	*44.3*	*3.4*	*58.1*	*28.6*
+ HYE	*30.5*	*1.2*	*42.4*	*15.4*	*34.2*	*1.8*	*47.7*	*20.1*	*36.6*	*1.9*	*48.7*	*14.5*	*45.9*	*4*	*62.9*	*33.2*
+ HYW	**44.2**	**9.4**	49.3	22.5	**58.7**	**22.3**	**61.5**	**37.7**	52.2	11.8	53.2	18.9	58.5	16.3	65	36.5
+ DA-HYW$_{\text{hyw-mono}}$	41.8	5.7	48.3	21.8	39	4.4	50.1	24.1	51	10.3	53.1	18.8	58.3	14.7	66.1	37.6
+ {HYW, DA-HYW$_{\text{hyw-mono}}$}	43.3	8.7	48	21.3	57.3	21.2	60.6	36.6	**52.8**	**12.3**	**53.7**	**19.3**	59.1	15.5	**66.6**	**38**
+ {HYE, HYW}	41.8	7	48.8	21.8	56.8	20.3	60.6	37	50.1	10.4	53.2	18.8	55.5	13.3	66.1	37.1
+ HYE + HYW (Double Finetune)	43.8	8.5	**49.4**	**22.7**	58.5	22.2	61	36.9	52.3	12.1	53.4	19	**59.5**	**17.1**	65.8	36.9

Evaluated on:	HYW-test (Supervised)											
Subset	Houshamadyan				Watchtower				Wikipedia			
Direction	EN → HYW		HYW → EN		EN → HYW		HYW → EN		EN → HYW		HYW → EN	
Score / Model	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU	chrF3	BLEU
NLLB	*32.4*	*2.6*	*41.9*	*13.6*	*30.1*	*1.7*	*51.8*	*28.2*	*33.6*	*1.9*	*45.1*	*16.3*
+ HYE	*34.9*	*2.8*	*47.5*	*16.7*	*31*	*0.8*	*47.4*	*21.4*	*33.8*	*1.7*	*45.8*	*15.9*
+ HYW	**58.8**	**22.3**	61.2	31	**60.9**	**29.2**	**61.6**	**39.5**	42	6.2	46.6	17.5
+ DA-HYW$_{\text{hyw-mono}}$	49.3	9.7	56.7	24.5	44.8	8.8	56.2	31.7	41.3	4.9	46.4	16.8
+ {HYW, DA-HYW$_{\text{hyw-mono}}$}	57.2	19.9	60.6	30.1	60	28.3	60.9	38.4	**42.7**	6.2	46.9	17.6
+ {HYE, HYW}	55.8	18.7	60	29.6	59.6	27.8	60.8	38.3	41.2	5.8	**47.3**	**18.1**
+ HYE + HYW (Double Finetune)	58.6	22	**61.5**	**31.6**	60.7	28.9	61.3	38.7	42.1	**6.3**	46.8	17.5

Table 6.5.: Evaluation scores of the models within the scenario "Parallel Western Armenian data" on the Western Armenian supervised test subsets.

The results in Table 6.5 show some level of similarity with the general overview, but more importantly they illustrate each subset's adaptability for the introduced training data.

In EN → HYW direction we see larger differences in scores between the NLLB + HYW and NLLB + HYE + HYW. This suggests that there is some difference in the parameters of the said models and the subsets have favored one or the other. We see larger differences in the AALW and Hayernaysor subsets. AALW does not enjoy the inclusion of Eastern Armenian data with the extra finetuning session, probably because of its very distinct Western Armenian style. The book was written in 1913, during which Western Armenian was the dominant variant and Eastern Armenian influence was non-existent. Nowadays the intermixture of languages is more welcome since the Eastern variant has an updated vocabulary of technological words. Hayern Aysor conversely favors the doubly finetuned model showing the subsequent Eastern and Western Armenian finetuning is welcomed. This is possibly caused by the hybrid nature of this dataset including personal names with new orthography and the remaining Western Armenian in classical orthography, which is handled better by this model. Interestingly the joint finetuning did not work in Hayern Aysor which seems to introduce more confusion than adaptability.

The subsets of the Bible, Houshamadyan, and Watchtower seem to favor the NLLB + HYW and NLLB + HYE + HYW models, i.e. the models where the genuine Western Armenian-English dataset is solely included in a single finetuning session. These are the larger subsets within the Western Armenian-English corpus with specific domains. Any additional direct data caused in less scores for these datasets.

The scores of the opposite direction bring up a more complex picture. Here, the religious subsets favor the NLLB + HYW model as well with a difference of at least 0.6 BLEU score

to the second-best model. The doubly finetuned model is the second most successful model in HYW → EN direction with the AALW and Houshamadyan subsets. The qualitative investigation of the outputs of the doubly finetuned model regarding the Houshamadyan subset and the comparison with the outputs of NLLB + {HYW, DA-HYW$_{\text{hyw-mono}}$} have illuminated that the former model does a better job recognizing named entities and deliberately not translating these (e.g. Կարմիր Վանք is translated as "Garmir Vank" and not as "Red Monastery") semantically. As the Houshamadyan dataset is about the history and geography of Ottoman Armenian communities, there is a fair share of named entities in it. Lastly, joint finetuning has the best performance translating HYW → EN within the Wikipedia dataset. This is logically sound, as the articles within this dataset are mostly Eastern Armenian translations and contain both orthographies, especially for names. It is previously shown that while translating into English, seeing both orthographies directly from both Armenian variants is more favorable than double finetuning.

In summary, the results in EN → HYW direction show that subsets are more susceptible to domain mismatch and generally yield worse scores if the portion of their relevant data gets smaller with the larger training sets. In HYW → EN direction this pattern is not seen as powerful as in the other direction, here the inclusion of more data generally yields positive outcomes. The results follow the pattern of the combined testset, however, the gaps between the scores become slightly larger in the individual subset, showing that the gained knowledge of the models is distinguishable and is appreciated differently by each subset.

6.2.3. Evaluation on Unsupervised Testsets

Evaluated on:	**HYW-test (Unsupervised)**							
Subset	**Gulbenkian**				**VoC**			
Direction	**EN → HYW**		**HYW → EN**		**EN → HYW**		**HYW → EN**	
Score / Model	**chrF3**	**BLEU**	**chrF3**	**BLEU**	**chrF3**	**BLEU**	**chrF3**	**BLEU**
NLLB	*35.6*	*1.2*	*45.9*	*14.4*	*25.8*	*0.2*	*28.9*	*3.9*
+ HYE	*36.8*	*1.3*	*48.4*	*16.7*	*26.9*	*0.2*	*35*	*4.7*
+ HYW	46.6	6.4	49.7	17.2	32.3	0.9	35.2	6.1
+ DA-HYW$_{\text{hyw-mono}}$	47.8	6.7	**50.1**	**18.1**	32.7	1.5	37.5	**7.3**
+ {HYW, DA-HYW$_{\text{hyw-mono}}$}	**48.5**	**7.2**	**50.1**	17.8	**33.4**	**1.7**	37.7	**7.3**
+ {HYE, HYW}	43.4	5.1	50	17.5	31.4	1.3	**37.8**	**7.3**
+ HYE + HYW (Double FT)	47	6.1	49.5	17	32.9	1.3	36.2	5.6

Table 6.6.: Evaluation scores of the models within the scenario "Parallel Western Armenian data" on the Western Armenian unsupervised test subsets.

The lower gains on the unsupervised datasets relative to the baseline, as seen in Table 6.6, shows the inadequacy of coverage by the collected data, which is the typical case for low-resource languages. Nevertheless, the results here show a different picture from the previous sections. For EN → HYW direction, combining monolingual and parallel Western Armenian datasets has resulted in improvements, a desirable outcome indicating that both datasets include valuable information about Western Armenian. Regarding HYW → EN,

the Gulbenkian dataset favors the monolingual data more, possibly from better domain overlap. Investigating VoC, three models have almost the same performance, but the scores are significantly worse than any other subset indicating the models cannot capture the literary/artistic style very well.

6.3. In-Depth Analysis 1: Impact of Domain vs. Language

The outcomes of the previous two scenarios have indicated that information with matching domains can have a greater impact. In this experiment, we focus on two subsets within the Western Armenian-English parallel dataset: Bible and Wikipedia. Both of these subsets also exist in the Eastern Armenian-English parallel dataset. This creates an opportunity to investigate whether the knowledge gained from the same language data or the same domain is more important for the specific subsets. The experiment could give more information about the nature of the subsets and give insights into data collection or building domain-specific Armenian translation models in the future. We trained several models with different configurations shown in Table 5.1 to have a clearer comparison. These models are ranked based on their performance in both directions. We also investigate if the same configuration has the same impact on both subsets.

6.3.1. Evaluated on Western Armenian Bible

Evaluated on:	**HYW-Bible-test**					
Direction	**EN → HYW**			**HYW → EN**		
Model \ **Score**	**chrF3**	**BLEU**	**Rank**	**chrF3**	**BLEU**	**Rank**
NLLB	*34.4*	*2.6*	-	*50.2*	*23.7*	-
+ HYW-Bible	58.4	22	①	61	36.9	①
+ HYE-Bible	32.6	1.6	③	40.5	12.9	⑥
+ HYW-Wiki	28.3	1	⑥	28	5.6	⑦
+ HYE-Wiki	26.5	0.4	⑦	39.9	14.4	⑤
+ {HYE-Bible, HYW-Wiki}	32.9	1.5	⑤	43.9	17.3	④
+ {HYE, HYW-Wiki}	33.2	1.5	④	48.2	21.7	②
+ HYE	34.3	1.8	②	47.7	20.1	③

Table 6.7.: Evaluation scores of the models within the experiment of "Impact of Domain vs. Language" on the Western Armenian Bible testset.

The results from Scenario 2 have shown that the Bible has a specific domain that does not enjoy the inclusion of additional training data. In Table 6.7, we see a similar picture as NLLB + HYW-Bible has the best performance in both directions by a large margin. In EN → HYW direction, all other models perform very poorly, indicating as previously that this direction is more sensitive towards both language and domain. Although both

scores are too low, the Bible subset chooses matching domain-mismatching language rather than matching language-mismatching domain. This could have two reasons: 1) Western Armenian Wikipedia's domain is too far from Western Armenian Bible. This is especially apparent while comparing the performances of NLLB + HYE-Bible and NLLB + {HYE-Bible, HYW-Wiki}, in which the inclusion of Western Armenian Wikipedia subset has a neutral to very slight negative impact; 2) Eastern Armenian Bible is written with the classical orthography which is the orthography of Western Armenian. Another interesting point is that all models except NLLB + HYW-Bible perform worse than the baseline in both directions, which shows that the baseline model was already trained with multilingual Bible data. This is more apparent in the opposite direction.

The scene in HYW → EN direction is somewhat different, as the models in 2-4th places are ranked according to the size of their training data. Interestingly, NLLB + HYE-Wiki performs better than NLLB + HYE-Bible with regard to BLEU scores, possibly data from Eastern Armenian Wikipedia provides new examples, whereas data from Eastern Armenian Bible does not and instead makes the model forget the baseline multilingual Bible knowledge.

In summary, the Bible dataset favors information regarding the matching domain more than the matching language. This is apparent in both directions. In EN → HYW direction the match of both domain and language is more important than the opposite direction as any mismatch results in huge drops in performance, whereas in the opposite direction, the mismatching training data still makes the model able to translate with some level of quality.

6.3.2. Evaluated on Western Armenian Wikipedia

Evaluated on:	**HYW-Wiki-test**					
Direction	**EN → HYW**			**HYW → EN**		
Score / **Model**	**chrF3**	**BLEU**	**Rank**	**chrF3**	**BLEU**	**Rank**
NLLB	*33.6*	*1.9*	-	*45.1*	*16.3*	-
+ HYW-Wiki	37.1	4.5	①	39.5	12.8	⑤
+ HYE-Wiki	34.4	1.7	②	43.2	14.4	④
+ HYW-Bible	22.4	0.3	⑥	30.5	5.3	⑥
+ HYE-Bible	20	0.3	⑦	22	1.9	⑦
+ {HYE-Wiki, HYW-Bible}	34.3	1.7	③	44.3	15.3	③
+ {HYE, HYW-Bible}	34.2	1.6	⑤	45.3	15.7	②
+ HYE	33.8	1.7	④	45.8	15.9	①

Table 6.8.: Evaluation scores of the models within the experiment of "Impact of Domain vs. Language" on the Western Armenian Wikipedia testset.

The evaluation on the Western Armenian Wikipedia dataset, as seen in Table 6.8, shows similarities with the results on Western Armenian Bible in EN → HYW direction, but in

the opposite direction, the results seem to diverge. Previous results have shown that EN → HYW direction is more susceptible to data of the same domain. This is also the case here, as the performances are ranked by domain first then by language.

In the opposite direction, however, there is a completely new ranking in which most interestingly the model with matching domain and matching language has the fifth position in the ranking. This is possibly caused by the non-direct translations within the examples of Western Armenian Wikipedia. Learning from Eastern Armenian Wikipedia seems to have worked better for translating from Western Armenian, as the size of this dataset is approximately two times larger than its Western Armenian counterpart; and in parallel to previous results, this direction benefits from more Eastern or Western Armenian examples. The models in the first five ranks are sorted by their training data size. It is also important to mention that all models perform worse than the baseline, indicating that it was already trained on multilingual Wikipedia examples. Another interesting finding is that information from the Bible does not help translate Wikipedia texts as much as information from Wikipedia for biblical texts.

The results from both subsets show a couple of important points: 1) To increase the quality of EN → HYW translations giving more data in the same domain is more important than in the same language, however the results from the scenarios have also shown that not including any Western Armenian example ultimately results in Eastern Armenian outputs and therefore low scores. 2) For HYW → EN direction, giving additional data with less regard to language or domain is generally beneficial, however, this is most likely caused by the baseline model's existing knowledge about both the Bible and Wikipedia.

6.4. In-Depth Analysis 2: Impact of Segmentation in Knowledge Transfer

Evaluated on:	**HYE-test**				**HYW-test (zero-shot)**				**HYW-test (double finetune)**			
Direction	**EN → HYE**		**HYE → EN**		**EN → HYW**		**HYW → EN**		**EN → HYW**		**HYW → EN**	
Score / Threshold	**chrF3**	**BLEU**	**chrF3**	**BLEU**	**chrF3**	**BLEU**	**chrF3**	**BLEU**	**chrF3**	**BLEU**	**chrF3**	**BLEU**
0	**48.9**	**16.3**	**53.9**	**28**	**33.9**	**1.9**	**47.7**	**19.3**	**51.4**	**15.5**	55.1	27.9
50	**48.9**	16.2	53.8	27.8	33.7	1.8	47.4	19.1	51.3	**15.5**	55.1	28
1000	48.7	15.9	53.7	27.7	33.6	1.8	47.2	18.8	51.3	15.4	**55.3**	**28.1**
50000	44.9	12.6	52.2	26	32.9	1.6	44.7	16.4	49.3	13.5	54.3	27
Double Finetune	-	-	-	-	-	-	-	-	*54.2*	*17.1*	*57.4*	*29.3*

Table 6.9.: Evaluation scores of the models within the experiment of "Impact of Segmentation in Knowledge Transfer" on the Eastern and Western Armenian combined testsets. The scores under "HYE-test" and "HYW-test (zero-shot)" belong to the models that are trained on Eastern Armenian-English parallel data only, whereas under double finetune belong to the models that are trained on Eastern Armenian-English then on Western Armenian-English data.

In this experiment, we encoded our Eastern and Western Armenian parallel datasets in such a way that their tokens have the maximum overlap, in other words, the amount of tokens that are missing from one dataset is minimized. This is achieved by replacing

rarely occurring tokens with more common ones, if possible. With this method the type overlap between the HYW-test and HYE-train tokens is increased from 99.37% to 99.7% which represents the zero-shot case and between the HYW-test and the union of HYW and HYE-train tokens is increased from 99.77% to 99.85% which represents the double finetune case.

The scores in Table 6.9 follow a constant decline across all subsets in both cases when the threshold is increased. This is possibly due to longer sequences from the finer segmentation being harder for the models to learn from. Additionally, custom encoding effectively makes the model unable to fully utilize the gained knowledge before the finetuning sessions as during finetuning the knowledge of the baseline NLLB model must be reshaped according to the new distribution of tokens. Table 5.3 shows that the overlap between Eastern and Western Armenian tokens is already quite large and the results show that increasing the overlap brings tinier gains than the introduced complexity caused by finer segmentations. Longer sequences have more dependencies and therefore a more complex distribution to parametrize. With increasing threshold values the drop in score is larger in EN $\rightarrow$ HYE/W direction, possibly due to the smaller knowledge about Armenian variants than about English. Another takeaway is that the vocabulary provided by Team NLLB [133] is good enough for knowledge transfer between the Armenian variants.

7. Conclusion

The deterioration and extinction of a language is a tragic sight, especially if one speaks that language themselves. Bringing and building modern tools for these languages could slow down and even stop this process. The greatest obstacle is to find tiny streams of data in the vast landscape brought by the age of information. A tough, but fulfilling challenge.

Our contribution to the community can be brought under two main categories: 1) The first neural network based machine translation model for Western Armenian. Machine translation models have perhaps the greatest impact among other NLP tools since they can be directly used by anybody. In the future, after polishing the models we want to publicly share the models and host the service via a website. 2) A parallel corpus of Western Armenian-English with more than 100,000 examples and an additional monolingual Western Armenian corpus with more than 1 million examples. We want to share all the collected data which is not under any copyright. By providing the first annotated data in Western Armenian we look forward to the inclusion of the language in further NLP research.

The main goal of this work was **to build a Western Armenian English translation model and achieve a certain level of translation quality**. Although automatic metrics generally do a worse job than human assessments, there is a correlation between the two as explained in subsubsection 2.2.2.6. The best model in EN $\rightarrow$ HYW direction achieves a BLEU score of 17.1, which is an acceptable performance within a low-resource language setting. In the opposite direction, the highest BLEU score reached is 29.8, which edges the state-of-the-art 30-40 BLEU score range. It is important to mention that these scores reflect the performance on the mixture of supervised and unsupervised testsets, with a tipped balance towards supervised testsets. Scores from the unsupervised testsets have proven that the parallel corpus does not yet have adequate domain coverage in order to make high-quality translations.

We have found out from the first scenario that using monolingual Western Armenian data and a pretrained machine translation model that supports Eastern Armenian to generate a synthetic parallel corpus and using this as training data for Western Armenian-English translation task results in a translation performance level that is close to the models that are trained with genuine Western Armenian-English parallel data, indicating this is a quick and cheap solution where the requirements for translation quality is not so high. Scenario 2 illustrated that some subsets do not welcome additional data and usually result in confusion and inferior results. Additionally, double finetuning usually performs similarly to only including genuine Western Armenian-English parallel set. In EN $\rightarrow$ HYW direction, information from a matching domain is more important and the training data has to have a high portion of examples from the matching domain to keep higher scores, any non-matching additional data lowers the portion and causes confusion. In HYW $\rightarrow$ EN direction this is generally not the case as the inclusion of more data generally results in

very slight improvements but definitely no deterioration. The in-depth analysis experiment regarding the effect of matching domain against matching language has shown that on both of the subsets the matching domain information is more important than matching language information, however in EN → HYW direction training without information that matches both language-wise and domain-wise results in huge drops in translation performance. In HYW → EN direction the drop in performance is not so critical. This is presumably caused by the pretrain knowledge of the NLLB model. Another outcome of the first in-depth analysis experiment is that the quality of Western Armenian Wikipedia is inferior to its Eastern counterpart as the knowledge gained for Eastern Armenian Wikipedia is more valuable for translating texts of Western Armenian Wikipedia. The second in-depth analysis experiment showed that the token overlap between Western and Eastern Armenian datasets is already quite high and the attempt to increase it with custom encoding introduces complexity to the learning process in the form of longer sequences and therefore results in lower scores.

We have also investigated three research questions that shed light on the features of Western Armenian regarding neural machine translation:

RQ1: *Which known methods used in data collection, preprocessing, or implementation of the machine translation model could be used to improve the translation quality to/from Western Armenian?*

We used several methods and compared them in various scenarios and additional experiments. Specifically, we employed the training schemes of double and joint finetuning, data augmentation techniques, and custom-encoded training data to maximize knowledge transfer. Although there is no single superior technique that boosts the translation performance universally, each method proved to be useful either in a translation direction or for a subset. Scenario 2 showed that for EN → HYW direction in a general sense the double finetuned model was the best or the second best model with a similar general performance to the NLLB + HYW model, indicating introducing Eastern Armenian knowledge in this manner brings less confusion and in some specific cases even benefits to the translation quality (see Hamazkayin and Hayern Aysor subsets in supervised evaluation). While observing the supervised and unsupervised subset in a broader sense, generally the model with the highest portion of the matching domain within its training data gets the highest score for that subset. This effect was less impactful in HYW → EN direction, where instead the model with the larger training data performs better.

RQ2: *How do (multilingual) machine translation models trained on Eastern Armenian data perform while translating to/from Western Armenian?*

We trained models with no parallel Western Armenian data and evaluated their performance in Scenario 1. Results have shown that the baseline NLLB model and Eastern Armenian finetuned models have an acceptable performance while translating in HYW → EN direction. As expected, the models have performed poorly in the opposite direction however feeding monolingual Western Armenian data to generate synthetic example pairs and training upon them has proved to be a cheap and quick solution to boost performance in this direction.

RQ3: *How can we utilize data/knowledge of Eastern Armenian when training for Western Armenian?*

We used both joint and double finetuning training schemes to utilize Eastern Armenian knowledge. Both techniques have excelled in some instances. We also tried to maximize the knowledge transfer by employing a custom encoding that tries to minimize the amount of out-of-vocabulary tokens and maximize the number of tokens that could be found in both Western and Eastern Armenian training examples. However, the custom encoding's theorized benefits were overshadowed by the complexity introduced by longer sequences. To sum up, Eastern Armenian data is similar enough to contain useful information however Western Armenian data is essential to generate correct translations. In addition to semantic and syntactic divergence, the differences in orthography and romanization introduce another level of complexity to knowledge transfer. Possibly, learning the mapping between Western and Eastern Armenian morphemes could greatly increase the translation quality, since the languages have the same origin and share a substantial amount of linguistic features.

7.1. Shortcomings

In this section, we accumulate and analyze the aspects of this work that fall short of expectations. Analyzing these aspects not only ensures more transparent communication but also points out where to improve next.

As this work is in the scope of a master's thesis, all the subprocesses such as data collection, preprocessing, and model training are done by a single individual. Although the supervision is done as intensively as possible, there could be always mistakes, especially in dataset generation.

We investigate the shortcomings in their respective categories.

7.1.1. Data

As the name low-resource language suggests, it is naturally expected never to find adequate amounts of high-quality data directly. As indicated by the results, the translation scores receive a significant drop in unsupervised testsets, pointing out that the training data is not general enough. Although our corpus includes texts for various domains, there is still great room for an improved parallel corpus.

The creation of a corpus mainly depends on two major subprocesses: Data collection and alignment/mining. As Western Armenian is one of the languages that is yet to be fully exposed to the internet, finding digital material is not trivial and the digital material is not yet in text form, requiring additional tools like OCR. Open-source OCR software like Tesseract OCR yields suboptimal results as mentioned in chapter 4 and subsection 4.3.1, requiring manual supervision for correction and therefore additional time.

Aligning/mining is perhaps the heart of corpus creation. In most of the collected documents, the translations do not fully correspond to each other and these kinds of matches must be removed. Handling this manually becomes impossible with the sizes of training data for neural models. Bitext miner models come in aid to this situation, but

sadly LASER, the bitext mining model provided by Meta along with NLLB does not support Western Armenian and the issue[1] stated in GitHub mentions the mined bitexts of Eastern Armenian are either missing or have low quality. This has led to the choice of a "DIY" mining method as explained in subsection 4.3.3, instead of relying on LASER, which can also yield suboptimal results. This also has shifted the focus during data collection from comparable bitexts to direct translations.

Going back to a higher perspective, finding Western Armenian-English bitext is rather tough because the pair is quite uncommon relative to languages like Turkish, Arabic, and French; even though the bitexts in these pairs also have to be digitized. However, an English-centric approach is based on our estimation more beneficial in terms of both exposure and available technical resources.

7.1.2. Models

All the models were finetuned on the baseline model of NLLB. In order to have a better assessment of the different architectural approaches, comparisons with other state-of-the-art models would be most suitable. This would however increase the scale of work and amount of the model sets with each additional baseline. The trained models do not particularly use the power of multilinguality although they each have the ability. As mentioned in the previous subsection, the inclusion of more pairs with Western Armenian should be prioritized to better utilize multilingual embeddings.

7.1.3. Evaluation

As previously mentioned, the evaluation is performed only under standard automatic metrics. These generally lack the contextual information about what the models do right or wrong, unlike an assessment by a human. This was never considered in the scope of this work, but it is surely interesting and insightful to obtain the assessment of (non-)professional individuals of different language proficiency levels and should be included in future research.

7.2. Future Work

Working with a language that has flown under the radar of the natural language processing community has its difficulties, but there are even more opportunities to create, improve, and gain new insights. We want to list a few suggestions on how the research about Western Armenian natural language processing could continue.

1. **Building necessary tools for data collection/corpus building:** The quality of the training data is what makes a neural model stand out, therefore building tools that boost productivity and efficiency while creating Western Armenian corpora have the utmost importance. Most of the Western Armenian scripture is not in

[1] https://github.com/facebookresearch/LASER/issues/129

digital text form, so building an OCR software with a high level of confidence is perhaps the first step towards enriching the sources.

Segmenting sentences is an important preprocessing step in data creation and both variants of Armenian currently lack a neural sentence segmentation model and rely on rule-based solutions which are of a very rigid nature and unable to handle "erroneous" inputs (e.g. the usage of the colon :, instead of Armenian end-of-sentence character :) and thus requiring an additional correction session.

Although both variants show some similarities, the results have shown that in order to translate in EN $\rightarrow$ HYW direction, Western Armenian content is definitely required. For individuals who are barely exposed to both variants will most likely struggle to disambiguate them. The Western Armenian content, which is actually translated from Eastern Armenian, contains parts that are not used in the Western variant. Such parts must be identified and discarded to ensure the quality of the corpus and is a hard task without the knowledge of both variants. A neural language identification tool that correctly identifies Eastern and Western Armenian will be therefore most beneficial.

For state-of-the-art translation performance, the models require training sets with sizes of billions of examples. Such training sets can only be built with mined bitexts from comparable texts. Extending existing bitext miners to support Western Armenian is perhaps the most impactful step to build better and larger corpora since it allows the opportunity to benefit from texts that are not necessarily direct translations of each other.

2. **Building Corpora:** This is a task of multiple dimensions. The Western Armenian corpus can be extended by new domains, by new language pairings, and in a broader sense by data that is suitable for different natural language processing tasks such as speech recognition, named entity recognition, etc.

3. **Building NLP Models:** Machine Translation is merely a subtask within the whole natural language processing environment. After building the necessary data for the models, other NLP tasks such as speech recognition, natural language understanding, summarization, etc. would further bolster the language's resistance against the threat of extinction. Additionally as previously mentioned there is a plethora of methods for improving the quality of performance of each NLP task. In this work, we investigated the most popular methods that suited the setting, which means there are still a lot of methods and techniques left to experiment.

   Shaping the word embeddings of the Western Armenian language is perhaps the most important step towards enabling the inclusion of the language in other natural language processing tasks. This not only ensures a baseline performance and minimum resources but also speeds up the data collection process. The word embeddings could first be generated in a monolingual setting for the sake of simplicity and then could be adapted into the multilingual environment to reach state-of-the-art standards.

Qualitative results have shown that orthographies of both Armenian variants play a significant role in knowledge accumulation and transfer, suggesting an orthographic translator between classical and modern Armenian orthography could eliminate complexities and generate better synthetic data, in a broader sense the same argument could be made for simply building a translator between Western and Eastern Armenian.

7.3. Final Word

In this work, we attempted to introduce the "definitively endangered" Western Armenian language. We investigated its current status regarding the NLP research; and what has been and is yet to be done. As a low-resource language that has a relatively low internet presence, we made an extensive search identifying potential resources that can be transformed into suitable training data for neural models. We also evaluated how the models perform when trained only with Eastern Armenian data or when Eastern Armenian data is given along with Western Armenian data in various ways, with the thought that the similarity of languages could boost the translation performance.

Our work contributes to the community with two end-products: 1) The first neural translation model that supports Western Armenian, which impacts more directly the communities by constructing a linguistic bridge between them; 2) the parallel corpus, which is aimed more toward the machine translation and natural language processing researchers. We certainly do not claim that both products have the quality to be the gold standard as there is yet large room to improve. Nevertheless, we hope that they could become the spark that starts the fire of natural language processing research for the Western Armenian language to assist in its modernization, and in its obtaining a larger place on the stage called the Internet. The contributions will be made public in the near future via a dedicated website. We want to continue on this journey by creating and curating more resources and by extending our research with the goal of helping to slow down or even eliminate the threat of extinction.

Bibliography

[1] Ahmed Abdelali et al. “The AMARA Corpus: Building Parallel Language Resources for the Educational Domain”. In: *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*. Reykjavik, Iceland: European Language Resources Association (ELRA), May 2014, pp. 1856–1862. URL: http://www.lrec-conf.org/proceedings/lrec2014/pdf/877_Paper.pdf.

[2] Hracheay Acharean. *Classification des dialectes arméniens*. H. Champion, 1909.

[3] Roee Aharoni, Melvin Johnson, and Orhan Firat. *Massively Multilingual Neural Machine Translation*. URL: https://arxiv.org/pdf/1903.00089.

[4] Alham Fikri Aji et al. *In Neural Machine Translation, What Does Transfer Learning Transfer?* 2020. DOI: 10.5167/UZH-188224.

[5] R. Ananthakrishnan et al. “Some issues in automatic evaluation of english-hindi mt: more blues for bleu”. In: *Icon* 64 (2007).

[6] Gor Arakelyan, Karen Hambardzumyan, and Hrant Khachatrian. *Towards JointUD: Part-of-speech Tagging and Lemmatization using Recurrent Neural Networks*. URL: https://arxiv.org/pdf/1809.03211.

[7] Naveen Arivazhagan et al. *Massively Multilingual Neural Machine Translation in the Wild: Findings and Challenges*. URL: https://arxiv.org/pdf/1907.05019.

[8] Timofey Arkhangelskiy, Oleg Belyaev, and Arseniy Vydrin. “The creation of large-scale annotated corpora of minority languages using UniParser and the EANC platform”. In: *Proceedings of COLING 2012: Posters*. 2012, pp. 83–92.

[9] Mikel Artetxe and Holger Schwenk. *Margin-based Parallel Corpus Mining with Multilingual Sentence Embeddings*. 2018. DOI: 10.18653/v1/P19-1309. URL: https://arxiv.org/pdf/1811.01136.

[10] Mikel Artetxe and Holger Schwenk. “Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond”. In: *Transactions of the Association for Computational Linguistics* 7 (2019), pp. 597–610. ISSN: 2307-387X. DOI: 10.1162/tacla00288.

[11] Mikel Artetxe et al. *Unsupervised Neural Machine Translation*. URL: https://arxiv.org/pdf/1710.11041.

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

[13] Karen Avetisyan. "Dialects Identification of Armenian Language". In: *Proceedings of the Workshop on Processing Language Variation: Digital Armenian (DigitAm) within the 13th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, 2022, pp. 8–12. URL: https://aclanthology.org/2022.digitam-1.2.

[14] Karen Avetisyan and Tsolak Ghukasyan. *Word Embeddings for the Armenian Language: Intrinsic and Extrinsic Evaluation*. URL: https://arxiv.org/pdf/1906.03134.

[15] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. *Layer Normalization*. URL: https://arxiv.org/pdf/1607.06450.

[16] Arun Babu et al. *XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale*. URL: https://arxiv.org/pdf/2111.09296.

[17] Varuzhan Baghdasaryan. "ArmSpeech-POS: Eastern Armenian Part-of-Speech Tagged Corpus". In: *International Journal of Scientific Advances (IJSCIA)* 4.2 (2023), pp. 265–270.

[18] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. *Neural Machine Translation by Jointly Learning to Align and Translate*. URL: https://arxiv.org/pdf/1409.0473.

[19] Satanjeev Banerjee and Alon Lavie. "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments". In: *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. 2005, pp. 65–72.

[20] Ankur Bapna et al. *Building Machine Translation Systems for the Next Thousand Languages*. URL: https://arxiv.org/pdf/2205.03983.

[21] Adrien Barbaresi. "Trafilatura: A Web Scraping Library and Command-Line Tool for Text Discovery and Extraction". In: *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, 2021, pp. 122–131. URL: https://aclanthology.org/2021.acl-demo.15.

[22] Francesco Barbieri, Luis Espinosa Anke, and Jose Camacho-Collados. *XLM-T: Multilingual Language Models in Twitter for Sentiment Analysis and Beyond*. URL: https://arxiv.org/pdf/2104.12250.

[23] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc, 2009.

[24] Piotr Bojanowski et al. "Enriching Word Vectors with Subword Information". In: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 135–146. ISSN: 2307-387X.

[25] Francis Bond and Ryan Foster. "Linking and extending an open multilingual wordnet". In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2013, pp. 1352–1362.

[26] Peter F. Brown et al. “A statistical approach to machine translation”. In: *Computational linguistics* 16.2 (1990), pp. 79–85.

[27] Peter F. Brown et al. “Class-based n-gram models of natural language”. In: *Computational linguistics* 18.4 (1992), pp. 467–480.

[28] Chris Callison-Burch, Miles Osborne, and Philipp Koehn. “Re-evaluating the Role of Bleu in Machine Translation Research”. In: *11th Conference of the European Chapter of the Association for Computational Linguistics*. Trento, Italy: Association for Computational Linguistics, 2006, pp. 249–256. URL: https://aclanthology.org/E06-1032.

[29] George L. Campbell. *Concise compendium of the world’s languages*. Routledge, 2003.

[30] Isaac Caswell, Ciprian Chelba, and David Grangier. *Tagged Back-Translation*. URL: https://arxiv.org/pdf/1906.06442.

[31] Samuel Chakmakjian and Ilaine Wang. “Towards a Unified ASR System for the Armenian Standards”. In: *Proceedings of the Workshop on Processing Language Variation: Digital Armenian (DigitAm) within the 13th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, 2022, pp. 38–42. URL: https://aclanthology.org/2022.digitam-1.6.

[32] Kyunghyun Cho et al. *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. URL: https://arxiv.org/pdf/1406.1078.

[33] Kyunghyun Cho et al. *On the Properties of Neural Machine Translation: Encoder-Decoder Approaches*. URL: https://arxiv.org/pdf/1409.1259.

[34] Christos Christodouloupoulos and Mark Steedman. “A massively parallel corpus: the bible in 100 languages”. In: *Language resources and evaluation* 49 (2015), pp. 375–395.

[35] Alexandra Chronopoulou, Dario Stojanovski, and Alexander Fraser. *Reusing a Pretrained Language Model on Languages with Limited Corpora for Unsupervised NMT*. URL: https://arxiv.org/pdf/2009.07610.

[36] 文本翻_机器翻-百度*AI*放平台. 18/08/2023. URL: https://ai.baidu.com/tech/mt/text_trans.

[37] James Clackson. “The linguistic relationship between Armenian and Greek”. PhD thesis. University of Cambridge, 1992.

[38] Alexis Conneau et al. *Unsupervised Cross-lingual Representation Learning at Scale*. URL: https://arxiv.org/pdf/1911.02116.

[39] Raj Dabre, Atsushi Fujita, and Chenhui Chu. “Exploiting Multilingualism through Multistage Fine-Tuning for Low-Resource Neural Machine Translation”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 1410–1416. DOI: 10.18653/v1/D19-1146. URL: https://aclanthology.org/D19-1146.

[40] Raj Dabre, Tetsuji Nakagawa, and Hideto Kazawa. "An empirical study of language relatedness for transfer learning in neural machine translation". In: *Proceedings of the 31st Pacific Asia conference on language, information and computation*. 2017, pp. 282–286.

[41] *DeepL Translate: The world's most accurate translator*. 29/07/2023. URL: https://www.deepl.com/en/translator.

[42] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. URL: https://arxiv.org/pdf/1810.04805.

[43] *Dictionary and online translation - Yandex Translate*. 18/08/2023. URL: https://translate.yandex.com/en/.

[44] Hossep Dolatian, Daniel Swanson, and Jonathan Washington. "A Free/Open-Source Morphological Transducer for Western Armenian". In: *Proceedings of the Workshop on Processing Language Variation: Digital Armenian (DigitAm) within the 13th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, 2022, pp. 1–7. URL: https://aclanthology.org/2022.digitam-1.1.

[45] Anaid Donabedian-Demopoulos. *Middle East and Beyond-Western Armenian at the crossroads: A sociolinguistic and typological sketch*. 2018.

[46] Anaid Donabedian-Demopoulos and Nisan Boyacioglu. *La lemmatisation de l'arménien occidental avec NooJ*. 2007.

[47] Sebastian Drude, Intangible Cultural Heritage Unit's Ad Hoc Expert Group, et al. "Language vitality and endangerment". In: (2003).

[48] Nan Du et al. "GLaM: Efficient Scaling of Language Models with Mixture-of-Experts". In: *Proceedings of the 39th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri et al. Vol. 162. Proceedings of Machine Learning Research. PMLR, 2022, pp. 5547–5569. URL: https://proceedings.mlr.press/v162/du22c.html.

[49] Ethnologue. *Languages of the World*. 25/07/2023. URL: https://www.ethnologue.com/.

[50] Nicholas Evans and Stephen C. Levinson. "The myth of language universals: Language diversity and its importance for cognitive science". In: *Behavioral and brain sciences* 32.5 (2009), pp. 429–448.

[51] Marzieh Fadaee, Arianna Bisazza, and Christof Monz. *Data Augmentation for Low-Resource Neural Machine Translation*. 2017. DOI: 10.18653/v1/P17-2090. URL: https://arxiv.org/pdf/1705.00440.

[52] Angela Fan et al. "Beyond English-Centric Multilingual Machine Translation". In: *J. Mach. Learn. Res.* 22.1 (2021). ISSN: 1532-4435.

[53] Mikel L. Forcada, Gema Ginest\'\irez-Sánchez, and Francis M. Tyers. "Apertium: a free/open-source platform for rule-based machine translation". In: *Machine translation* 25 (2011), pp. 127–144.

[54] T. Ghukasyan and K. Avetisyan. “RESEARCH AND DEVELOPMENT OF A DEEP LEARNING-BASED LEMMATIZER FOR THE ARMENIAN LANGUAGE”. In: - (), p. 92.

[55] Tsolak Ghukasyan et al. “pioNER: Datasets and Baselines for Armenian Named Entity Recognition”. In: *2018 Ivannikov Ispras Open Conference (ISPRAS)*. 2018, pp. 56–61. DOI: 10.1109/ISPRAS.2018.00015.

[56] *Google Translate*. 29/07/2023. URL: https://translate.google.com/.

[57] Mitchell A. Gordon and Kevin Duh. *Distill, Adapt, Distill: Training Small, In-Domain Models for Neural Machine Translation*. URL: https://arxiv.org/pdf/2003.02877.

[58] Naman Goyal et al. “The Flores-101 Evaluation Benchmark for Low-Resource and Multilingual Machine Translation”. In: *Transactions of the Association for Computational Linguistics* 10 (2022), pp. 522–538. ISSN: 2307-387X. DOI: 10.1162/tacl{\textunderscore}a{\textunderscore}00474.

[59] Yvette Graham et al. “Continuous measurement scales in human evaluation of machine translation”. In: *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*. 2013, pp. 33–41.

[60] Edouard Grave et al. *Learning Word Vectors for 157 Languages*. URL: https://arxiv.org/pdf/1802.06893.

[61] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[62] Kevin Heffernan, Onur Çelebi, and Holger Schwenk. *Bitext Mining Using Distilled Sentence Representations for Low-Resource Languages*. URL: https://arxiv.org/pdf/2205.12654.

[63] Sepp Hochreiter. “Untersuchungen zu dynamischen neuronalen Netzen”. In: *Diploma, Technische Universität München* 91.1 (1991), p. 31.

[64] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

[65] Hossep Dolatian. *Armenian resources – Hossep Dolatian*. 18/08/2023. URL: https://you.stonybrook.edu/deovlet/armenian-resources/.

[66] Heinrich Hübschmann. “Über die Stellung des Armenischen im Kreise der indogermanischen Sprachen”. In: *Zeitschrift für vergleichende Sprachforschung auf dem Gebiete der Indogermanischen Sprachen* 23.1. H (1877), pp. 5–49. ISSN: 09372229.

[67] Inalco. *Le projet PRC "DALiH - Digitizing Armenian Linguistic Heritage" est lauréat de l’AAPG 2021 de l’ANR*. 2021. URL: http://www.inalco.fr/actualite/projet-prc-dalih-digitizing-armenian-linguistic-heritage-laureat-aapg-2021-anr.

[68] Mike Izbicki. “Aligning Word Vectors on Low-Resource Languages with Wiktionary”. In: *Proceedings of the Fifth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2022)*. Gyeongju, Republic of Korea: Association for Computational Linguistics, 2022, pp. 107–117. URL: https://aclanthology.org/2022.loresmt-1.14.

[69] Gevorg B. Jahukian. *On the position of Armenian in the Indo-European languages*. 1979.

[70] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. "GloVe: Global Vectors for Word Representation". In: *Empirical Methods in Natural Language Processing (EMNLP)*. 2014, pp. 1532–1543. URL: http://www.aclweb.org/anthology/D14-1162.

[71] Jeroen Ooms. *tesseract: Open Source OCR Engine*. 2023.

[72] Melvin Johnson et al. "Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation". In: *Transactions of the Association for Computational Linguistics* 5 (2017), pp. 339–351. ISSN: 2307-387X. DOI: 10.1162/tacla00065.

[73] Jonas Gehring et al. "Convolutional Sequence to Sequence Learning". In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 1243–1252. URL: https://proceedings.mlr.press/v70/gehring17a.html.

[74] Pratik Joshi et al. *The State and Fate of Linguistic Diversity and Inclusion in the NLP World*. URL: https://arxiv.org/pdf/2004.09095.

[75] Armand Joulin et al. *FastText.zip: Compressing text classification models*. URL: https://arxiv.org/pdf/1612.03651.

[76] Nigoghos Kalayjian. "Armenian Sentiment Analysis and Emotion Recognition Using Bidirectional Deep Learning Models". PhD thesis. 2022.

[77] Katharina Kann, Ophélie Lacroix, and Anders Søgaard. "Weakly Supervised POS Taggers Perform Poorly on Truly Low-Resource Languages". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 34.05 (2020), pp. 8066–8073. ISSN: 2159-5399. DOI: 10.1609/aaai.v34i05.6317.

[78] Armen Samuel Karamanian. "'He Wasn't Able to Understand What I Was Saying': The Experiences of Returnees' Speaking Western Armenian in 'Eastern' Armenia". In: *PORTAL Journal of Multidisciplinary International Studies* 16.1-2 (2019), pp. 120–140. DOI: 10.5130/pjmis.v16i1-2.6290.

[79] L. Khachatryan. "An Armenian grammar for proper names". In: *Formalising Natural Languages with NooJ: Selected Papers from the NooJ 2012 International Conference*. Ed. by Silberztein, M., and Anaïd Donabédian. Newcastle, UK: Cambridge Scholars Publishing, 2012.

[80] L. Khachatryan. "Formalization of proper names in the Western Armenian press." In: *Formalising Natural Languages with NooJ: Selected Papers from the NooJ 2011 Interna- tional Conference*. Ed. by Kristina Vučković et al. Newcastle, UK: Cambridge Scholars Publishing, 2011, pp. 75–85.

[81] Victoria Khurshudyan and Misha Daniel. "EASTERN ARMENIAN NATIONAL CORPUS". In: *"Dialog'2009"* 509-518 (2009). URL: https://shs.hal.science/halshs-01497348.

[82] Victoria Khurshudyan et al. “Eastern Armenian National Corpus: State of the Art and Perspectives”. In: *Proceedings of the Workshop on Processing Language Variation: Digital Armenian (DigitAm) within the 13th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, 2022, pp. 28–37. URL: https://aclanthology.org/2022.digitam-1.5.

[83] Ahmed El-Kishky et al. “CCAligned: A Massive Collection of Cross-lingual Web-Document Pairs”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 5960–5969. DOI: 10.18653/v1/2020.emnlp-main.480. URL: https://www.aclweb.org/anthology/2020.emnlp-main.480.

[84] Ahmed El-Kishky et al. *XLEnt: Mining a Large Cross-lingual Entity Dataset with Lexical-Semantic-Phonetic Word Alignment*. 2021. arXiv: 2104.08597 [cs.CL].

[85] S. C. Kleene. “Representation of Events in Nerve Nets and Finite Automata”. In: *Automata Studies. (AM-34)*. Ed. by J. McCarthy and C. E. Shannon. Annals of Mathematics Studies. Princeton, NJ: Princeton University Press, 1956, pp. 3–42. ISBN: 9781400882618. DOI: 10.1515/9781400882618-002.

[86] Guillaume Klein et al. “OpenNMT: Open-Source Toolkit for Neural Machine Translation”. In: *Proceedings of ACL 2017, System Demonstrations*. Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 67–72. URL: https://www.aclweb.org/anthology/P17-4012.

[87] Tom Kocmi and Ondřej Bojar. “Trivial Transfer Learning for Low-Resource Neural Machine Translation”. In: *Proceedings of the Third Conference on Machine Translation: Research Papers* (), pp. 244–252. DOI: 10.18653/v1/W18-6325. URL: https://arxiv.org/pdf/1809.00357.

[88] Philipp Koehn and Christof Monz. “Manual and automatic evaluation of machine translation between european languages”. In: *Proceedings on the Workshop on Statistical Machine Translation*. 2006, pp. 102–121.

[89] Philipp Koehn 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*. Prague, Czech Republic: Association for Computational Linguistics, 2007, pp. 177–180. URL: https://aclanthology.org/P07-2045.

[90] Garry Kuwanto et al. *Low-Resource Machine Translation Training Curriculum Fit for Low-Resource Languages*. URL: https://arxiv.org/pdf/2103.13272.

[91] Surafel M. Lakew, Mauro Cettolo, and Marcello Federico. *A Comparison of Transformer and Recurrent Neural Networks on Multilingual Neural Machine Translation*. URL: https://arxiv.org/pdf/1806.06957.

[92] Percy Liang. “Semi-supervised learning for natural language”. PhD thesis. Massachusetts Institute of Technology, 2005.

[93] Yu-Hsiang Lin et al. *Choosing Transfer Languages for Cross-Lingual Learning*. URL: https://arxiv.org/pdf/1905.12688.

[94] Pierre Lison and Jörg Tiedemann. "Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles". In: (2016).

[95] Qi Liu, Matt Kusner, and Phil Blunsom. "Counterfactual Data Augmentation for Neural Machine Translation". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, 2021, pp. 187–197. DOI: 10.18653/v1/2021.naacl-main.18. URL: https://aclanthology.org/2021.naacl-main.18.

[96] Yinhan Liu et al. "Multilingual Denoising Pre-training for Neural Machine Translation". In: *Transactions of the Association for Computational Linguistics* 8 (2020), pp. 726–742. ISSN: 2307-387X. DOI: 10.1162/tacla00343.

[97] Thang Luong et al. "Addressing the Rare Word Problem in Neural Machine Translation". In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Beijing, China: Association for Computational Linguistics, 2015, pp. 11–19. DOI: 10.3115/v1/P15-1002. URL: https://aclanthology.org/P15-1002.

[98] Shuming Ma et al. *DeltaLM: Encoder-Decoder Pre-training for Language Generation and Translation by Augmenting Pretrained Multilingual Encoders*. URL: https://arxiv.org/pdf/2106.13736.

[99] Enrique Manjavacas, Ákos Kádár, and Mike Kestemont. "Improving Lemmatization of Non-Standard Languages with Joint Learning". In: *NAACL-HLT* (). URL: https://arxiv.org/pdf/1903.06939.

[100] Marat M. Yavrumyan. "Universal Dependencies for Armenian." In: *International Conference on Digital Armenian, Abstracts*.

[101] Benjamin Marie, Raphael Rubino, and Atsushi Fujita. "Tagged Back-translation Revisited: Why Does It Really Work?" In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, 2020, pp. 5990–5997. DOI: 10.18653/v1/2020.acl-main.532. URL: https://aclanthology.org/2020.acl-main.532.

[102] Hrach Martirosyan. "The place of Armenian in the Indo-European language family: the relationship with Greek and Indo-Iranian". In: *Journal of language relationship* 10.1 (2013), pp. 85–138.

[103] Antoine Meillet. *Les dialectes indo-européens*. Vol. 1. E. Champion, 1922.

[104] Tomas Mikolov et al. "Distributed representations of words and phrases and their compositionality". In: *Advances in neural information processing systems* 26 (2013).

[105] Tomas Mikolov et al. *Efficient Estimation of Word Representations in Vector Space*. URL: https://arxiv.org/pdf/1301.3781.

[106] Tomas Mikolov et al. "Recurrent neural network based language model". In: *Interspeech*. Vol. 2. 2010, pp. 1045–1048.

[107] Scott Miller, Jethran Guinness, and Alex Zamanian. “Name tagging with word clusters and discriminative training”. In: *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*. 2004, pp. 337–342.

[108] Toan Q. Nguyen and David Chiang. *Transfer Learning across Low-Resource, Related Languages for Neural Machine Translation*. URL: https://arxiv.org/pdf/1708.09803.

[109] Kishore Papineni et al. “Bleu: a method for automatic evaluation of machine translation”. In: *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 2002, pp. 311–318.

[110] Holger Pedersen. “Armenisch und die Nachbarsprachen”. In: *Zeitschrift für vergleichende Sprachforschung auf dem Gebiete der Indogermanischen Sprachen* 39.3 (1906), pp. 334–485. ISSN: 09372229. URL: http://www.jstor.org/stable/40846285.

[111] Holger Pedersen. *Le groupement des dialectes indo-européens*. Vol. 11. Høst, 1925.

[112] Maja Popović. “chrF: character n-gram F-score for automatic MT evaluation”. In: *Proceedings of the tenth workshop on statistical machine translation*. 2015, pp. 392–395.

[113] Matt Post. “A Call for Clarity in Reporting BLEU Scores”. In: *Proceedings of the Third Conference on Machine Translation (WMT18)* (). URL: https://arxiv.org/pdf/1804.08771.

[114] Alec Radford et al. “Improving language understanding by generative pre-training”. In: (2018).

[115] Surangika Ranathunga et al. “Neural Machine Translation for Low-Resource Languages: A Survey”. In: *ACM Comput. Surv.* 55.11 (2023). ISSN: 0360-0300. DOI: 10.1145/3567592.

[116] Ricardo Rei et al. *COMET: A Neural Framework for MT Evaluation*. URL: https://arxiv.org/pdf/2009.09025.

[117] Nils Reimers and Iryna Gurevych. “Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Nov. 2020. URL: https://arxiv.org/abs/2004.09813.

[118] Nils Reimers and Iryna Gurevych. *Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging*. URL: https://arxiv.org/pdf/1707.09861.

[119] Frank Rosenblatt. “The perceptron: a probabilistic model for information storage and organization in the brain”. In: *Psychological review* 65.6 (1958), p. 386.

[120] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. “Learning representations by back-propagating errors”. In: *nature* 323.6088 (1986), pp. 533–536.

[121] Piotr Rybak and Alina Wróblewska. *Semi-Supervised Neural System for Tagging, Parsing and Lematization*. 2020. DOI: 10.18653/v1/K18-2004. URL: https://arxiv.org/pdf/2004.12450.

[122] Nipun Sadvilkar and Mark Neumann. "PySBD: Pragmatic Sentence Boundary Disambiguation". In: *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*. Online: Association for Computational Linguistics, 2020, pp. 110–114. URL: https://www.aclweb.org/anthology/2020.nlposs-1.15.

[123] Happy Scribe. *Armenian Transcription Service | Armenian Audio to Text*. 18/08/2023. URL: https://www.happyscribe.com/transcribe-armenian.

[124] Rico Sennrich, Barry Haddow, and Alexandra Birch. *Improving Neural Machine Translation Models with Monolingual Data*. URL: https://arxiv.org/pdf/1511.06709.

[125] Rico Sennrich, Barry Haddow, and Alexandra Birch. "Neural Machine Translation of Rare Words with Subword Units". In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, 2016, pp. 1715–1725. DOI: 10.18653/v1/P16-1162. URL: https://aclanthology.org/P16-1162.

[126] Noam Shazeer et al. *Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer*. URL: https://arxiv.org/pdf/1701.06538.

[127] SIL International ISO 639-3 Registration Authority. *Registration Authority decision on Change Request no. 2017-023: to create the code element [hyw] for Western Armenian*. URL: https://iso639-3.sil.org/sites/iso639-3/files/change_requests/2017/CR_Comments_2017-023.pdf.

[128] Max Silberztein. "NooJ: a linguistic annotation system for corpus processing". In: *Proceedings of HLT/EMNLP 2005 Interactive Demonstrations*. 2005, pp. 10–11.

[129] Aaron Smith, Christian Hardmeier, and Joerg Tiedemann. "Climbing Mont BLEU: The Strange World of Reachable High-BLEU Translations". In: *Proceedings of the 19th Annual Conference of the European Association for Machine Translation*. 2016, pp. 269–281. URL: https://aclanthology.org/W16-3414.

[130] Matthew Snover et al. "A Study of Translation Edit Rate with Targeted Human Annotation". In: *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*. Cambridge, Massachusetts, USA: Association for Machine Translation in the Americas, 2006, pp. 223–231. URL: https://aclanthology.org/2006.amta-papers.25.

[131] Miloš Stanojević et al. "Results of the WMT15 metrics shared task". In: *Proceedings of the Tenth Workshop on Statistical Machine Translation*. 2015, pp. 256–273.

[132] Amanda Stent, Matthew Marge, and Mohit Singhai. "Evaluating evaluation methods for generation in the presence of variation". In: *International conference on intelligent text processing and computational linguistics*. 2005, pp. 341–351.

[133] NLLB Team et al. *No Language Left Behind: Scaling Human-Centered Machine Translation*. URL: https://arxiv.org/pdf/2207.04672.

[134] Benoît Thouin. "The METEO system". In: *Translating and the Computer: Practical experience of machine translation*. 1981.

[135] Jörg Tiedemann. “OPUS-Parallel Corpora for Everyone”. In: *Baltic Journal of Modern Computing* 4.2 (2016).

[136] Sh T. Tigranyan and T. G. Ghukasyan. “Post-OCR Correction of Armenian Texts Using Neural Networks”. In: - , : - 2 (2020), p. 22.

[137] *Transcribe Armenian Audio into Text - Armenian | Vocalmatic*. 18/08/2023. URL: https://vocalmatic.com/languages/transcribe-armenian-armenian-to-text.

[138] United Nations Educational, Scientific and Cultural Organization. *Atlas of the World’s Languages in Danger*. 3. ed., entirely rev., enlarged and updated. Paris, France, 2010. ISBN: 9789231040955.

[139] Shyam Upadhyay, Jordan Kodner, and Dan Roth. *Bootstrapping Transliteration with Constrained Discovery for Low-Resource Languages*. URL: https://arxiv.org/pdf/1809.07807.

[140] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[141] Chahan Vidal-Gorène, Victoria Khurshudyan, and Anaïd Donabédian-Demopoulos. “Recycling and Comparing Morphological Annotation Models for Armenian Diachronic-Variational Corpus Processing”. In: *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*. Barcelona, Spain (Online): International Committee on Computational Linguistics (ICCL), 2020, pp. 90–101. URL: https://aclanthology.org/2020.vardial-1.9.

[142] David Vilar et al. “Human evaluation of machine translation through binary system comparisons”. In: *Proceedings of the Second Workshop on Statistical Machine Translation*. 2007, pp. 96–103.

[143] Hongyu Wang et al. *DeepNet: Scaling Transformers to 1,000 Layers*. URL: https://arxiv.org/pdf/2203.00555.

[144] Rui Wang and Ricardo Henao. “Wasserstein Cross-Lingual Alignment For Named Entity Recognition”. In: *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2022, pp. 8342–8346. DOI: 10.1109/ICASSP43922.2022.9746120.

[145] Warren Weaver. “Translation”. In: *Proceedings of the Conference on Mechanical Translation*. 1952.

[146] Ruibin Xiong et al. “On Layer Normalization in the Transformer Architecture”. In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 10524–10533. URL: https://proceedings.mlr.press/v119/xiong20b.html.

[147] Linting Xue et al. *mT5: A massively multilingual pre-trained text-to-text transformer*. URL: https://arxiv.org/pdf/2010.11934.

[148] Xiang Yu, Ngoc Thang Vu, and Jonas Kuhn. “Ensemble Self-Training for Low-Resource Languages: Grapheme-to-Phoneme Conversion and Morphological Inflection”. In: *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*. Online: Association for Computational Linguistics, 2020, pp. 70–78. DOI: 10.18653/v1/2020.sigmorphon-1.5. URL: https://aclanthology.org/2020.sigmorphon-1.5.

[149] Mengjiao Zhang and Jia Xu. “Byte-based Multilingual NMT for Endangered Languages”. In: *Proceedings of the 29th International Conference on Computational Linguistics*. Gyeongju, Republic of Korea: International Committee on Computational Linguistics, 2022, pp. 4407–4417. URL: https://aclanthology.org/2022.coling-1.388.

[150] Barret Zoph et al. “Designing effective sparse expert models”. In: *arXiv preprint arXiv:2202.08906* 2 (2022).

[151] Մեծ Հայք համազգային ցանց. 12/08/2022. URL: https://aws.ican24.net/.

[152] Վիճակագրութիւն — Ուիքիփեդիա. 11/08/2023. URL: https://hyw.wikipedia.org/wiki/%D5%8D%D5%BA%D5%A1%D5%BD%D5%A1%D6%80%D5%AF%D5%B8%D5%B2:%D5%8E%D5%AB%D5%B3%D5%A1%D5%AF%D5%A1%D5%A3%D6%80%D5%B8%D6%82%D5%A9%D5%B5%D5%B8%D6%82%D5%B6.

A. Appendix

A.1. Default Training Parameters

Parameter	Value	Parameter	Value
-adam-betas	(0.9, 0.98)	-langs	// path to flores200 languages file
-adam-eps	1e-06	-lang-pairs	eng_Latn-hye_Armn, hye_Armn-eng_Latn
-add-data-source-prefix-tags		-lr	5e-03
-arch	transformer	-lr-scheduler	inverse_sqrt
-attention-dropout	0.1	-max-epoch	25
-batch-size	32768	-max-source-positions	1024
-best-checkpoint-metric	nll_loss	-max-target-positions	1024
-clip-norm	0.0	-max-tokens	512
-criterion	label_smoothed_cross_entropy	-max-tokens-valid	1024
-decoder-attention-heads	16	-memory-efficient-fp16	
-decoder-embed-dim	1024	-min-params-to-wrap	10
-decoder-ffn-embed-dim	4096	-on-cpu-convert-precision	
-decoder-langtok		-optimizer	adam
-decoder-layers	12	-relu-dropout	0.0
-decoder-normalize-before		-replication-count	1
-dropout	0.1	-sampling-method	temperature
-enable-m2m-validation		-sampling-temperature	1
-encoder-attention-heads	16	-seed	420
-encoder-embed-dim	1024	-share-all-embeddings	
-encoder-ffn-embed-dim	4096	-skip-invalid-size-inputs-valid-test	
-encoder-langtok	src	-stop-min-lr	1e-09
-encoder-layers	12	-task	translation_multi_simple_epoch
-encoder-normalize-before		-use-local-shard-size	
-finetune-from-model	// path to nllb200-600m-distilled	-update-freq	1
-fixed-dictionary	// path to vocabulary file of nllb-200-600m-distilled	-warmup-updates	1000
-gradient-as-bucket-view		-warmup-init-lr	1e-07
-label-smoothing	0.1	-weight-decay	0.0

Table A.1.: Default Parameters for Training