MULTILINGUAL INFORMATION RETRIEVAL: A REPRESENTATION BUILDING PERSPECTIVE

by
Ion Madrazo

A dissertation
submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Computing
Boise State University

December 2019

BOISE STATE UNIVERSITY GRADUATE COLLEGE

DEFENSE COMMITTEE AND FINAL READING APPROVALS

of the dissertation submitted by

Ion Madrazo

Dissertation Title: Multilingual Information Retrieval: A representation building perspective

Date of Final Oral Examination: 1st December 2019

The following individuals read and discussed the dissertation submitted by student Ion Madrazo, and they evaluated the presentation and response to questions during the final oral examination. They found that the student passed the final oral examination.

Maria Soledad Pera, Ph.D.	Chair, Supervisory Committee
Michael Ekstrand, Ph.D.	Member, Supervisory Committee
Edoardo Serra, Ph.D.	Member, Supervisory Committee
Hoda Mehrpouyan, Ph.D.	Member, Supervisory Committee

The final reading approval of the dissertation was granted by Maria Soledad Pera, Ph.D., Chair of the Supervisory Committee. The dissertation was approved by the Graduate College.

Aitor, Ama, Atte, zuentzat.

ACKNOWLEDGMENTS

I would like to take advantage of this section to express my gratitude to all the people that in a more direct or indirect manner contributed to make this dissertation possible:

I really appreciate the help received from all the **committee members** and from the Computer Science department at **Boise State University**. You guided me to take the right path for succeeding in this dissertation.

Thank you to everybody at **Research Computing**, for providing all the help we required when using the R2 cluster. This work would not have been possible without the computational resources you provided.

Thank you to the **PIReT group** for the weekly discussions we had during these last years. You allowed me to find out about my errors while providing guidance for future ideas, making my work stronger and more relevant to the research community.

Eskerrik asko **Iker eta Mikel**, egunerokotasunin eman diuzten laguntziatik. Konfiau laiketan batemat gerku eukitziek asko lagundu izen ditt urte hauetan. Eman eurre, laste zeate ta zueke!

Eskerrik asko **ama, aita eta Aitor** astero mundun bestaldetik nerekin eoteatik. Holako urruti eotie goorra da danontzat, baine zuen babesa dauketela jakittiek asko lagundu ditt aurrea etten. Aitor lan hau zuretzat diju bereziki, holako moruko bat ettea allako zeala seguru naolako. Eman eurre!

Sole, no hay palabras suficientes para darte las gracias por todo lo que has hecho por

mi durante estos ultimos 3 años. Gracias por acogerme y estar siempre a mi lado en buenos y no tan buenos momentos. Al fin cierra este capitulo, uno de los mas importantes de mi vida, el cual no podria haber concluido sin tu ayuda. En este punto comienza una nueva aventura para ambos, en la cual siempre te mantendre como ejemplo a seguir para poder seguir progresando tal y como me enseñaste. #SvenniesAlwaysMakeItTrue

ABSTRACT

Information Retrieval (IR) has changed the way we access digital resources and satisfy our daily information needs. Popular IR tools like Search Engines, Recommendation Systems, and Automatic Question Answering sites, act as a deterrent for information overload while fostering (at least in theory) the democratization of access to resources. Yet, in their majority, IR tools are built with a traditional user in mind. This causes users who deviate from the norm, e.g., users with low educational background, visually-impaired users, or users who speak different languages, to be undeserved and thus struggle to find the information they require. In this manuscript, we present novel methodologies that can enable better adaptation of IR tools to non-traditional users. We focus on two aspects in which users can differ from the traditional: *language* and *reading skills*. We study and address such difficulties, and discuss how they affect IR systems. Particularly, we allocate research efforts to three main areas: (1) readability assessment, where we introduce the first featureless architecture, enabling it to be used in any language without specific tuning; (2) cross-lingual word embedding generation, where we address the English-dependency problem of state-of-the-art strategies via a hierarchical mapping strategy that takes advantage of the language family tree; and (3) cross-lingual sentence embedding generation, where we present a novel representation learning framework based on a hierarchical sequence-to-sequence model that enables better representations for low-resource languages. Each of the strategies that result from this work can be leveraged in the design of IR systems that better support

non-traditional users. In fact, to demonstrate how they can be integrated to address the needs of non-traditional users, we also conduct an analysis of four different readability assessment strategies (based on our three models) in terms of their language transfer capabilities, demonstrating the use of the aforementioned models in low-resource language scenarios. Despite the contributions presented, results indicate that there is still a long path towards building IR systems that fully address the needs of non-traditional users, in areas including representation of typologically isolated low-resource languages or more fine-grained multilingual readability assessment.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

EN – English

EU – Basque

ES – Spanish

IR – Information Retrieval

NE – Named Entity

NLP – Natural Language Processing

PoS – Part of Speech

TF-IDF – Term Frequency - Inverse Document Frequency

CHAPTER 1

INTRODUCTION

Information Retrieval (IR) is the area of study focused on identifying a set of resources within a given collection that are relevant to a user's information need [20]. Applications based on IR are ubiquitous when it comes to dealing with information overload, where only few resources (e.g., documents, websites, or products) among those in a large collection of mostly irrelevant resources are relevant to a user. Amid the most well-known IR applications we find (1) search engines [33], such as Google or Bing, which have become the main entry portal to the Internet for most users, (2) personalized recommendation services, which have proliferated among sites in the domain of books, movies, e-commerce, or social networks [202, 193], and (3) automatic question-answering applications, which are used by community question answering sites to improve their services by reducing question duplication and overall answering time [90], as well as by other IR applications in an effort to offer users concrete responses to their inquiries, instead of simple resource lists [267].

Most work pertaining to IR is focused on enhancing search, recommendation, and question answering tasks through algorithms and techniques designed with a traditional user in mind. However, there are groups of users for whom expectations of what is relevant differ from the traditional, causing their information needs to remain unfulfilled by existing IR tools. For example, consider expert domain users, who look for resources on a particular domain, multilingual users, who do not benefit from an IR tool that only offers resources in a single language, children, who might be interested in resources that differ from what appeals to adults, and users with low educational attainment, who might have difficulties understanding texts retrieved in response to queries posted on a search engine.

The need to consider non-traditional users in the retrieval process is becoming a major concern in the IR area. This is manifested by the emergence of workshops such as KidRec [78, 123], specifically focused on search and recommendation algorithms for children, or the increasing works on algorithm bias [73, 182], which evaluate to what extent current algorithms fulfill the information needs of users that deviate from the average in terms of gender, age, or any other demographic aspect from the prototypical average user.

The emerging interest in adapting IR applications to non-traditional users has given rise to multiple new areas which support the main retrieval tasks, especially focusing on designing strategies for more adequately representing resources as well as better filtering and presenting information to the users that deviate from the standard. Some supporting areas include:

- Strategies for automatically determining the reading level of resources in order to filter them for users with different reading comprehension capabilities [41].

- Document simplification techniques used to adapt retrieved resources to the reading level of the user [53].

- Summarizing and synthesizing techniques that support the use of IR applications by users with different attention deficits [194]

- Voice-based dialogue systems that aid visually impaired users in interacting with an IR application [206].
- Machine translation strategies that enable non-native users to understand retrieved resources [146].
- Multilingual document representation strategies that enable the retrieval of resources across different languages, allowing users that know multiple languages to obtain the full benefit of their knowledge [55].
- Domain-specific document representation strategies that can improve the relevance of retrieved resources in specific domains, such as medicine [118] .

In this dissertation work, we present novel solutions that existing IR applications can leverage for adaptation purposes and in turn better support non-traditional users. The spectrum of non-traditional users is too wide and their information needs vary too deeply among them to be considered as a unit. Therefore, in order to control for scope, we consider two aspects that affect the degree to which users differ from traditional ones: language and reading skills.

Reading skills

When it comes to search, recommendation, and question answering tasks, it is of special importance to provide users with resources they can fully understand, since otherwise these resources become non-relevant [45]. For this reason, IR applications have historically benefited from considering the readability level of documents when measuring the degree to which they are relevant to users. Rabbit [202], a book recommendation system for

K-12 students, uses a readability score for ensuring that books are not only of interest for readers, but also understandable for them. With a similar goal, but in a search context, Collins-Thompson and Callan [45] use a re-ranking strategy based on the readability of retrieved documents, so that the ones that are more adequate to the reading skill of the user are ranked higher. Kanungo and Orr [133] take advantage of readability assessment for improving the way summary snippets are created in Yahoo!, ensuring that generated snippets are more readable.

Most of the aforementioned applications estimate text readability via traditional readability formulas. These formulas rely on shallow features, which can be easily adapted to multiple languages and provide a simple way of determining text complexity. Despite their ease of adaptation, traditional formulas are known to lack precision in their readability estimations. In other words, they can classify nonsense text as simple to read, just because it contains short and frequently-used words [52]. For this reason, researchers have pursued more sophisticated methods for readability assessment that depend upon more in-depth text analysis [7, 93]. These formulas still take advantage of shallow features, but incorporate more complex features based on the syntax and semantics of the text. With the addition of new text complexity indicators, the tools became more precise, but at the same time more constrained regarding their language adaptability [26, 81].

The flexibility offered by traditional formulas was indeed one of their most valuable traits. This is reflected by the fact that traditional formulas are the ones more commonly used by the main stakeholders of readability assessment: educators, book publishers or developers of applications oriented to children and language learners. Flesch-Kincaid [88] is widely used by educators for assessing the complexity of texts, as mentioned by the Council of Chief School Officers [35]. Lexile [158] and Accelerated Reader (AR) [237], which are also based on traditional complexity indicators like sentence length and word difficulty [158, 209], are two of the standards used by most book publishers to categorize their books in terms of reading level [159]. Even in the IR area, most of the applications that take advantage of readability still favor the less precise, but adaptable technology developed in the 20^{th} century [173, 28, 45, 77, 133, 87, 202] evidencing the need to pursue formulas that are both as precise as state-of-the-art counterparts but as adaptable as traditional formulas.

Language

The use and quality of IR systems is not uniformly distributed among populations of different languages given that most work pertaining to IR is focused on enhancing search, recommendation, and question answering tasks through algorithms and techniques designed for a single language, primarily English. As a result, these strategies need to be adapted or re-implemented in order to be applied to resources and information needs expressed in different languages. This creates an uneven scenario, one where users that speak popular languages, such as English, have a large amount of IR tools at their disposal, while users of underrepresented languages, such as Basque, have few or none [24]. Moreover, users who are skilled at more that one language are required to use multiple IR tools in order to satisfy their information needs, when instead they could be using tools that retrieve resources in all the languages they know. Finally, work duplication also becomes an issue, as researchers study, design, and deploy strategies that are similar to each other, only differing on the language for which they are applied.

Multilingual Information Retrieval aims at designing IR applications that can work in multiple languages at the same time. Unfortunately, existing techniques in the multilingual IR area are still rudimentary as they are mostly focused on language pairs rather than on multiple languages, corpora is scarce, and generating language independent representations of resources is still an open task.

1.0.1 Topic Statement

Study and design strategies for adapting existing Information Retrieval tools for non traditional users, from a language and reading-skill perspective.

Manuscript Structure

The rest of this manuscript is organized as follows: in Chapter 2, we describe studies and other pertinent literature that provide foundational knowledge we draw from when outlining the research contributions defined in this manuscript. Thereafter, in Chapters 3 to 6, we present the proposed strategies and corresponding findings that can be used for adapting IR applications so that they can better serve non-traditional users.

In Chapter 3, we address the limitations of state-of-the-art readability assessment strategies regarding the need of language-dependent features. For doing so, we introduce the first featureless readability assessment strategy that directly relies on words as input. Not needing human-engineered features implies that the strategy can be applied to any language without requiring any adaptation. We assess the performance of the strategy in seven different languages to validate our claims and demonstrate that it is possible to build a readability assessment strategy that can work regardless of the input language.

Even if the strategy presented in Chapter 3 is multilingual, in the sense that it can be used in any language, it does not benefit from its multilingual nature. Its results are strongly dependant on the quality of the corpora available for the corresponding language and there is no knowledge transfer among languages, e.g., nothing from what the model learned for English readability assessment can be applied to Spanish or French. Cross-lingual embeddings i.e., word embeddings that have same representations regardless of the language, have proven to be successful for enabling learning transference among different languages. However, generating cross-lingual embeddings is still an open area. Existing cross-lingual embeddings are trained in a pairwise fashion, mapping the embeddings of all languages into a single pivot language. This creates a cross-lingual space that is strongly biased towards the pivot language, usually English [62]. The more typologically distant a language is from the pivot, the worse its mapping quality is [15], and thus the worse are the knowledge transfer options among languages.

In Chapter 4, we address the pivot dependency problem in cross-lingual word embeddings by creating a hierarchically compositional embedding space that does not rely on selecting a pivot language. Instead of a single pivot, our strategy takes advantage of the language family tree in order to generate multiple pivoting points that are selected that minimize differences between languages and maximize the overall quality of each space.

Word level knowledge can be useful for some tasks, however, representing syntactic information, such as the one found in sentences, can be important for other tasks. For example, in the case of readability assessment knowing which words are difficult can provide a good foundation. However, literature has shown that state-of-the-art readability assessment performance is only achieved when considering structural information, such as syntax [93, 7, 135, 172]. For this reason, it is important that we build not only word level cross-lingual representations but also sentence level representations, a task we address in Chapter 5.

Finally, in Chapter 6, we compare the performance of four strategies built with different levels of transfer learning capabilities. For doing so, we take the readability assessment task as case study and compare the model presented in Chapter 3, with two new models that take advantage of the strategies presented in Chapters 4 and 5, respectively.

As a by-product of our main research path (aimed at the breadth of the IR area), we created several other strategies oriented to adapt existing IR strategies to non-traditional users. In Appendix 1, we present the adaptation of a hashtag recommendation strategy based on the reading level of the user. In Appendix 2, we showcase the utility of incorporating the reading-level signal as part of a question answering strategy. Lastly, in Appendix 3, we explore the usage of readability assessment as part of a query recommendation strategy for children.

CHAPTER 2

BACKGROUND AND RELATED WORK

Given the focus of the dissertation work, we describe foundational related work in the areas of (i) readability assessment, (ii) multilingual IR, and (iii) cross-lingual word embeddings.

2.1 Readability Assessment

From the past six decades, different readability assessment systems have been developed [26, 81, 143]. We offer below an in-depth discussion on readability assessment, from traditional formulas to state-of-the-art techniques. We also discuss formulas applied to estimate the levels of complexity of various types of texts written in different languages.

2.1.1 Traditional Readability Assessment

Traditional readability formulas, such as Flesch [88], Dale-Chall [36], and Gunning FOG [6], make use of shallow features, mostly based on ratios of characters, terms, and sentences. Flesch [88] (in Equation 2.1) is based on a linear combination of the average length of words and average length of sentences in a document.

$$F = 206.835 - 1.015 \times (\frac{totalWords}{totalSentences}) - 84.6 \times (\frac{totalSyllabes}{totalWords}) \tag{2.1}$$

Kincaid et al. [141] adapted the Flesch formula to American education grade levels, in order to predict a grade level instead of a number between 0 and 100, as the traditional Flesch formula does. This updated strategy, also known as Flesch-Kincaid (in Equation 2.2), uses the same features as Flesch, i.e., length of words and sentences, but combines them using different weights.

$$FK = 0.39 \times (\frac{totalWords}{totalSentences}) + 11.8 \times (\frac{totalSyllabes}{totalWords}) - 15.59 \tag{2.2}$$

Other alternatives, such as Dale-Chall [51] (in Equation 2.3), introduce the concept of simple and complex terms, taking advantage of a manually-generated list[1] of 3000 easy terms. The frequency of these terms is used as an indicator of text complexity, together with the already known average sentence length.

$$DC = 15.79 \times (\frac{difficultWords}{totalSentences}) + 0.0496 \times (\frac{totalWords}{totalSentences}) \tag{2.3}$$

Similar to the Dale-Chall, Gunning Fog [109] (in Equation 2.4) also considers the occurrences of simple or complex terms. However, instead of using a list to define complex terms, Gunning's readability formula considers a term complex if it has more than 3 syllables.

$$FOG = \frac{totalWords}{totalSentences} \times \frac{totalComplex}{totalWords} \tag{2.4}$$

SMOG [180] is an improvement over Gunning Fog, in terms of precision. It takes

[1] The full list can be found in http://www.readabilityformulas.com/articles/dale-chall-readability-word-list.php

advantage of a non-linear strategy (in Equation 2.5) that combines the total number of complex terms in a text and the total number of sentences. The method to determine whether a term is complex is the same as the one used in the Gunning Fog formula.

$$SMOG = 1.0430 \times \sqrt{\frac{30 \times totalComplex}{totalSentences}} + 3.1291 \tag{2.5}$$

Lasbarhet's index (Equation 2.6), also known as LIX, predicts the difficulty to comprehend a text for a foreign reader. Similar to the aforementioned formulas, it is based on the frequency of occurrence of complex terms per sentence. A term is considered simple if it has less than 6 characters; the number of sentences is computed based on the number of periods in the text.

$$LIX = \frac{totalWords}{totalPeriods} \times \frac{totalComplex}{totalPeriods} \tag{2.6}$$

The formulas described thus far are a sample of the most popular among the hundreds available to date. Further details on existing traditional formulas (which are mostly based on sentence and term counts) can be found on the recent surveys [26, 81].

2.1.2 State-of-the-Art Readability Assessment

The simplicity of the traditional formulas, makes them easily adaptable to languages other than English. This is showcased in the Spaulding's readability formula for Spanish [228], which uses the same two indicators as Dale-Chall's and Gunning Fog's, i.e., ratio of difficult terms and average length of sentence in a document, with weights adapted to the Spanish language. Moreover, by being rudimentary enough to be computed manually, traditional formulas, provide a simple way of estimating a text's complexity. A teacher or a librarian could compute the formula from the first few pages of a book, and estimate whether the book is adequate for a reader without having to read the book in its entirety. Unfortunately, depending on the type of texts they are applied to, these formulas have been shown to lack precision in their readability level estimations. This is evidenced by Spaulding [228], who demonstrated that completely nonsensical text can be predicted to be easily readable by traditional readability formulas. For example, the phrase *sv eni sar ein de er*, would be considered as easily readable by all the aforementioned readability formulas, just because it has short terms, even if it is completely nonsensical, or the term *quark* which is considered simple by most of the traditional readability formulas, due to its length, despite being a high level technical term [250].

The increase in popularity of machine learning techniques and the need to improve predictive quality of traditional formulas lead the readability assessment into a new era of study. An era where readability formulas take advantage of supervised learning techniques to combine tens or hundreds of indicators. Even if shallow features are still included in current readability assessment tools, they are usually considered baseline features, and features that consider other language aspects, such as syntax or semantics, play a more significant role [26].

2.1.3 Readability Assessment by Languages

Adapting readability assessment tools to several languages have been the main focus for researchers on recent years. This is evidenced by the fact that there exist at least one prediction formula for each of the most popular languages spoken worldwide. A descrition and representative sample of these formulas and tools is below.

For **English**, the readability assessment system presented in [7] predicts only two levels of difficulty, simple or complex, using elaborate features, such as ambiguity among terms in a text. Other authors [80], orient their system for assessing the difficulty level of a text for people with intellectual disabilities, using features that intend to detect how well a text is structured. The readability prediction system for financial documents presented in [31] is based, among other features, on the presence of active voice and number of hidden verbs. It is also important to mention two commercial readability assessment tools, Lexile[2] and AR[3], which are widely-used among academic professionals in the USA and more than 150 publishers [159]. Even if their algorithms are not public, they are known to rely on shallow features [155]. The literature pertaining to readability for text in English is abundant. For a more in-depth discussion on readability assessment for texts in English refer to [26, 81].

In contrast to English, **Spanish** readability assessment has not seen any significant improvement regarding features in recent years, as most of the existing works are still based on shallow features. Among the well-known readability assessment tools for Spanish, SSR [228] examines sentence length and number of infrequent words per sentences, whereas LC and SCI [12] consider the density of low frequency words. Other alternatives, like the ones introduced in [64, 229], incorporate strategies to combine the aforementioned methods in thei quest to improve readability estimation.

Compared to other languages, **Basque** readability assessment is reduced to only one system. This is due to the fact that Basque is considered a minority language and shares little similarity with most spoken languages. So far, ErreXail [101] is the only system created for

[2] https://www.lexile.com/

[3] http://www.renaissance.com/products/accelerated-reader/atos-analyzer

Basque readability assessment. ErreXail predicts two different readability values, simple or complex, using features mostly based on ratios of common natural language processing labels, such as Part-of-Speech (PoS) tags or morphology annotations.

Similar to Basque, the literature for **Arabic** readability assessment is also very recent. Al-Ajlan et al. [4] propose a tool based on only two features: average letters per term and average terms per sentence. These features were analyzed using a Support Vector Machine in order to classify text as simple or complex. Forsyth [91] examine a significantly larger amount of features than previous studies, demonstrating the validity of lexical and discourse features for Arabic readability assessment. In a simpler approach, El-Haj and Rayson [74] present a modification of the Flesch formula. Apart from the common Flesch indicators, this formula also includes information about short, long and stress syllables, as well as textual aspects that are only found on formal texts.

For **Italian** and **Russian**, the research conducted by Dell'Orletta et al. [56] and Karpov et al. [135], respectively, demonstrate the importance of structural features for readability prediction. Both works combine several syntactic features, including features that measure the complexity of syntactic trees.

Unlike readability assessment tools for English, Spanish, and Italian, to name a few, structural features do not seem to have such a positive influence for **Chinese**. Therefore, most of the literature pertaining to Chinese readability assessment have been focused only on lexical features, such as the TF-IDF (Term Frequency - Inverse Document Frequency) of terms [39, 45].

Rather than focusing on the general reader, François and Fairon [93] develop a system for **French** with foreign language learners in mind. The objective was to determine which features were more important for a foreign language learner to understand a text. They tested lexical, syntactical and semantic features and showed that semantic ones performed poorly in their case. Uitdenbogerd [241] considers the same task. However, his study only focuses on English natives that learned French. As a novelty, he introduces a feature that considered the occurrence of true cognates, terms that were same or similar in both languages, since those terms are the ones than this audience easier learns. Wang [250] also relies on the use of true cognates for readability assessment, developing an automatic true cognate identifier.

2.1.4 Readability Assessment by Document Type

Traditional readability assessment has usually been oriented to relatively long text snippets [6, 36, 88]. While state-of-the-art [26, 91, 101] alternatives maintain this trend, recent works explore methods for assessing the readability of other types of document.

Several studies focus on the analysis of readability for single sentences [56, 135]. Most these studies are usually part of text simplification systems, which use readability assessment for choosing which sentences need simplification. Dell'Orletta et al. [56] develop a readability assessment tool for Italian sentences, combining lexical, syntactical and semantic features. De Clercq and Hoste [54], Karpov et al. [135] developed a similar study for Russian, making a big emphasis on syntactical features. Both studies concluded that structural features have the most relevance for sentence readability prediction.

Web-related readability assessment has also been studied. Web pages are usually challenging for readability assessment given their varied topics and formats. Collins-Thompson et al. [46] assesses the readability of search results by considering information in both the title and the snippet retrieved by the search engine, and the full content of the pointed web page. The authors take advantage of language models for predicting readability, since these models are the most adequate for predicting the readability of short and noisy texts, such as web pages and their snippets [46].

Yu and Miller [266] present a Firefox plugin to automatically enhance the readability of web pages for Asians that do not speak fluent English. For doing so, their systems considers several structures known to be complex for non-English native speakers and applied several transformations to make them more readable. Along similar lines, Kanungo and Orr [133] develop a readability assessment tool for search result summaries. Their system combines several traditional readability formulas, such as Flesch or Gunning-FOG, with some novel features specifically designed for their task. The latter refer to features that measure the number of strange characters or repeated keywords, in order to detect spam summaries. This is an important aspect for these type of documents, because spammers try to trick search engines with summaries full of keywords, that are usually recognized as simple by readability assessment tools [133].

Even if books also contain long snippets, which traditional readability formulas are able to handle, copyright regulations make book content difficult to obtain. In order to overcome this issue, Denning et al. [57] present a readability assessment tool for K-12 books, which goes beyond traditional textual content snippets. Their system explores available book metadata, such as its author or genre.

2.1.5 Applications of Readability Assessment

Educational applications have traditionally been the main focus for readability formulas. Popular tools such as Lexile and AR were specifically designed to help teachers and librarians select books for children. Both systems are currently commonly used by book publishers to catalog books given their readability level [159].

Readability assessment tools have also been incorporated into automatic book recommendation systems. Rabbit [202] suggests books for K-12 children considering multiple criteria that include several appealing factors for the reader as well as the readability score for the recommended books. This permits Rabbit to not only recommend books that are of interest to a reader, but also ensure that he is going to be able to understand them.

Text simplification also takes advantage of readability assessment tools [56, 80]. Knowing if a text needs to be simplified is an important prerequisite for such a system [56]. More specifically, being able to recognize which parts of a text are the ones making the text complex is also important. Single sentence readability assessment [56] has been used to handle both issues. Text simplification can be seen as an iterative process where a text can be infinitely simplified. For this task, knowing when to stop is also a must. Therefore, readability assessment has also been used to determine if a simplification is sufficient, both as an evaluation method or stopping criteria [229].

In some contexts, such as the medical domain [203] or food diseases [76], it is critical to provide people with documents that they can fully understand. For example, patients that properly understand documents disclosed to them before surgery, are known to be less anxious before the operation and obtain more satisfactory results during posterior treatment [203]. Therefore, some institutions are currently enforced by law to ensure that the documents they generate match the reading level of average people [27, 95]. Several studies [27, 95, 124, 200, 203] verify that this enforcement is indeed fulfilled, yet, most of the studies show that documents that are supposed to be suited for average readers, tend to be too specialized and even well-educated adults have trouble understanding them [27, 95, 124, 200, 203].

Web search engines are increasingly getting more personalized towards their users. With the goal of providing users with resources that are both of interest and match their level of understanding, several applications have incorporated a readability signal in their systems. Examples that do so include the work of Collins-Thompson and Callan [45] who re-rank resources based on the readability levels, and the work of Kanungo and Orr [133] who take advantage of readability assessment for improving the way summary snippets are created in Yahoo!.

Even if the application domains discussed in this section are the most prominent, they are not the only ones that benefit from high-quality readability assessment. Other applications, such as translation [114] or dyslexia-related studies [220] also depend upon complexity assessment.

2.1.6 Feature Fusioning Techniques in Readability Assessment

When estimating the readability level of a document, analyzing features in isolation is not enough. Instead, it is important to generate a single score that simultaneously considers the information captured in each individual feature. This leads to a more well-rounded assessment of the document and thus a better estimation of its level of difficulty.

In addressing the feature fusioning challenge for readability assessment, a variety of techniques have been considered [26]. Collins-Thompson and Callan [45] use a naive Bayes model, whereas Denning et al. [57] take advantage of a linear regression and François and Fairon [93] adopt a logistic regression approach. Most state-of-the-art systems, however, have used Support Vector Machines [81, 233, 218, 26] making this technique the most popular in the area.

2.1.7 Multilingual Readability Assessment

While the number of readability assessment systems dedicated to single languages is high, little research has been done regarding multilingualism. To the best of our knowledge, only two studies have applied their proposed strategies to more than one language [54, 172]. De Clercq and Hoste [54] present a readability assessment strategy for Dutch and English. Their study focuses on comparing which features are valuable for each on the languages analyzed, concluding that the best feature set for both languages is significantly similar. Madrazo Azpiazu and Pera [172] extend the number of languages compared to six (English, Spanish, Italian, French, Catalan, and Basque) and conduct a similar feature comparison. As a result, they determined that while typologically similar languages benefited from similar readability assessment features, this phenomenon did not occur for typologically isolated languages, such as Basque. Even if these studies shine alight on the multilingual aspects of readability assessment, the proposed strategies cannot be easily adapted to any arbitrary language. This is due to the fact that the proposed techniques are dependant on human engineered features that are created with a language in mind and need to be specifically adapted to provide valuable signal for other languages.

2.2 Multilingual Information Retrieval

We discuss below techniques that are background and/or core for the area of multilingual IR.

2.2.1 Automatic Machine Translation

Automatic Machine Translation refers to any strategy that given a text t_s in a source language, generates a text t_t in target language, where the meaning of t_t is as similar as possible to t_s. The similarity between t_s and t_t is conditioned by the strategy used for translation purposes, as well as by the availability of corpora. Automatic Machine Translation strategies can be classified into three categories:

- **Direct Translation.** In this case, pieces of text from t_s, such as words or phases, are matched using a translation dictionary and directly replaced by their translations. This is the simplest of the translation strategies and it can be useful in contexts where parallel corpora is not abundant or in IR strategies that require no structural information, e.g., bag-of-words comparisons [213, 22].

- **Rule Based Translation.** In this instance, translation depends upon syntactic rules that are either written by humans or automatically generated. Rule sets need to be written per language pair, and extending or fixing a problem in the translation system requires one to analyze hundreds of rules, which makes handling multiple languages incrementally tedious. Rule based systems can be useful for languages with scarce parallel resources, as humans can write rules without the need of large amounts of text [271, 37].

- **Statistical Translation.** This strategy relies on probability distributions learned from a parallel corpora to predict the most probable t_t for t_s. To estimate reliable probability distributions, this strategy requires large amounts of parallel corpora, and the precision of the translation is highly dependent on the size of it. This type of translation barely requires any human labor which makes it useful for tools that translate across multiple languages. It is the most common strategy used for building IR applications nowadays [163, 145].

Translation-based approaches were the default approach when researchers started to build early multilingual IR applications. However, as we describe in Section 2.2.3, state-of-the-art strategies find alternative techniques to avoid using machine translation, given its known lack of accuracy and processing cost [84].

2.2.2 Cross-lingual Embeddings

A *word embedding* is a numerical representation of a word in a multidimensional space [189]. Each dimension represents a latent feature which partially describes the meaning of a word, facilitating the comparison, in terms of meaning, of any word pair in a numerical way. Figure 2.1 illustrates word embeddings of *days of the week* and *months* in a 2-dimensional space. Note the clusters formed by embeddings associated with words of days of the week and months, reflecting that they are close not only in space but in meaning. Word embeddings are specially useful for developing prediction models, as they convey a representative numerical view of words, an important requirement for most of the machine learning techniques used as part of IR applications [186].

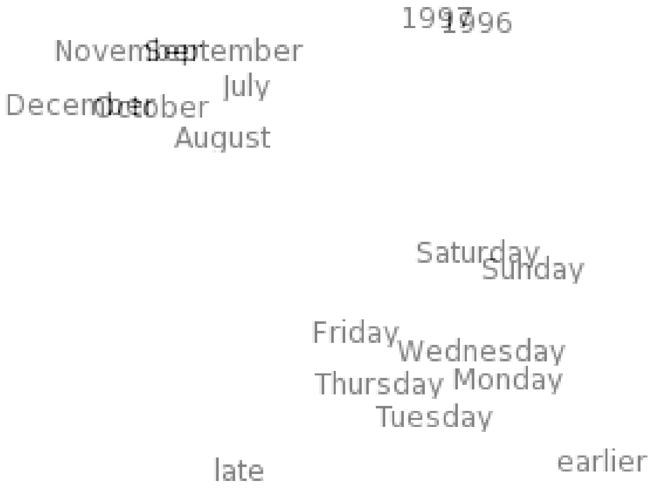


Figure 2.1: 2-dimensional representation of monolingual embeddings generated by Turian et al. [239], which capture groupings of days of the week and months.

Multiple methods have been designed for generating monolingual word embeddings [189, 201]. However, traditional embedding generation methods are constrained to individual languages, due to the fact that embeddings trained for different languages with a monolingual strategy do not share the same space and therefore are not comparable. Being able to generate cross-lingual word embeddings that share the same space independently of the language would make possible to relate words across languages (see Figure 2.2) and simplify the process of designing IR-centric applications that can work across different languages. In the remainder of this section, we describe recent advances in the area of cross-lingual word embeddings. We categorize existing strategies into *monolingual mapping*, *input-modification*, and *cross-lingual training* based strategies.

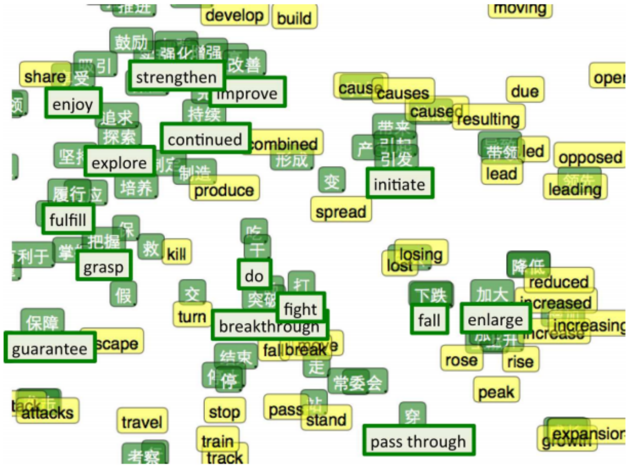


Figure 2.2: 2-dimensional visualization of Chinese and English embeddings in a shared space. Yellow represents English, green represents Chinese, and translations from Chinese to English are represented with a green square [273].

Monolingual Mapping

As stated by Mikolov et al. [188], relative positions among words within a language tend to hold across languages. This phenomena is illustrated in Figure 2.3, where *numbers* and *animal* names in Spanish and English are represented by their 2-dimensional embeddings, following a similar spatial organization. Several researchers exploit this phenomena for finding a mapping function across language spaces, so that both spaces can easily be transformed into the other and used in multilingual systems.

Monolingual mapping strategies rely on monolingual embeddings trained without considering any cross-lingual information. Patterns found in these monolingual embeddings are used to generate a transformation matrix capable of mapping word representations from their respective language spaces into a shared space. For training the aforementioned transformation matrix, it is common to take advantage of word pairs that are extracted from a translation dictionary or a word aligned parallel corpora. The most popular strategy used for training each monolingual space is *word2vec*[4] [189] in its two variants: skip-gram with negative sampling and continuous bag of words [211].

One of the first strategies for monolingual mapping was proposed by Mikolov et al. [188], who calculate a transformation matrix that goes from a source language into a target language. This is done by minimizing the distance between a word in the target language representation and a word in the source language representation transformed by the matrix, using Equation 2.7.

$$\min_{W} \sum_{i} ||Wx_i - z_i||^2 \tag{2.7}$$

where W is the transformation matrix and x_i and z_i are the embeddings of the i^{th} word pair in a bilingual dictionary.

Several researchers extend the technique introduced by Mikolov et al. [188] by amending limitations in its loss function. Xing et al. [259] find that using different functions for (1) learning word representations, (2) measuring the distance between them, and (3) calculating the transformation matrix is inconsistent and leads the embedding training process to underperform. To solve this, the authors instead use cosine similarity for all the aforementioned functions. Similarly, Lazaridou et al. [153] argue that using least-squares as loss function leads the embedding generation process to create hubs, i.e., words that appear in the surrounding area of too many other words. Consequently, they introduce the

[4] Word2vec is a popular monolingual embedding strategy that considers that words that occurred next to each other are more semantically related than distant words. The strategy uses an sliding window technique, where a neural network is asked to predict a given word w based on its context formed by n words preceding and following w. See the work of Mikolov et al. [189] for more detailed information.

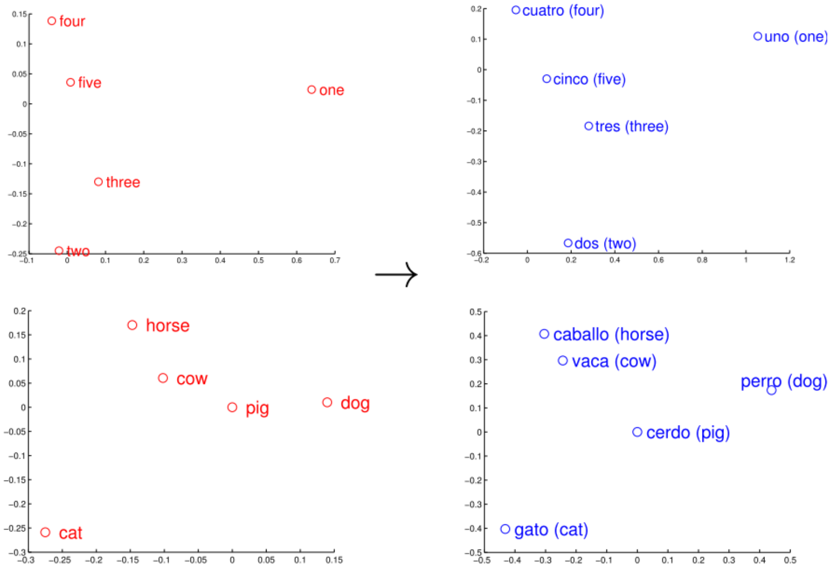


Figure 2.3: A 2-dimensional representation of numbers and animals in English and Spanish by Mikolov et al. [188]

.

concept of intruders, i.e., words that are near the projection of another word, but far from the real translation of it. The authors replace randomly selected negative samples used by Mikolov et al. [188] with intruders in order to provide the system with more informative samples.

For learning a mapping among the embeddings in each language, Upadhyay et al. [242] propose an approach based on canonical correlation analysis. Instead of learning one linear transformation that goes from the space of language A to the space of language B, this novel strategy learns one transformation per language that goes from the original language space to a new shared space. Ammar et al. [9] extend this framework in order to consider more than two languages.

Guo et al. [110] generate a mapping by simply using the alignments of words in a parallel corpus. For each word w in a source language, they gather the words (and their frequencies) in the target language that w is aligned with. Thereafter, w is assigned the average of all the aligned embeddings in the target language, weighted by their alignment frequency. One of the critiques for this method, however, is that it only generates embeddings for words that have an alignment, which is a constraint given the limited amount of parallel corpora. To partially bypass this issue, Guo et al. [110] take advantage of the edit distance, which allows the system to also capture words with similar roots.

Artetxe et al. [14] generalize previous works on embedding generation based on monolingual mapping by starting with a simple learning loss function, and incrementally adding more constraints to the function, including orthogonalization or normalization of the mapping matrix. An interesting fact about this work is that in adding these constraints, they obtain both the loss functions proposed by Xing et al. [259] and the one proposed by Upadhyay et al. [242], demonstrating an actual relation between both solutions.

Strategies based on monolingual mapping require less time for training than other strategies, due to the fact that monolingual embeddings are already trained and the only parameters to be learned are the ones of the mapping [188, 14]. Despite their fast training times, strategies based on monolingual mapping are highly constrained by the quality of the monolingual embeddings and the availability of word alignments in order to estimate an accurate mapping function.

Strategies based on Input Modification

Strategies based on input modification aim to modify the training data, to make it possible to directly use learning techniques designed for monolingual embeddings, such as Continuous

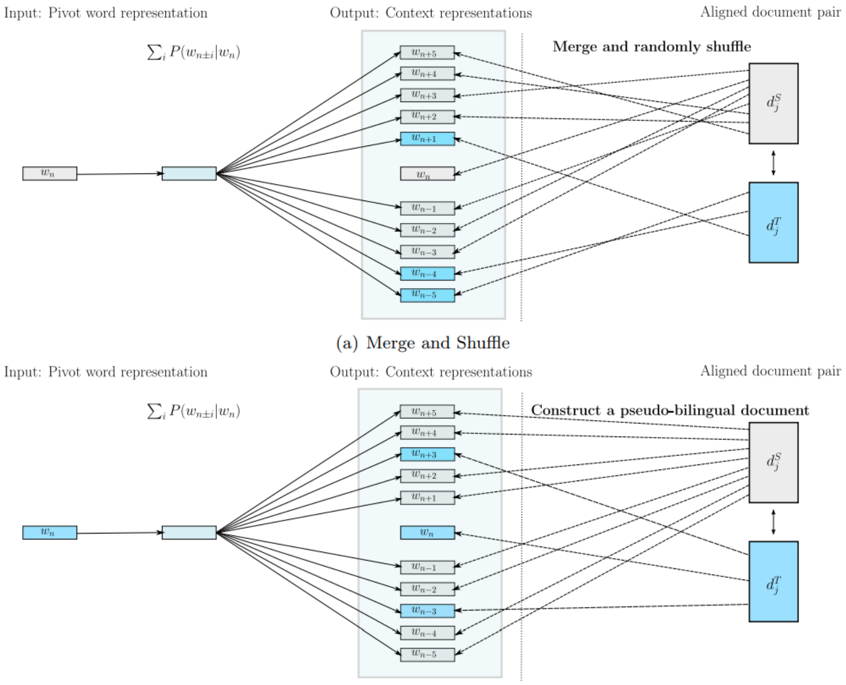
Bag of Words (CBOW) or the skip-gram model proposed by Mikolov et al. [189], to learn cross-lingual ones. These data modifications are mostly based on replacements of words or merging several monolingual corpora into one.

Xiao and Guo [258] take advantage of two monolingual corpora and translations pairs obtained from Wiktionary.org. They create a multilingual dictionary based on words from both corpora and their translations. Each word and its corresponding translation is used as a single entry in this dictionary and is therefore treated as a single embedding vector. Both monolingual corpora (modified using the new dictionary) are fed to a monolingual training strategy to generate the multilingual embeddings.

Duong et al. [69] propose a similar strategy to the one presented by Xiao and Guo [258] that extracts translation pairs from the Panlex.org dictionary. This strategy does not force word translation pairs to point to the same word embedding, but alternatively replaces words with their translation, expecting the monolingual embedding learning strategy to learn a similar vector to its translation based on the surrounding words. The proposed strategy explicitly handles polysemy by using an expectation maximization model.

Instead of modifying the input fed to the monolingual strategy on-the-fly, Gouws and Søgaard [103] create a new pseudo bilingual corpus that is fed to the popular CBOW [189] monolingual strategy. The new corpus is based on the concatenation of two monolingual corpora, in which words are replaced with their translated counterparts with 50% probability.

Ammar et al. [9] propose a similar approach to Gouws and Søgaard [103] that rather than replacing words with its translations, replaces them with unique identifiers. These identifiers are common for a word in the source language, a word in the target language, and their respective synonyms. This strategy forces each word and its translation to point to exactly the same vector, in a similar way to the approach proposed by Xiao and Guo [258].


Input: Pivot word representation
Output: Context representations
Aligned document pair
$\sum_i P(w_{n\pm i}|w_n)$
Merge and randomly shuffle
w_{n+5}
w_{n+4}
w_{n+3}
w_{n+2}
w_{n+1}
w_n
w_{n-1}
w_{n-2}
w_{n-3}
w_{n-4}
w_{n-5}
d_j^S
d_j^T
Construct a pseudo-bilingual document

(a) Merge and Shuffle

(b) Length-Ratio Shuffle

Figure 2.4: Strategies for merging documents: Random shuffle and length-ratio shuffle, which evenly intercalates words from each document [248].

The strategy presented by Vulić and Moens [248] also merges two monolingual corpora into one. However, instead of simply concatenating the two corpora, they shuffle words from each document in an alternate way (see Figure 2.4). For this strategy to work, they use a document-aligned multilingual corpus, arguing that the structure of the documents will be similar enough so that the words appear in similar contexts to their translated counterparts.

Input modification techniques are in general trivial for implementation, as they benefit from already existing monolingual embedding generation tools, making them useful for quick experiments. However, they are limited by the amount of parallel data available, which for some languages is scarce.

Cross-lingual Training

Unlike the strategies discussed in Section 2.2.2, which modify data and feed it to a non-cross-lingual strategy, the strategies discussed below are specifically designed for training cross-lingual representations and therefore directly or indirectly consider both monolingual and cross-lingual properties. For this reason, the objective functions most commonly found in these type of strategies are composed of a monolingual and a cross-lingual function [103]. The monolingual function measures how accurate the word representations are in each of the languages considered, while the cross-lingual one quantifies how well embedding properties translate across languages.

Hermann and Blunsom [115] present CVM, a model for generating bilingual embeddings that uses a sentence aligned corpus for training purposes. For each sentence in a bilingual sentence pair, CVM generates a sentence embedding that consists of the sum of the embeddings of all the words in them. CVM aims to minimize the Euclidean distance between pairs of sentence embeddings, subsequently creating multilingual word embeddings. Hermann and Blunsom [116] introduce DocCVM, which enhances CVM so that it can also consider document aligned corpora arguing that sentence aligned corpora is limited. This new model combines the sentence embeddings generated with CVM to create a document level representation, the distance of which is minimized across languages (see Figure 2.5). Kočiskỳ et al. [144] also use a document-aligned corpora in a two step system that first learns automatically the word alignments across documents, and then generates word embeddings based on those word alignments.

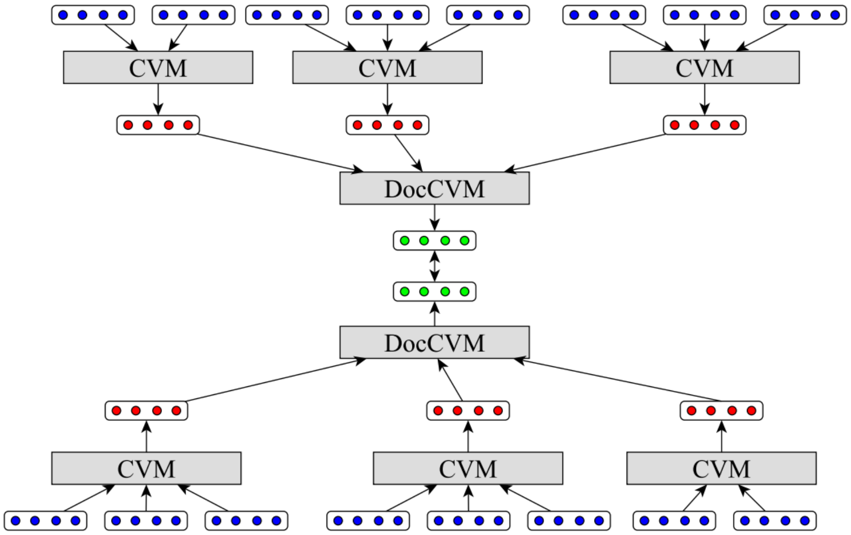


Figure 2.5: Compositional document model for bilingual embedding generation. The structure described the computation graph for two documents with three sentences, each composed of 3 words [116].

Several researchers have used dimensionality reduction techniques to generate cross-lingual embeddings. Huang et al. [122] extend the popular Latent Semantic Analysis algorithm to make it translation invariant. Søgaard et al. [224] use documents in Wikipedia to generate a term-document matrix, i.e., assigning a binary vector to each word that indicated whether the word appears in each document. The dimensionality of this matrix is later reduced to generate fixed-size real-valued embedding vectors for representing the meaning and relations across words. Zou et al. [273] propose to use a term-term vector instead, where the vector assigned to each word in a source language represents the co-occurrence frequencies with each word in a target language calculated from a parallel corpora. Thereafter, the authors apply matrix factorization [154] for reducing the dimensionality of the vectors. In a similar manner, Shi et al. [221] consider a word co-occurrence matrix for generating word representations. However, their model also uses additional monolingual data by considering co-occurrence matrices within the language.

Lauly et al. [152] take advantage of an auto-encoder for generating embeddings across languages. They use sentence pairs of a parallel corpora as inputs and outputs for the auto-encoder, minimizing four different reconstruction errors: (1) source to source language, (2) target to target, (3) source to target and (4) target to source. The authors propose an auto-encoder approach based on a tree-shaped encoding-decoding structure, which they modify in a posterior work by using a sparse binary vector for representing each word in a sentence, improving the quality of their embeddings [215]. Figure 2.6 illustrates the behavior of both structures when reconstructing the sentence *the dog barked* from French.

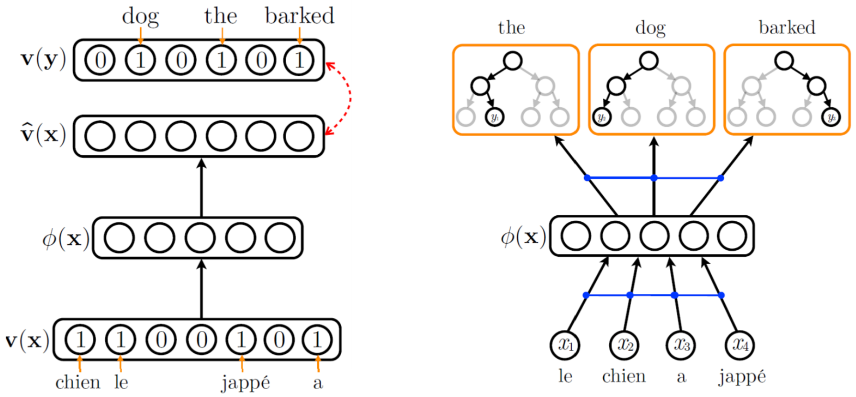


Figure 2.6: Left: Bilingual autoencoder based on the binary reconstruction error. Right: Tree-based bilingual autoencoder. In this example, both structures reconstruct the bag-of-words for the English sentence “the dog barked” from its French translation “le chien a jappe” [215].

The popularity of word2vec models presented by Mikolov et al. [189] has lead researchers to try to adapt these techniques for cross-lingual embedding training. Luong et al. [167] adapt the skip-gram model by including words from a target language as context words. Instead of predicting only words that appear close to a given word, their system

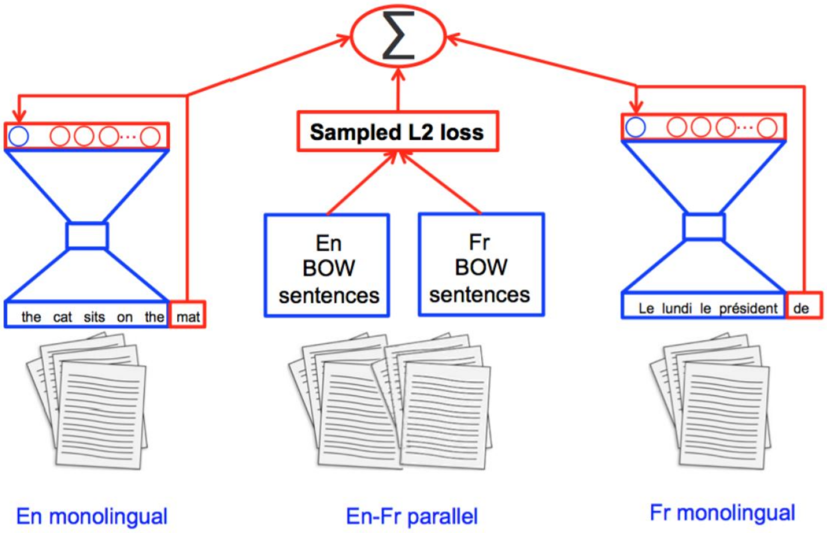


Figure 2.7: Visual description of the model presented by Gouws et al. [104] for cross-lingual embedding generation. The loss function represented as $\sum$ combines three different values: the loss function for French, the loss function for English, and the cross-lingual loss function.

also predicts words that appear close to the translation of the word. The authors examined several word-alignments methods concluding that for the two languages they consider (German and English), the performance of a simple monotonic alignment, i.e., each word is aligned with the word in the same position, is comparable among the performance of other methods. Coulmance et al. [49] also extend the skip-gram model. However, rather than combining word contexts by aligning words, they consider all words in a translation sentence pair to be context words of each other. This coincides with the conclusion of Luong et al. [167] who demonstrated that a precise alignment is not as important for this task. Rather than combining context across languages together, Gouws et al. [104] combine two separate objective functions: the traditional skip-gram objective function for each language and a novel cross-lingual regularization function, which minimizes the distance among the representations of all pairs of words among sentence pairs.

Overall, strategies based on cross-lingual training offer the best adaptability to a specific problem, given that they can work with all types of corpora, from simple word alignments to document aligned data. These strategies, however, are more laborious to implement than monolingual-mapping and input modification and require to minimize multiple objective functions making them slower to train.

In this section we have introduced three different strategies for training cross-lingual embeddings based on monolingual-mapping, input modification and cross-lingual training. Each of these techniques has different benefits and drawbacks, in terms of ease of implementation, needs of corpora, and performance. Having said that, they are useful when it comes to designing and developing a variety of IR applications, which we discuss in Section 2.2.3. For more in-depth information on cross-lingual embedding models, refer to the survey created by Ruder [211].

2.2.3 Multilingual Information Retrieval Applications

We discuss below different multilingual IR applications. We limit our discussion to applications in three popular IR areas: Recommendation, Search, and Question Answering.

Multilingual Recommendation

A recommendation system aims at identifying items that are of interest to a user given some historic data [210]. The recommendation process is usually based on generating both item and user representations that are compared to determine the degree to which an item is appealing to a user. Recommendations are generated using two main strategies:

collaborative filtering and *content based*. Strategies based on collaborative filtering are inherently cross-lingual, as they do not consider text in its computation, instead they analyze rating patterns across users. However, content based strategies require analyzing text for understanding the users' preferences and provide recommendations, and thus, need to be adapted for multilingual environments. In the rest of this section, we describe several techniques that can be applied in a multilingual environment.

In order to provide news recommendations across users that understand both English and Bengali, Ferdous and Ali [82] propose an strategy that takes advantage of an ontology (shown in Figure 2.8) specifically created for the news domain. This ontology captures the most important facts in a news article, such as when the action took place, what happened or who took part on the action. Given articles historically read by the user, and a collection of articles to retrieve, a representation of each news article is created based on the proposed ontology, which is later used by the system to match relevant news articles. The degree of similarity between the user profile based on read articles and potential articles to be recommended is measured using a function that considers (1) the number of words matched across the ontology of each article, where the words in Bengali are previously translated to English, and (2) the ratio of classes shared across the two ontologies.

Yang et al. [264] generates cross-lingual recommendations of Google news groups across Chinese and English users, where news groups are defined as a set of news articles that share a common topic. After a preprocessing step that considers tokenization, stop-word removal, stemming and boundary detection (specific for Chinese), each news article is represented as a term frequency vector. These term vectors, however, are not language independent given that words are not shared across both languages. To amend this, the

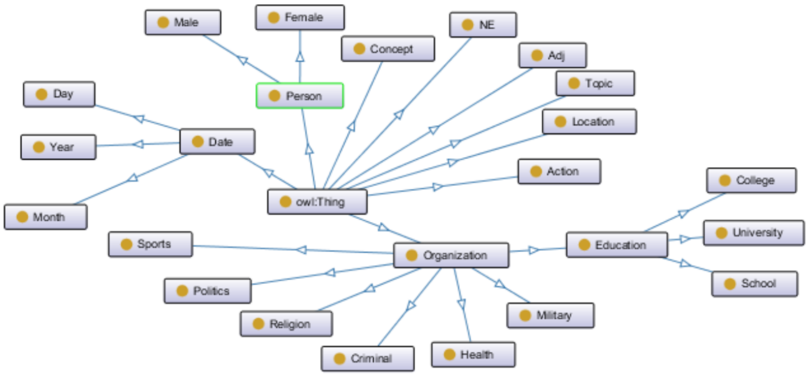


Figure 2.8: Ontology used for representing the content of a news article [82].

authors use a bilingual dictionary to translate each word from Chinese to English or vice versa, depending on the direction of the recommendations. After creating term frequency vectors for all news articles, each news group can be seen as a cluster of multidimensional points (as shown in Figure 2.9), where each point represents a news article. Lastly, a supervised learning approach that measures to what extent do two news group clusters overlap is applied in order to recommend the most relevant (overlapping) ones.

Unlike the strategy proposed by Yang et al. [264], the one introduced by Magnini and Strapparava [175] takes advantage of MultiWordnet, a lexical database that defines semantic relations including synonymy, hyponymy, or hyperonymy across word senses. One useful feature of MultiWordnet is that each word sense (referred as synset) has a unique identifier that is language independent. Magnini and Strapparava [175] use this feature to build language independent user profiles based on a graph of synsets, a sample of which is illustrated in Figure 2.10. These graphs are compared to synsets obtained from news articles to measure their relevance and recommend them to the corresponding users, regardless of the language of the articles.

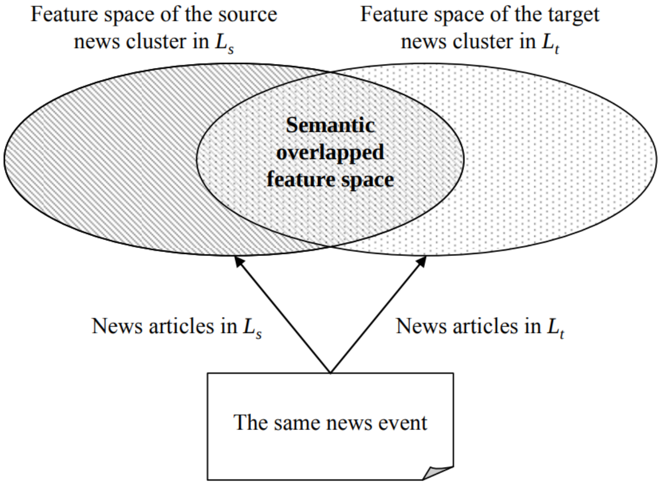

Feature space of the source news cluster in L_s
Feature space of the target news cluster in L_t
Semantic overlapped feature space
News articles in L_s
News articles in L_t
The same news event

Figure 2.9: Representation of two news group clusters and the semantic overlap between them , where L_s and L_t represent the source and target language respectively [264].

Recommendation of scientific papers can also be enhanced by a multilingual strategy as Uchiyama et al. [240] demonstrated with OSUSUME, a system that recommends scientific papers in English given Japanese user-defined keywords. OSUSUME uses the input keywords provided by the user to retrieve intermediary Japanese papers containing them. Thereafter, the abstracts of these papers are translated into English and keywords are extracted from them. Finally the English keywords are used by OSUSUME to retrieve scientific papers in English to be recommended to the user. In the same domain, Lai and Zeng [150] propose a cross-lingual paper recommendation for digital libraries that work in Chinese and English. Their strategy considers implicit feedback obtained from users' interaction with the digital library, including the search queries or the documents downloaded for reading. This information is aggregated to create a user profile that consists of a vector of frequency weighted terms, such as the one shown in Figure 2.11, which showcases a representation of a document mostly focused on information retrieval. The

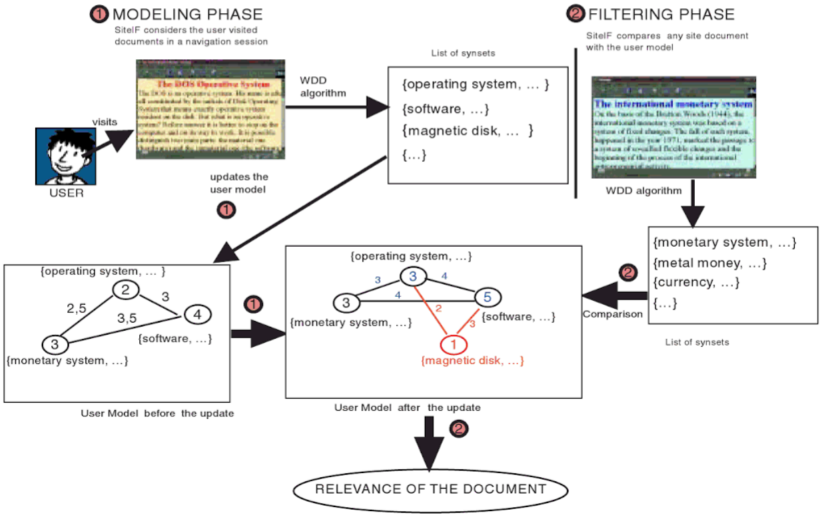


Figure 2.10: Description of the algorithm used by Magnini and Strapparava [175]. A user profile is built using a graph of synsets, which is compared to synsets obtained from news articles to be recommended.

vector representing the user profile is compared with vectors created in a similar way for each document in the digital library. The most similar ones are recommended to the user using cosine similarity as relevance function. In order to address the language gap in the aforementioned vectors, Lai and Zeng [150] use a machine translation tool and translate all user profiles to both English and Chinese.

$$V_c(A002)=\{(\text{信息检索},0.45),(\text{协同过滤},0.3),(\text{文本挖掘},0.15)\}$$

$$\begin{aligned}V_f(A002)=\{&(Information\ \mathrm{Re}trieval,0.45),(Collaborative\ Filtering,0.3),\\&(Text\ Mining,0.15)\}\end{aligned}$$

Figure 2.11: User profile defined by a vector of weighted terms , where V_c and V_f are the vector representations in native and target language respectively Lai and Zeng [150].

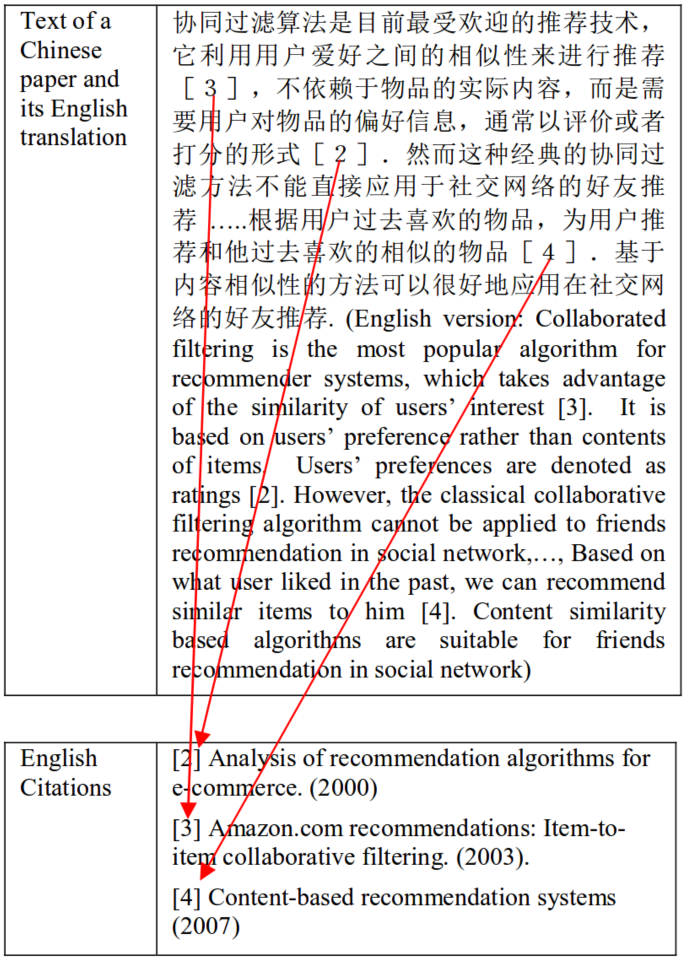
A novel task in the area of paper recommendation was proposed by Tang et al. [235]: context aware cross-lingual citation recommendation. For a given text context in a scientific paper in Chinese, the proposed system ranks and recommends English citations suitable for that context. The solution goes beyond simply translating the papers, as Uchiyama et al. [240] and Lai and Zeng [150] did, since it defines two linear functions for mapping the textual content from both the citation context and the citation abstract to a lower dimensional embedding space that is shared across languages. These two embeddings are compared using Equation 2.8 to measure the relevance of each citation given the context.

$$f(q, d) = q^T W (d^T F)^T \tag{2.8}$$

where q and d are the TF-IDF[5] vectors of the context and the abstract of the candidate citation, and W and F are two linear mapping matrices learned using a supervised training process.

Takasu [232] presents a strategy for cross-lingual keyword recommendation based on latent topic analysis. Similar to he work of Tang et al. [235], the strategy in Takasu [232] does not require explicit translation of documents as it rather converts both documents and keywords into a single feature space shared across languages. This feature space is defined as a probability distribution of latent topics obtained using an ad-hoc extension of Latent Dirchllet Allocation (LDA) that considers multiple languages. This extended LDA model is trained over a document aligned parallel corpus, assuming that parallel documents follow a similar distribution of topics. Takasu [232] evaluates the strategy by recommending keywords across English and Japanese, concluding that the obtained accuracy is comparable to the one obtained in a monolingual environment.

[5] TF-IDF is a word relevance weighting strategy that considers a word to be more important to a document the more frequent it is on it, and the less frequent it is on a general corpora [3].



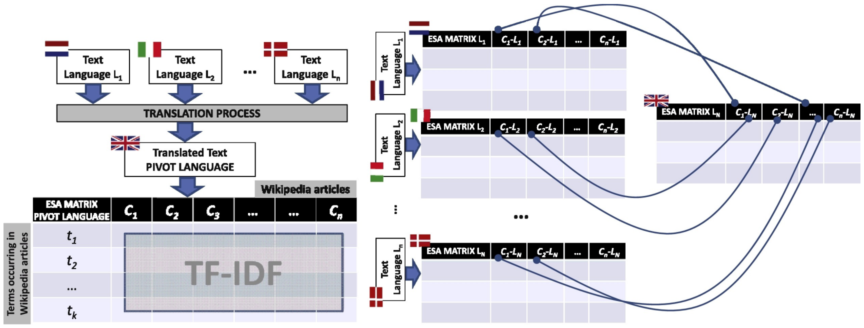
Text of a Chinese paper and its English translation	协同过滤算法是目前最受欢迎的推荐技术，它利用用户爱好之间的相似性来进行推荐［3］，不依赖于物品的实际内容，而是需要用户对物品的偏好信息，通常以评价或者打分的形式［2］．然而这种经典的协同过滤方法不能直接应用于社交网络的好友推荐.....根据用户过去喜欢的物品，为用户推荐和他过去喜欢的相似的物品［4］．基于内容相似性的方法可以很好地应用在社交网络的好友推荐. (English version: Collaborated filtering is the most popular algorithm for recommender systems, which takes advantage of the similarity of users’ interest [3]. It is based on users’ preference rather than contents of items. Users’ preferences are denoted as ratings [2]. However, the classical collaborative filtering algorithm cannot be applied to friends recommendation in social network,…, Based on what user liked in the past, we can recommend similar items to him [4]. Content similarity based algorithms are suitable for friends recommendation in social network)

English Citations	[2] Analysis of recommendation algorithms for e-commerce. (2000) [3] Amazon.com recommendations: Item-to-item collaborative filtering. (2003). [4] Content-based recommendation systems (2007)

Figure 2.12: Description of the cross-lingual context-aware citation recommendation task addressed by Tang et al. [235].

Education is a domain that can also benefit from multilingual recommendations. Schmidt et al. [217] present a cross-lingual recommendation strategy for CROCODIL, a on-line platform for supporting resource based learning in German and English. The proposed strategy extends the Explicit Semantic Analysis (ESA) model by considering Wikipedia’s interlingual links to make it functional in a cross-lingual environment. ESA assumes that the meaning of a word can be represented with a multidimensional vector, in a similar way to the word embedding strategies we describe in Section 2.2.2. However, unlike embeddings for which the meaning of each dimension is unknown, in ESA the meaning of each dimension is explicitly described. Schmidt et al. [217] consider a word to have as many dimensions as documents in Wikipedia, where each dimension represents the number of times the word appeared in the corresponding document divided by the number of words on it. This representation, however, is still language dependent as Wikipedia has different documents for each language. To amend the language dependency issue, the authors take advantage of interlingual links in Wikipedia, that define relations between the same articles across languages. Based on these links, Schmidt et al. [217] define a new dimensional space, where each dimension represents not only one document but all the documents in different languages that relate to the same topic. Using this multidimensional space, cosine similarity is applied as a distance function across words to generate recommendations. Similarly, Narducci et al. [193] leverage the ESA algorithm for content based item recommendation across Italian and English. To recommend movies and books (as illustrated in Figure 2.13), the authors consider two strategies: translation based ESA, which translates all documents to a pivot language (English in this case) prior to creating the ESA matrix, and cross-lingual ESA, which takes advantage of Wikipedia's interlingual links to create a common multidimensional space. Regardless of the ESA strategy employed, a vector representation is computed based on the description of each item. More specifically, the vector representation of each item is defined as the centroid of the ESA vectors of all the words on its description. Item representation are compared using cosine similarity.

Similar to Magnini and Strapparava [175], Lops et al. [164] take advantage of Multi-



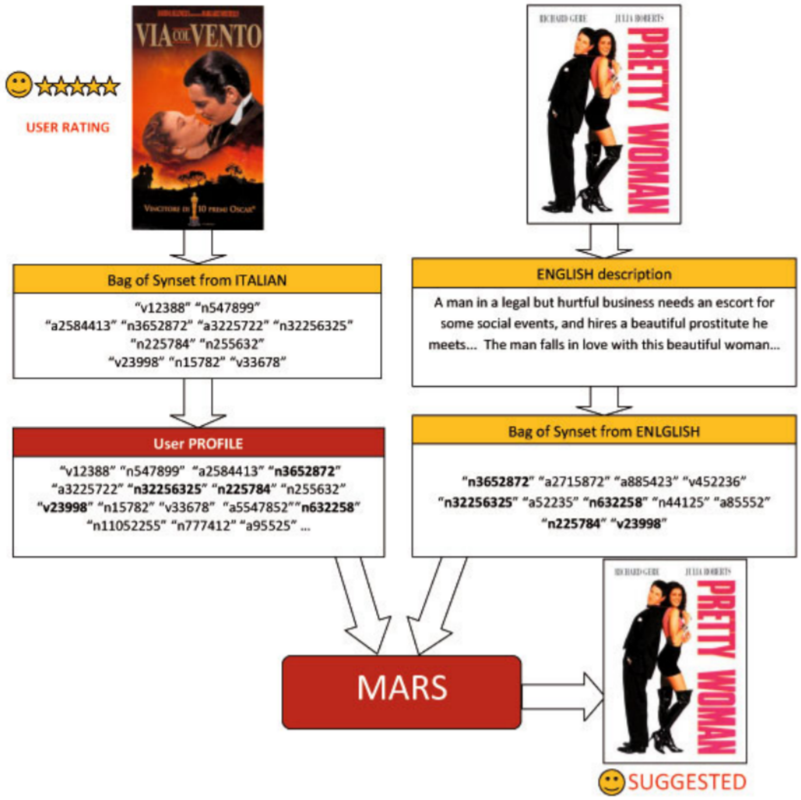
(a) Translation-based ESA (b) Cross-language ESA

Figure 2.13: Two strategies of extending ESA for cross-lingual applications as introduced by Narducci et al. [193]. Translation based ESA translates the documents to a pivot language before creating the matrix, while cross-lingual ESA takes advantage of Wikipedia's interlingual links to create a language independent space.

Wordnet for building a multilingual movie recommendation system. Instead of using a synset graph as Magnini and Strapparava [175] proposed, Lops et al. [164] generate a bag of synsets for each movie description, as illustrated in Figure 2.14, that works in a similar way to a bag of words, with the benefit of being language independent.

Musto et al. [190] compare two cross-lingual strategies for movie recommendation. The first one, originally presented in [164], takes advantage of MultiWordnet and a bag of synset model for comparing cross-lingual documents. The second strategy is based on an multilingual extension of Random Indexing [132], a traditionally monolingual embedding generation technique. For each movie description and its translations, the authors generate a document representation that is the result of aggregating the embedding vectors of all words on the document. This creates a multilingual vector representation for each movie that can be used for measuring similarity with respect to other movies, in order to recommend the most alike.

Instead of generating content-based cross-lingual recommendations, Komiya et al.


VIA COL VENTO
Vincitore di 10 premi Oscar
USER RATING
RICHARD GERE
JULIA ROBERTS
PRETTY WOMAN
Bag of Synset from ITALIAN
"v12388" "n547899"
"a2584413" "n3652872" "a3225722" "n32256325"
"n225784" "n255632"
"v23998" "n15782" "v33678"
ENGLISH description
User PROFILE
"v12388" "n547899" "a2584413" "n3652872"
"a3225722" "n32256325" "n225784" "n255632"
"v23998" "n15782" "v33678" "a5547852""n632258"
"n11052255" "n777412" "a95525" ...
Bag of Synset from ENLGLISH
"n3652872" "a2715872" "a885423" "v452236"
"n32256325" "a52235" "n632258" "n44125" "a85552"
"n225784" "v23998"
MARS
SUGGESTED

A man in a legal but hurtful business needs an escort for some social events, and hires a beautiful prostitute he meets... The man falls in love with this beautiful woman...

Figure 2.14: Generating movie recommendations using the bag of synsets model presented by Lops et al. [164].

[147] address the problem of cross-lingual collaborative filtering based recommendation. They present a task where two e-commerce websites (one in English and one in Japanese) have worked independently in the past and are currently being merged. The only textual information shared across the two websites is the name of the product. Pure collaborative filtering techniques would not work for this problem, as there are two completely disjoint communities, meaning that items of the first community would never be recommended to users of the second community and vice versa. For addressing this issue, the authors propose to translate the product names in order to group products across both communities.

This grouping generates products that appear in both communities enabling traditional collaborative filtering strategies to generate recommendations across the two languages/-communities.

In this section, we discussed strategies for building multilingual recommendation systems. While most strategies are based on automatic machine translation, others rely on ontologies or lexical databases for addressing the multilingualism. Strategies that take advantage of dimensionality reduction techniques, such as word embeddings, are currently sparse, however, grounded on the history of other areas, such as search, we believe those strategies will get more popular in the near future. In addition, current multilingual recommendations systems are mostly oriented to recommending items with long textual content. We expect systems that deal with items with shorter textual content, such as tweets or movies to be the focus of future research in this area.

Multilingual Search

Given a collection of documents D and an input query q, a search (or retrieval) task is defined as the act of finding a list of n documents from D that are the most relevant to q. Relevance is defined by the information needs of the user typing the query. While search and retrieval have long be studied from a monolingual perceptive [20], strategies that can retrieve documents in multiple languages are scarce. Multilingual retrieval differs from monolingual retrieval in that the language of q and documents in D can be different. In the rest of this section, we describe existing strategies for building multilingual retrieval systems.

The retrieval strategy presented by Salton [213] translates queries by replacing words with their corresponding translations obtained from a English-Spanish dictionary previous to the retrieval phase. Ballesteros and Croft [22], however, empirically demonstrate that word-by-word translation is not enough and propose a new solution that considers phrases obtained from a parallel corpora for translation. Local translation methods, such as word-by-word or phrase translation, might not be enough for multilingual retrieval [94]. Therefore, Franz et al. [94] go beyond phrasal translation and consider a full automatic machine translation system for translating a query. This approach, however, requires large amounts of parallel corpora, which is hard to obtain. To amend this issue, the authors propose a strategy to generate a comparable corpora, i.e., pairs of documents that are expected to have similar contents but are not manually aligned, which is used to improve the machine translation system. This strategy treats as comparable, those documents that retrieve each other in a search engine, where the document matching is done via translation word pairs.

A similar approach to the one in [94] is taken by Nie et al. [195], as they also use a machine translation tool for translating queries but their main focus is on how to generate a parallel corpus. Nie et al. [195] argue that the Internet contains a vast amount of multilingual resources that can be exploited for training machine translation tools. Their strategy considers the documents retrieved by Altavista (a popular search engine at the time) in response to queries that denote a multilingual website. These queries include phrases such as *"version anglaise"* or *"in French"*, in an attempt to search for anchor points oriented to change the language of a web page. Thereafter the authors use the HTML structure of the retrieved pages to extract parallel pieces of text.

Arguing that the errors produced by automatic machine translation hinder the performance of cross-lingual retrieval systems, Gollins and Sanderson [99] propose a strategy to reduce the error of the machine translation by merging different translation results. As illustrated in Figure 2.15, their system translates the input query to multiple pivot languages, obtaining intermediary results which are translated subsequently to the target language. This pivot strategy creates multiple translation candidates that are merged to perform the retrieval.

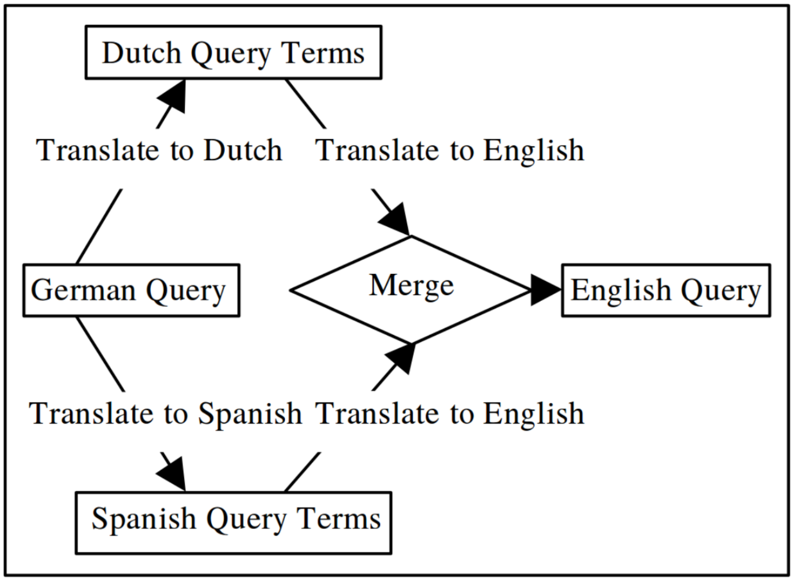


Figure 2.15: Query translation strategy proposed by Gollins and Sanderson [99]. The input query is first translated to two intermediary languages (Spanish and Dutch) and later translated to English.

Instead of translating queries, Oard and Hackett [196] translates available documents so that monolingual retrieval strategies can be used when the user submits a query. While this approach has no time penalty on retrieval time, it requires translating large amounts of document which can be time consuming.

A comparison across query-translation and document-translation, two popular strategies for cross-lingual retrieval based on machine translation is conducted by McCarley [181].

The authors find no statistical different among both strategies, so instead, they propose an hybrid strategy that aggregates relevance scores provided by aforementioned strategies outperforming both previous strategies.

Littman et al. [162] describe an approach based on latent semantic indexing (LSI) that requires no machine translation for cross-lingual document retrieval. LSI is a retrieval strategy that generates a term-document matrix, where each cell represents whether a given term is in a given document. This matrix is reduced using Singular Value Decomposition [100], which generates two matrices: term-latent and latent-document. The rows in the former matrix represent terms while the columns in the latter matrix represent documents in a low dimensional space. This process is illustrated in Figure 2.16. Littman et al. [162] extend the definition of a document in LSI, and consider that a document contains all its words and their translations. This extended representation facilitates training models for multiple languages into a same multidimensional space.

Similar to Landauer et al. [151], Potthast et al. [205] use a vector representations technique for document retrieval: cross-lingual ESA. Each document d is represented with a vector, where each dimension indicates the similarity of d with a specific Wikipedia article. The query follows the same representation and is compared with candidate documents using cosine similarity. Similar to Narducci et al. [193], the cross-linguality is in this case introduced by taking advantage of Wikipedia's interlingual links that facilitate to group articles in different languages. Sorg and Cimiano [225] instead consider the full category[6] structure of Wikipedia (illustrated in Figure 2.17). Sorg and Cimiano [225] create two algorithms: Cat-ESA, which is solely based on categories, and a hybrid named Tree-ESA,

[6] A category is a generalization of multiple Wikipedia articles, e.g., the *monorail* article is under the *rail transport* category, which in turn is under a more general category named *transportation*.

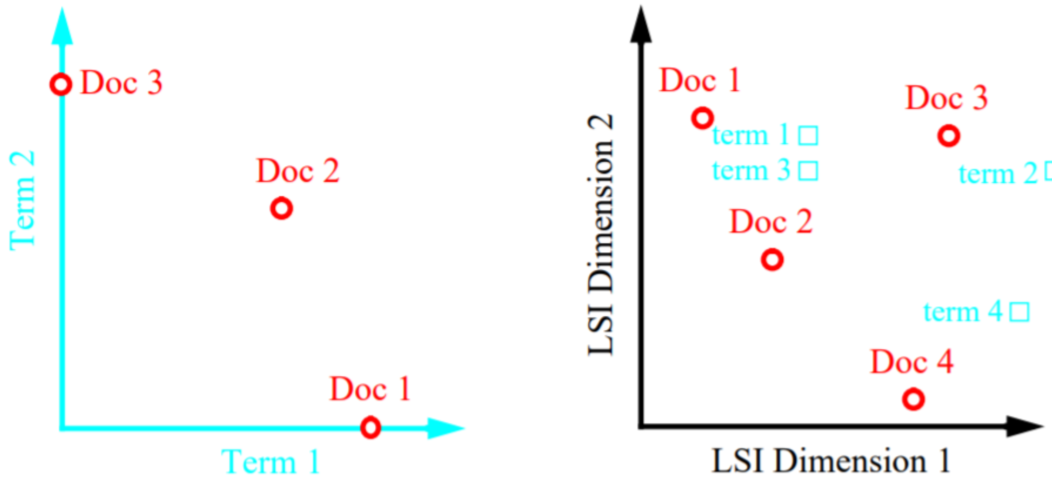


Figure 2.16: Document representations in the standard term space vs a reduced LSI space with both documents and terms represented on it by Littman et al. [162].

which considers the whole structure of Wikipedia including articles and categories.

Vulić et al. [249] argue that parallel corpora[7] and translation dictionaries are a scarce source for certain languages, while comparable [8] corpora is usually abundant. Consequently, they present a cross-lingual search strategy based on Bilingual Latent Dirichlet Allocation that solely relies on comparable corpora. For doing so, their strategy models the probability of a query being generated from the word distribution of a document using cross-lingual topical models.

Cimiano et al. [43] present a comparison of three popular strategies for cross-lingual IR: ESA, LSI, and LDA. They evaluated the three algorithms when trained on Wikipedia and

[7] A parallel corpora is a set of documents that are aligned across languages. In a bilingual parallel corpora, each document in source language has a corresponding document in target language which is an exact translation.

[8] Comparable corpora differs from parallel corpora in that document pairs are only meant to be topically related and do not need to be exact translations.

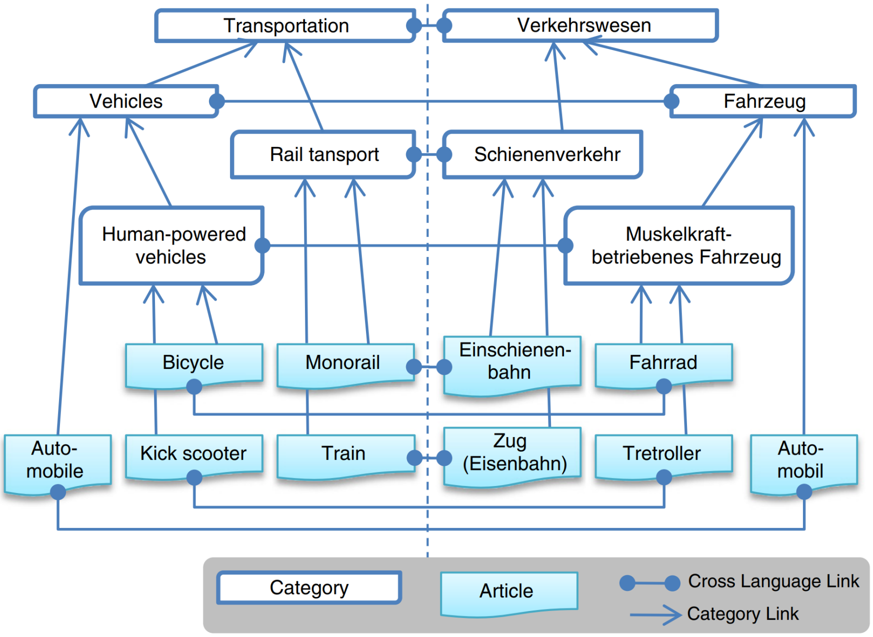


Figure 2.17: Wikipedia articles, categories and link structure exploited for document representations in concept spaces (CL-ESA, Cat-ESA and Tree-Esa) by Sorg and Cimiano [225].

when trained on the database of documents collection that can be retrieved by their system, concluding that ESA outperforms both counterparts when trained on Wikipedia while ESA and LSI provide a similar performance when trained on the document collection.

Similar to Littman et al. [162], Narducci et al. [191] present a cross-lingual strategy that does not translate queries or documents. Instead, it leverages cross-lingual embedding generated using the strategy introduced by Vulić and Moens [248], which we described in Section 2.2.2. Documents are modeled using a compositional approach, where the representation of each document is a weighted sum of the embeddings of all the words appearing on it. The weight is inspired by the IDF normalization factor in TF-IDF, meant to give less importance to words that appear too often and therefore lack information.

Narducci et al. [192] present an strategy for cross-lingually retrieving websites from public administration services, i.e., e-gov sites, addressing the need of foreigners that know which public service they need but do not know its name in the local language (examples shown in Figure 2.18). For doing so, they analyze four different strategies: (1) Wikipedia Miner, an automatic service for cross-tagging Wikipedia articles, (2) Tagme, an automatic semantic annotator of short documents, (3) DBpedia Spotlight, a name entity detection tool, and (4) ESA. The resulting experiments demonstrate that ESA is the one that best performs for this task. Wikipedia Miner, Tagme, and ESA algorithm rely on Wikipedia's interlingual links for cross-linguality, while DBpedia Spotlight relies on the links in the DBpedia knowledge graph.

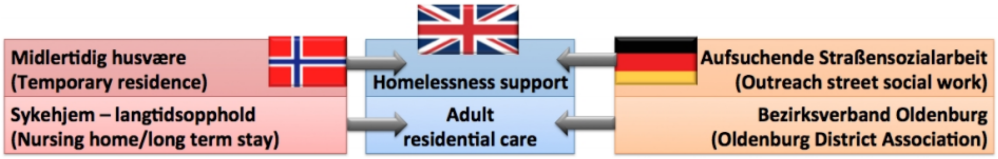

Midlertidig husvære
(Temporary residence)
Sykehjem – langtidsopphold
(Nursing home/long term stay)
Homelessness support
Adult
residential care
Aufsuchende Straßensozialarbeit
(Outreach street social work)
Bezirksverband Oldenburg
(Oldenburg District Association)

Figure 2.18: Examples of e-gov services linked by the algorithm of Narducci et al. [192] along with their translations provided by Bing.

Instead of focusing on the retrieval algorithm of a search engine, Albano et al. [5] focus on how to cluster the retrieved results in order to show the user resources that consider the possible different senses of the input query. For doing so, they present a cross-lingual word sense disambiguation (WSD) strategy that not only represents word senses based on context words, but also considers their translations. As Albano et al. [5] mention, this multilingual context not only allows to compare word senses cross-lingually, but it also improves the accuracy of WSD. The authors use WSD for extracting the senses of snippets of retrieved resources and clustering them given their similarity.

Pham et al. [204] propose a movie search algorithm based on ontology data. The authors take advantage of Linked Open Data that includes information from IMDB and DBpedia related to entities in the movie domain such as actors, films, or directors. Their algorithm conducts a simple matching of words for finding the entities and exploits links across languages, such as the ones illustrated in Figure 2.19, to showcase results in multiple languages.

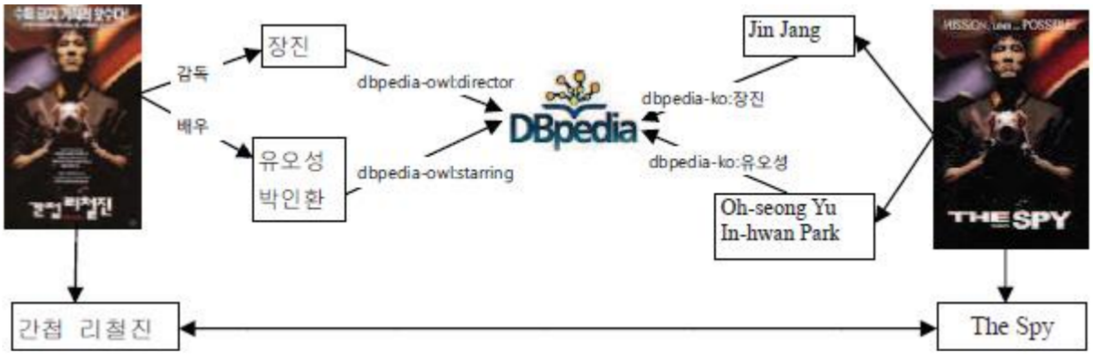


Figure 2.19: Matching Entities in DBpedia across Korean and English by Pham et al. [204].

A learning to rank strategy for cross-lingual text retrieval is introduced by Rahimi and Shakery [207]. For doing so, the authors define several features intended to measure the relevancy of a document given a query, such as the frequency of the query words in the document. A supervised learning strategy is used to train the ranking model based on implicit feedback provided by users in the form of clicked documents. Note that several textual features do not work when comparing documents in two different languages. In order to amend this, the authors take advantage of a word-to-word translation model trained on a sentence-aligned parallel corpus.

In this section, we described prominent strategies for multilingual search and retrieval.

While initially researchers focused on developing strategies based on machine translation, current strategies focus on creating multidimensional representations of the documents that can work in a cross-lingual manner. Strategies for creating document representations are often based on word co-occurrence metrics, whereas some ontology based techniques seem to find their place for specific problems such as movie search. Moreover, it is noticeable that all multilingual approaches oriented to search are only focused on the retrieval algorithm itself, and ignore other important modules including search intent identification query transformation, or query suggestion. We anticipate multilingual works that focus on the aforementioned modules among IR and Natural Language Processing research communities.

Multilingual Question Answering

Given a question q and a document collection D, question answering (QA) aims at finding the most suitable answer(s) to q that satisfy the need of the user that wrote it. The answer to q can be obtained from a question answering community, in which case the QA algorithm would search for a similar question to q and provide an existing user generated answer; or it can be obtained from a general resource collection, where the QA system would typically highlight a passage where the answer is located [244]. Multilingual QA differs from traditional QA in that the language of q and the documents in D can be different. We describe below research efforts in the area of QA in a multilingual environment.

Similar to what happened for multilingual recommendation and search, automatic machine translation has been the starting point for multilingual QA. Whittaker et al. [254] propose to translate a given question to the language that answers are written using automatic machine translation tools in order to treat the task as a monolingual task. Bowden et al. [32] argue that translating short text pieces, as is the case with questions, can cause noise. Therefore, they propose to translate all documents in D instead. This method avoids the delay caused by automatic translation when the user is querying the system, however, it has the disadvantage of having to translate all documents in D to all possible question languages, which in some cases can be inviable.

As stated by Ferrández et al. [84], machine translation is imprecise and using it on a QA system hinders its performance, as shown in the CLEF QA challenge, where multilingual approaches retrieved less than 50% of correct answers retrieved by monolingual approaches [84]. In order to amend this situation, Ferrández et al. [84] propose to use Wikipedia's interlingual link structure to match questions in two different languages. Figure 2.20 illustrates the system proposed by Ferrández et al. [84], where apart from the traditional question analysis and answer extraction module there is a module that takes advantage of Wikipedia's structure to relate questions and answers in different languages. One of the strengths of this strategy is that it can automatically deal with Named Entities (phrases denoting proper person, location or organization names) as those are directly described in Wikipedia (see Figure 2.21). Magnini et al. [176] also rely on Wikipedia's intelingual links for QA, however, their proposed methodology includes several other modules, such as multiword recognition and answer type detection in their system for a more accurate question answering.

A strategy based on word embeddings is presented by Chen et al. [38] for Chinese-English cross-lingual question retrieval. Ford doing so, the authors train monolingual word embeddings for each language based on the skip-gram model proposed by Mikolov et al.

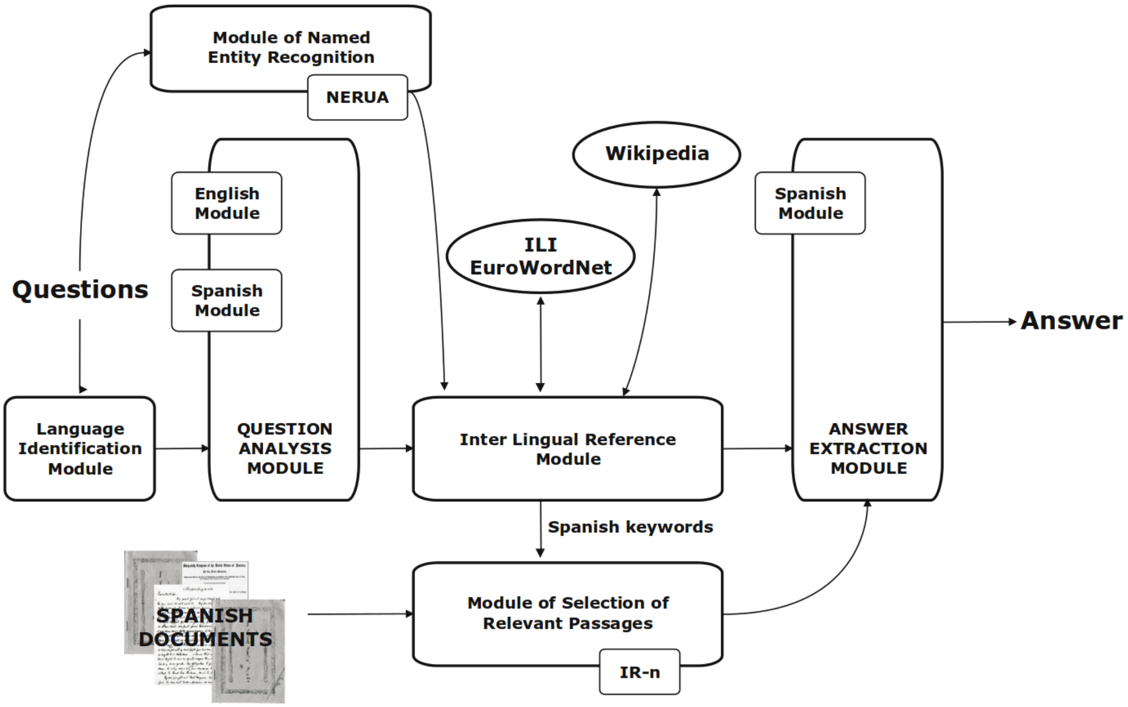


Figure 2.20: Cross-lingual Question Answering strategy proposed by Ferrández et al. [84]. A novel Wikipedia based content matcher co-exists with traditional question analysis and answer extraction modules.

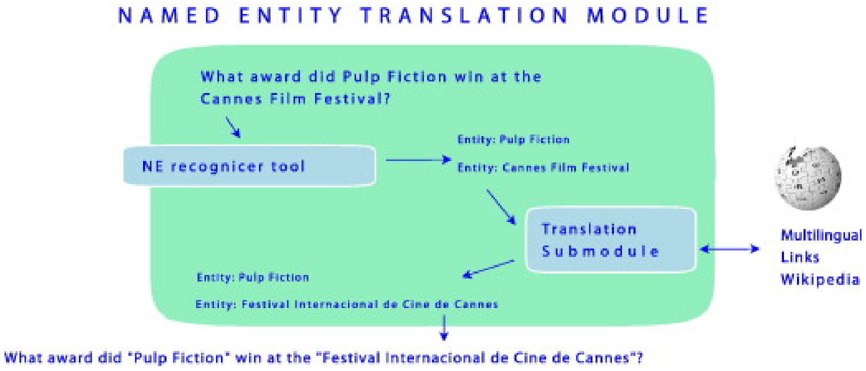

NAMED ENTITY TRANSLATION MODULE
What award did Pulp Fiction win at the Cannes Film Festival?
NE recognicer tool
Entity: Pulp Fiction
Entity: Cannes Film Festival
Translation Submodule
Multilingual Links Wikipedia
Entity: Pulp Fiction
Entity: Festival Internacional de Cine de Cannes
What award did "Pulp Fiction" win at the "Festival Internacional de Cine de Cannes"?

Figure 2.21: Named Entity translation strategy by Ferrández et al. [85].

[189]. Questions represented by the concatenation of word embeddings are fed to two dual convolution neural networks, one for Chinese and one for English. During training, both networks are fed with question pairs that are known to mean the same in Chinese and English, aiming to minimize the distance between the representation generated by both networks. This process creates sentence representations located into a cross-lingual multidimensional space. The authors use the distance between representations to create a ranking of most relevant question-answer pairs.

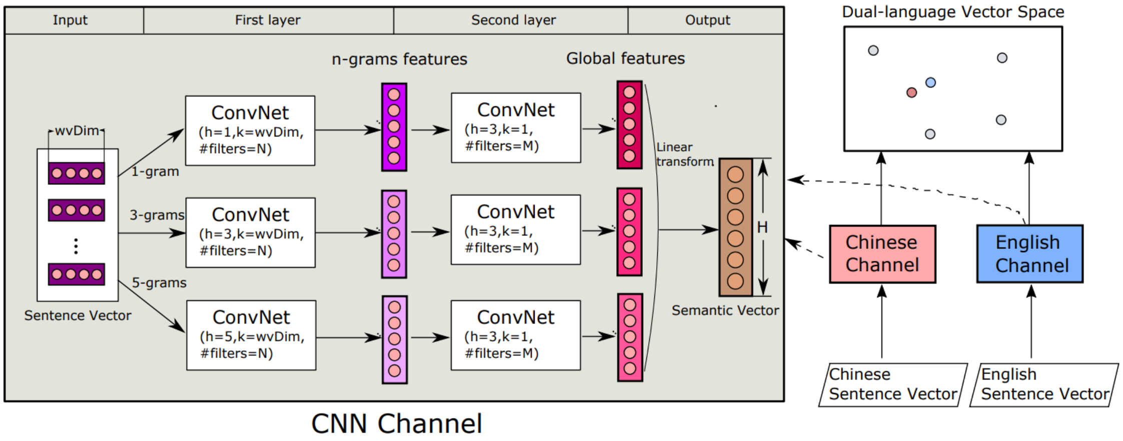


Figure 2.22: QA based on neural networks by Chen et al. [38]. Questions in different languages are mapped to a dual vector space for comparison.

Da San Martino et al. [50] propose a technique for cross-lingual question retrieval that relies on a neural network that computes the similarity between two questions. This neural network (depicted in Figure 2.23) receives a question pair (marked by the community as similar), which is translated into a multidimensional space using word embeddings. These embedding representations are forwarded to two fully connected layers along with several other linguistic features proposed by the authors to produce a similarity score. In order to address the language gap across sentence representations the authors use the cross-lingual embedding learning strategy proposed by Luong et al. [167], which we described in Section 2.2.2.

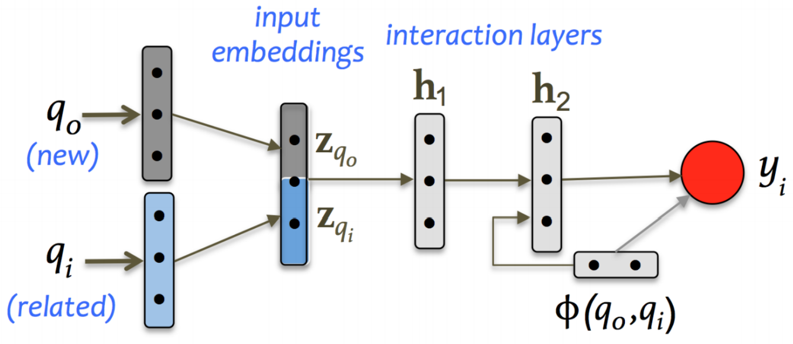


Figure 2.23: Description of the strategy proposed by Da San Martino et al. [50] for question answering. A question pair is feed to a two layer neural network along some other features.

Joty et al. [128] propose a question retrieval strategy aimed at languages where labeled training data is non-existent, i.e., there is no question answering gold-standard. For doing so they propose to train a question retrieval model in a popular language that has large amounts of labeled data and use it in languages where only unlabeled data is available. Joty et al. [128] extend the model previously proposed by Da San Martino et al. [50] using adversarial training, a framework oriented to achieving better generalization on neural networks by creating fake data intended to help the model learn from its errors. This process generates high-level discriminative features that are language invariant as they are based on cross-lingual embeddings generated following the strategy proposed by Luong et al. [167].

In this section, we described research efforts in the area of multilingual QA. Initial approaches address the language gap by translating the questions or the answers to a common language. However, more recent techniques take advantage of structured data, such as Wikipedia, or language invariant word representation techniques, such as cross-lingual word embeddings.

CHAPTER 3

MULTIATTENTIVE RECURRENT NEURAL NETWORK ARCHITECTURE FOR MULTILINGUAL READABILITY ASSESSMENT

Abstract

We present a multiattentive recurrent neural network architecture for automatic multilingual readability assessment. This architecture considers raw words as its main input, but internally captures text structure and informs its word attention process using other syntax- and morphology-related datapoints, known to be of great importance to readability. This is achieved by a multiattentive strategy that allows the neural network to focus on specific parts of a text for predicting its reading level. We conducted an exhaustive evaluation using datasets targeting multiple languages and prediction task types, to compare the proposed model with traditional, state-of-the-art and other neural network strategies.

3.1 Introduction

For decades, readability assessment has been used by diverse stakeholders—from educators to public institutions—for determining the complexity of texts [26]. Traditional formulas do so by focusing only on superficial linguistic features, e.g., average length of sentences or syllables per word. This leads to criticism, as these formulas do not explore deeper levels of text processing and thus yield rough estimates of complexity, i.e., difficulty, that often lack accuracy [13]. In fact, traditional formulas can label a text as “easy to read", even if its content is completely nonsensical [52].

To improve the quality of automatic readability assessment, researchers turned to more sophisticated techniques that go beyond examining shallow features. These techniques, typically based on supervised machine learning, incorporate hundreds (even thousands) of features that describe a text from multiple perspectives: syntax, morphology, cohesion, discourse structure, and subject matter [56, 93, 57, 13]. The dependency on these numerous features, however, has made readability assessment tools too complex to deploy and apply to languages beyond the one for which they were originally designed for. Furthermore, feature and language dependency, along with lack of homogeneity in terms of readability scales, often prevent researchers from comparing new strategies with state-of-the-art counterparts, preventing community consensus on which features are the most beneficial for capturing text complexity [54].

Existing literature reflects the fact that applications that leverage text complexity analysis, including book recommendation or categorization [158, 202], web result summarization [133], and accessibility in the health domain [28, 87], still favor less precise but easier to implement alternatives, with Flesch as the most accepted choice [23, 25]. We argue that this is caused by the uncertainty induced by the lack of: uniformity of readability scales, adaptability among readability assessment tools, and benchmarks.

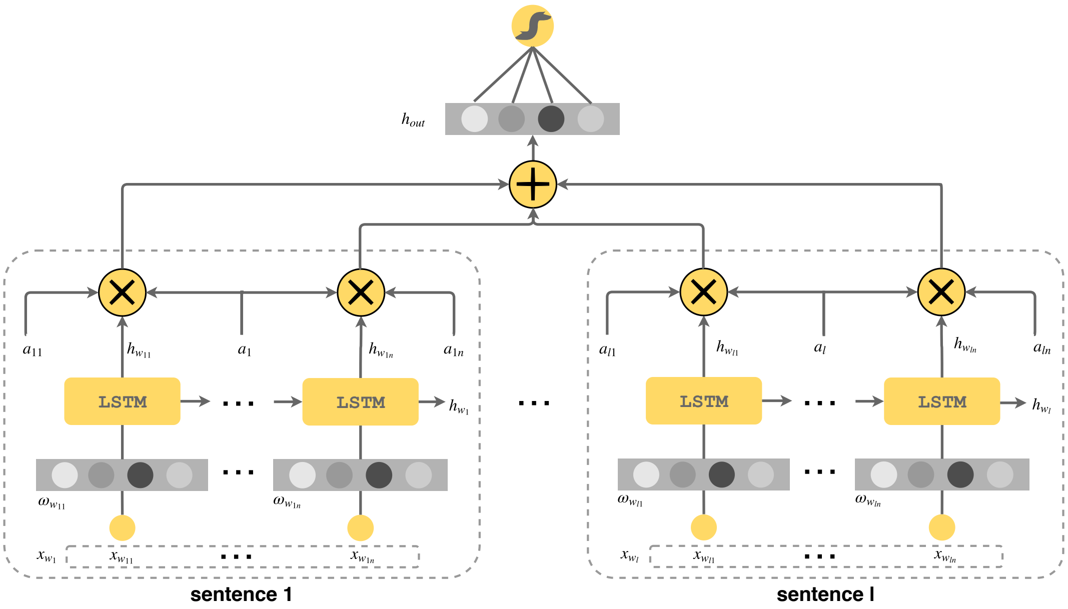
Areas of study that were historically heavily dependent on feature engineering, including sentiment analysis or image processing [177, 2], have made their way towards alternatives that do not involve manually developing features, and instead favor deep learning [251]. This resulted in more reproducible strategies; easily portable to other domains or languages, as they only require implementing the *structure* of a specific neural network and just rely on *core components* of resources, such as words, signals, or pixels, rather than features specifically designed for a domain or language.

Issues pertaining to readability assessment are not limited to performance and adaptability. As stated by Benjamin [26], a teacher should never use a readability score blindly when giving a text to a student, as specifics of the difficulties of the reader and the text should always be considered in this process. For this pairing to be successful, it is imperative for readability assessment tools to provide information beyond a single score. The explainability issue has been addressed in systems like Coh-Metrix [105] by showing users the individual values of the features incorporated in the system. This strategy, however, has been criticized by the education community as most features presented are not straightforward to understand for people without background in both computation and linguistics [75]. More intuitive explanations could greatly ease the use of readability tools.

In this manuscript, we present a multilingual automatic readability assessment strategy based on deep learning: **Vec2Read**[1]. We still follow the premise of words being the core components for a neural network that deals with text. However, in order to avoid the

[1] The implementation and evaluation framework code is available on a public repository: `https://github.com/ionmadrazo/Vec2Read`

Figure 3.1: Description of the general architecture of Vec2Read



aforementioned domain dependency issue and adapt the architecture to the readability task, we inform our model with part of speech (POS) and morphological tags. This is done by a *multiattentive structure* that allows the network to filter important words that influence the final complexity level estimation of a text. Apart from informing the network, the multiattentive structure can also be used to offer users further insights on which parts of a text have the most influence for determining its reading level.

Our research contributions include:

- We propose a multiattentive recurrent deep learning architecture specifically oriented to the readability assessment task.
- The proposed strategy is, to the best of our knowledge, the first capable of estimating readability in more than two languages.

- We incorporate an attention structure that allows a model to use multiple focuses of attention (with different degrees of importance) to inform word selection.
- We conduct an exhaustive evaluation based on different languages, readability-measuring scales, and datasets of varied sizes, in order to compare the performance of Vec2Read with existing baselines, a comparison that is rarely done in this area due to lack of benchmarks.
- We present an initial analysis on the use of attention mechanisms as a potential alternative for providing explanations for readability.

Task Definition. Given a text t, use model M to predict its reading level. The functionality of M is directly dependent on the characteristics of a dataset D used for training: *language* and *readability scale*. The scale can be discrete (binary or multilevel) or continuous. Any language is viable; for dataset availability we train M for Basque, Catalan, Dutch, English, French, Italian, and Spanish.

3.2 Related Work

Literature on automatic readability assessment is rich, not only in the languages for which existing strategies can be applied to, but also on the diversity of linguistic perspectives that have been explored [26, 13].

Feature engineering has been the main focus in the readability assessment area. Techniques that exploit *shallow features*, e.g., number of syllables per word and average sentence length, remain a prominent strategy for estimating complexity levels of texts in diverse languages [88, 228, 4] and show better prediction capabilities than more sophisticated features when considered individually [81]. Language models have also been proved useful when determining the reading level of a text [218]. The use of features capturing the *syntax* of a text have been demonstrated to be of great importance, as illustrated by Karpov et al. [135] who built a system that heavily relies on features based on POS tags and the syntactic dependency tree of a text. Structural features may not influence text complexity estimation for languages like Chinese, which is why some researchers favor analyzing *lexical representations*, i.e., term frequencies [39]. Even if not for most languages, *morphological* features have also been shown to be of great importance, in terms of influencing the complexity level of texts written in languages known to be morphologically rich, such as Basque [101]. For considering *semantic* information in a text, existing works incorporate features related to true or false cognates, as a manner to better capture text difficulty for non-native readers [93], or measure the coherence of the text based on graphical models [183, 184, 185]. Unlike the aforementioned techniques, which rely on engineering features for specific languages and tasks, Vec2Read uses a deep learning strategy that automatically detects patterns related to readability.

Historically, readability assessment tools have been designed and evaluated in one language. To the best of our knowledge, only De Clercq and Hoste [54] evaluate readability assessment performance in more than one language, i.e., Dutch and English, with the purpose of comparing the importance of features in each language. As presented in this manuscript, we go beyond two languages and instead quantify the performance of Vec2Read in seven different languages.

Attention mechanisms have been used with great success in several domains, including image classification [262], question answering [117], and automatic text translation [21].

The attention mechanism proposed for Vec2Read differs from the counterparts applied to the aforementioned tasks in the sense that it provides a composed attention score that can be decoupled to further analyze the influence individual words have in the overall complexity of a text from different linguistic perspectives.

3.3 Method

In this section we introduce Vec2Read, a multiattentive recurrent neural network architecture for readability assessment.

3.3.1 General Architecture

The general architecture of Vec2Read (illustrated in Figure 3.1) is designed to emulate the structure of a text. A text is inherently **recurrent**, as it is composed of a series of words that depend on each other in order to produce a message. A text is also **hierarchical**, as it is composed of structural components such as sentences or paragraphs in order to group information.Vec2Read takes into account both characteristics to better capture text structure. Unlike existing hierarchical neural networks that take advantage of both word and sentence level recurrent layers [265], Vec2Read has a single recurrent layer at word level; hierarchical information is used to generate both word and sentence level attention scores for creating a text representation.

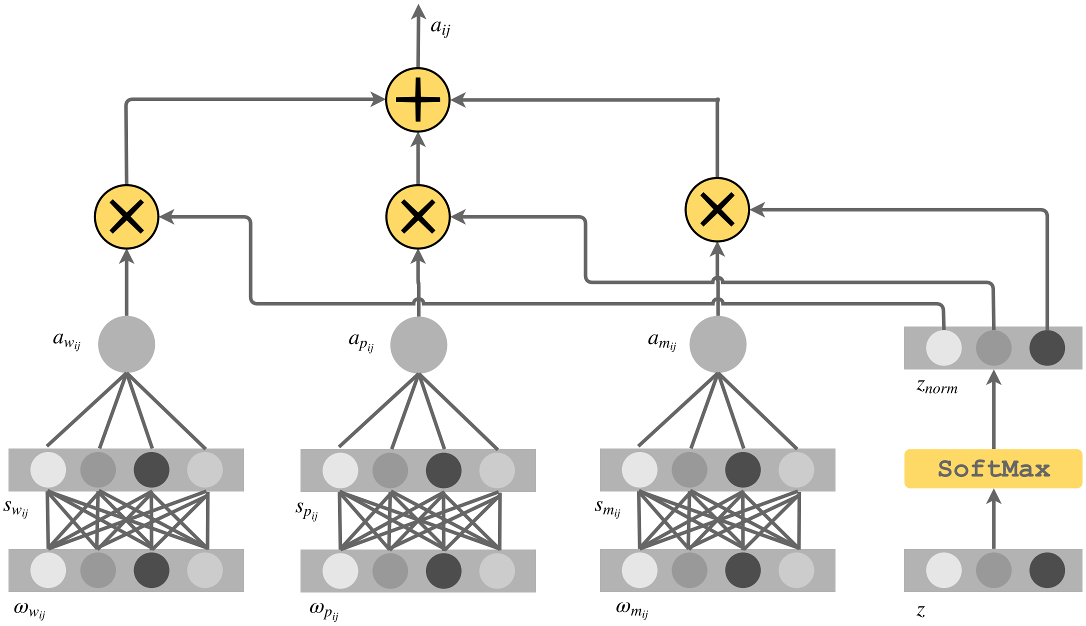
3.3.2 Input

Given a text t, let the input of Vec2Read be $x =< x_w, x_p, x_m >$, where x_w, x_p, and x_m represent data structures containing a sequence of tokens in t, their corresponding POS tags, and morphological tags, respectively. x_{w_i} refers to the i^{th} sentence in t and $x_{w_{ij}}$ is the j^{th} token in x_{w_i}. x_{p_i} and x_{m_i} refer to the POS and morphological tag sequences for x_{w_i}, whereas $x_{p_{ij}}$ and $x_{m_{ij}}$ represent the POS and the morphological tags for $x_{w_{ij}}$. Note that $x_{m_{ij}}$ contains a set of tags per word rather than a single token or POS label. For instance, given the word *plays*: $x_{w_{ij}} =$*"plays"*, $x_{p_{ij}} =$*"Verb"*, and $x_{m_{ij}} =$*"{Tense: present, Person: 3...}"*. To ease further processing, $x_{m_{ij}}$ always contains all possible morphological tags considered for the language, assigning a *Not applicable (NA)* value when the label cannot be applied to the token, e.g., tense would have a value of *NA* for all nouns. The number of tags used is language dependent. (See Section 3.4.1 for details on tag set used in the experiments).

3.3.3 Dense Vector Representations

Dense vector representations or embeddings have shown to be useful for representing discrete values, such as words, in applications dealing with text [234, 174]. Vec2Read converts all discrete values in x into dense vector representations before feeding them to the model. This is achieved by using a lookup table $\Omega_w \in \mathbb{R}^{v \times d}$ where each row is an embedding for a specific word in the vocabulary, v is the vocabulary size and d is the number of latent features used for representation. Similarly, lookup tables Ω_p and Ω_m are used for representing POS and morphological tags, respectively. $\omega_{w_{ij}}$ refers to the embedding of $x_{w_{ij}}$; $\omega_{p_{ij}}$ to the embedding of the POS tag of $x_{w_{ij}}$; and $\omega_{m_{ij}}$ to the embedding that captures the morphological information of $x_{w_{ij}}$ created by concatenating the representations of each morphological tag in $x_{m_{ij}}$. Ω_w, Ω_p, and Ω_m can be either initialized using random uniform distributions and then trained along with the other weights of our model or based on pretrained representations (see Section 3.4.1). Note that representations of each input type are maintained separately and can therefore be of different size.

Figure 3.2: Description of the multiattentive network for token in position j in sentence i.



3.3.4 Encoding Sentences and Words

A recurrent neural network (RNN) [108] is an extension of a traditional neural network where each node in a layer takes as input not only information from the previous layer but also from a node in the same layer located directly next to it. This creates a structure designed to handle sequences like words in a text. Unfortunately, traditional RNNs are prone to the vanishing gradient problem that makes them difficult to train, hindering final performance [119]. A Long Short Term Memory (LSTM) [120] addresses traditional

RNN's vanishing gradient problem by using several gates on each RNN cell responsible for storing or forgetting information from the cell state.

Vec2Read uses a bidirectional LSTM that considers the input sentences in forward and backward directions for creating representations of whole sentences and individual words. We refer to h_{w_i} as the representation of x_{w_i}, obtained by concatenating the outputs of the final states of the LSTM network in both the forward and final pass; $h_{w_{ij}}$ is the representation generated by the LSTM network at time step j (i.e., for word $x_{w_{ij}}$) for i, concatenating the outputs of forward and backward passes.

3.3.5 Textual Representation Layer

A final general representation of t, denoted h_{out}, is created by aggregating all the encoded word representations generated by the LSTM network (Equation 3.1). This is done using a weighted sum over $h_{w_{ij}}$, where the weights are defined by the attention mechanism described in Section 3.3.6.

$$h_{out} = \frac{\sum_{i=1}^{l} \sum_{j=1}^{n_i} a_i a_{ij} h_{w_{ij}}}{\sum_{i=1}^{l} n_i} \tag{3.1}$$

where a_i is the attention generated for sentence i, a_{ij} is the attention for $x_{w_{ij}}$, n_i reflects the number of tokens in sentence i, and l is the number of sentences in t. The denominator is a normalization factor meant to remove the effect of length in texts. This normalization factor is especially important for readability prediction, given that the network could otherwise learn to discriminate texts based mostly on length, due to a strong bias in readability datasets for harder texts to be longer. Informing the model with length distribution of texts in each reading level could lead to performance improvement in an experimental setting. However, doing so would not allow us to estimate model performance in a real scenario, where text length will rarely follow the distribution seen in training sets. Therefore, we favor a length-independent model.

3.3.6 Attention Mechanism

Vec2Read is designed to capture the general structure of t in order to predict its reading level. While one could argue that the reading level of a text is dependent on every one of its words, text simplification studies [97, 197] indicate that difficulty is generally introduced in a text by specific words and sentences–just a few hard sentences could significantly increase overall text difficulty. Following this intuition, Vec2Read uses an attention-generation mechanism (described in Figure 3.2) capable of predicting which parts of t have the most influence in its overall difficult. This way, our model can focus on the important parts of t and provide a more accurate readability estimation.

The attention mechanism of Vec2Read works on two levels: sentence and word. It detects which sentences have most influence towards determining the reading level of t and also which words are most influential. Each of these two-level predictions are composed of three attentions, oriented to consider the influence of each part of t from three linguistic perspectives: semantic, syntactic and morphologic.

We describe below how the multiattentive mechanism works at word level, then we detail how to adapt this model for the sentence level version.

Word Level Attention

The word level attention mechanism consists of three single attention mechanisms that are aggregated. Each individual attention network follows the same structure, a two layer neural network, only differing on the size of the input and the number of hidden units. We set the number of hidden units proportional to the input length (see Section 3.4.1 for configuration details). Specifically, we compute each attention score $a_{att_{ij}}$ as follows:

$$\begin{aligned} s_{att_{ij}} &= \sigma(W_{att} \times \omega_{att_{ij}} + b_{att}) \\ a_{att_{ij}} &= \sigma(W_{att2} \times s_{att_{ij}} + b_{att2}) \end{aligned} \tag{3.2}$$

where $att \in \{w, p, m\}$ is an attention type, W_{att} and W_{att2} are the weights of the first and second network layers, b_{att} and b_{att2} are their respective biases, $s_{att_{ij}}$ is an intermediary representation, and σ is a sigmoid activation function.

Similar to the model in Figure 3.1, the input for generating semantic and syntactic attention scores are $\omega_{w_{ij}}$ and $\omega_{p_{ij}}$. For calculating morphological attention scores, the input is instead the concatenation of each of the morphological tag embeddings in $\omega_{m_{ij}}$.

After generating a score using each single attention mechanism, Vec2Read **aggregates** them into one value that will be the final attention score predicted for $x_{w_{ij}}$. Previous works in feature engineering for readability assessment indicate that not all features are of equal importance for predicting the readability of a text [56, 101]. We believe that this phenomena also applies to attention generation, and therefore each single attention will not contribute equally to the final attention prediction.

To allow our model the flexibility of deciding which attention matters most, we use an attention aggregation strategy that assigns a different weight to each attention. $z =< z_w, z_p, z_m >$ is a vector containing the weights corresponding to each attention mechanism, which are automatically estimated during the training phase to allow Vec2Read to learn which attention has the most influence. We constrain the weights to sum to 1 by applying a softmax function to z:

$$z_{norm_{att}} = \frac{\exp(z_{att})}{\sum_{att} \exp(z_{att})} \tag{3.3}$$

The final attention a_{ij} for $x_{w_{ij}}$ is calculated as:

$$a_{ij} = \sum_{att} z_{norm_{att}} \times a_{att_{ij}} \tag{3.4}$$

Lastly, we constrain all word attentions in a sentence to sum to 1 using a softmax function.

Sentence Level Attention

Sentence level attention follows the same structure as word level attention described in Section 3.3.6, differing only on how the inputs of each single attention network are generated. In this case, for the semantic attention we use h_{w_i} vectors already defined in the general architecture (see Figure 3.1); for syntactic and morphological attentions we feed separate LSTM models using the sequence of syntactic and morphological embeddings in the sentence and use the output of the last recurrent step as input to the attention mechanism. We then normalize sentence level attentions so that they sum to one using a softmax function.

3.3.7 Output Layer

The output layer of Vec2Read is responsible for mapping h_{out} to a reading level prediction. Two different output layers are used depending on the type of prediction required in each task: discrete or continuous.

Discrete Prediction

To predict a discrete reading level for t, Vec2Read generates a probability distribution over each reading level $\hat{y} \in [0,1]^c$, where c represents the set of possible prediction classes, i.e., reading levels. This is achieved by applying a fully connected layer with a softmax activation function to h_{out} to ensure that the probabilities in $\hat{y}$ add up to one.

$$\hat{y} = \text{softmax}(W_{out} \times h_{out}^{\top} + b_{out}) \tag{3.5}$$

where $W_{out} \in \mathbb{R}^{|c| \times r}$ is the matrix of weights of the fully connected layer, b_{out} is a vector of length $|c|$ containing the biases, $|c|$ is the number of possible reading levels to be predicted, r is the number of latent features in h_{out}, and $\top$ refers to the transpose operation. The class that yields the highest probability is the one assigned to t.

Continuous Prediction

When the reading level of t is defined as a continuous value, Vec2Read generates a real value $\hat{y} \in [y_{min}, y_{max}]$, where y_{min} and y_{max} refer to the minimum and maximum readability score possible in the used scale. This is achieved by applying a fully connected layer with a min-max leaky rectified linear unit as activation function. The leaky version of this function is favored given its benefits in terms of avoiding neuron death during training [260].

$$\hat{y} = \vartheta(W_{out} \times h_{out}^{\top} + b_{out}) \tag{3.6}$$

$$\vartheta(q) = \begin{cases} y_{max} + \varepsilon * q, & q > y_{max} \\ q, & y_{min} < q < y_{max} \\ y_{min} - \varepsilon * q, & q < y_{min} \end{cases} \tag{3.7}$$

where $W_{out} \in \mathbb{R}^{1 \times r}$ is the matrix of weights of the fully connected layer, b_{out} is a bias, r is the number of latent features in h_{out}, $\top$ refers to the transpose operation, and ε is a constant set to 0.001 during training and to 0 during prediction.

3.3.8 Fitting Parameters

For fitting the parameters of our model we use stochastic gradient descent. This strategy computes the prediction of our model given specific data, and compares it to the actual objective value using an error or loss function. The goal is to minimize the error for which a gradient is backpropagated to each of the parameters in the model by subsequently updating them in a direction that will minimize the overall prediction error. As the objective function for training the model, we consider two different loss functions, depending on how the reading level is estimated.

For **discrete** predictions, we used cross-entropy:

$$H(y, \hat{y}) = -\sum_{i=1}^{|c|} y_i \log(\hat{y}_i) \tag{3.8}$$

where $\hat{y} =<\hat{y}_1, .., \hat{y}_{|c|}>$ is the probability distribution predicted by our model and $y =<y_1, .., y_{|c|}>$ is the one-hot encoded vector representing the target class.

For **continuous** predictions, we use instead mean square error (MSE):

$$\text{MSE} = \frac{1}{|D|} \sum_{d \in D} (\hat{y}_d - y_d)^2 \tag{3.9}$$

where D is a collection of texts in a given dataset, $|D|$ is the number of documents in D, and $\hat{y}_d$ and y_d are the prediction generated by our model for document d and its ground-truth, respectively.

3.4 Experiments and Discussion

In this section, we first describe model configuration. We then outline datasets and baselines considered for evaluation purposes. Lastly, we discuss the results of the analysis conducted to verify the overall performance of Vec2Read and showcase the validity of its attention mechanism.

3.4.1 Model Setup

We describe Vec2Read's configuration; parameters were empirically determined using a hold-out set described in Section 3.4.4.

Optimization. For fitting the parameters of our model, we used the Adaptive Movement Estimation [142]; learning rate = 0.001.

Initializations. For Ω_w we used a pretrained version of word embeddings, which were trained using a skip-gram algorithm on Wikipedia documents, as described in [30]. All the remaining weights and biases of our model, as well as initial states of LSTM layers, were initialized using a random uniform distribution.

Dimensions. The number of hidden units in the semantic, syntactic, and morphologic LSTM networks were empirically set to 128, 32, 64, respectively. The dimensions of the embedding representations were set to 300, 16, 16. Given that the input of the morphological attention combines multiple embeddings corresponding to the morphological labels used, the final dimension of $\omega_{m_{ij}}$ is $u \times 16$, where u is the number of tags used.

Tagging. We used SyntaxNet [11] trained on Universal Dependencies datasets v1.3 for computing the POS and Morphology tags of words. All POS and Morphology tags available in the dataset were used. Accuracy per language is varied, Dutch being the one with lowest accuracy (POS:89.89%, Morph:89.12%) and Catalan the language where the tagging is most accurate (POS:98.06%, Morph:97.56%)[2].

3.4.2 Datasets

For assessment and analysis purposes, we use several datasets based on both expert-labeled educational materials (Ikasbil, Newsela, Wizenoze) and crowd-source generated and simplified texts (MTDE, SimpleWiki, VikiWiki). We describe each dataset below; detailed statistics are in Table 3.1.

SimpleWiki. Simple.Wikipedia(.org) is a simplified version of the most representative articles in English Wikipedia written with simple vocabulary and grammar. These articles target readers who are learning English. We created a binary (simple or complex) dataset using the 131,459 articles available in Simple Wikipedia and their Wikipedia counterparts,

[2] For per language accuracy details see `https://github.com/mldbai/tensorflow-models/blob/master/syntaxnet/universal.md`

totaling 262,918 documents. The use of Simple.Wikipidia/Wikipedia articles has already proved to be useful for readability and simplification assessment [8], a fact we confirm in our qualitative analysis in Section 3.4.6. (See [255] for details on how articles on Simple.Wikipedia are simplified.)

VikiWiki. Vikidia(.org) is similar to Simple Wikipedia, but it is not constrained to articles written in English. Following a similar procedure to SimpleWiki, we created VikiWiki using all the articles in Vikidia along with their Wikipedia counterparts. The dataset is comprised of 70,514 documents: 23,648 in French, 9,470 in Italian, 8,390 in Spanish, 3,534 in English, 924 in Catalan, and 898 in Basque, uniformly distributed among simple and complex levels.

MTDE. MTDE is the dataset presented in [54], generated using crowd-sourcing techniques. It consists of 105 documents both in English and Dutch, each labeled with a score in 0-100 range that indicates its complexity.

Newsela. Newsela is an instructional content platform that provides reading materials for classroom use. As part of their research program, Newsela makes available a sample of their labeled corpora, which we use for evaluation. The dataset consists of 10,786 documents distributed among grade levels 2-12 (around 1,200 per level for English and 120 for Spanish). We excluded from our experiments grade levels 2, 10 and 11, as the amount of documents for those levels are significantly lower when compared to other classes (284, 11 and 2 respectively for English).

Ikasbil. Ikasbil [125] is an online resource for learning Basque containing articles leveled following the Common European Framework of Reference for Languages. Using this source, we created a dataset consisting of *5 reading levels* (A2, B1, B2, C1, and C2), with

200 documents per level. Level A1 was omitted due to insufficient documents.

Wizenoze. Dataset provided by Wizenoze [256], an online platform dedicated to easing the retrieval of (curated) resources suitable for the classroom setting. The dataset consists of 2,000 documents in English and Dutch, equally distributed and labeled using a 5-level readability scale (1-5).

	SimpleWiki		**Wizenoze**					**Newsela**							
	S	**C**	**1**	**2**	**3**	**4**	**5**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**12**
Words per text	111	5987	35	128	67	266	801	448	674	777	872	927	990	970	1169
Sentences per text	6	222	3	10	5	16	35	43	54	54	54	52	46	46	50
Syllables per word	1.31	1.37	1.40	1.41	1.44	1.52	1.53	1.27	1.30	1.33	1.36	1.39	1.40	1.43	1.42
Words per sentence	17	25	11	14	14	16	21	10	12	14	16	18	20	21	24
Ratio of unique words	0.69	0.32	0.86	0.79	0.79	0.65	0.55	0.44	0.42	0.42	0.42	0.43	0.43	0.43	0.43
Flesch-Kincaid	6.37	10.70	5.40	6.61	6.65	8.65	10.76	3.42	4.63	5.72	6.81	7.77	8.73	9.68	10.48

	WikiViki		**Ikasbil**					**MTDE***			
	S	**C**	**A2**	**B1**	**B2**	**C1**	**C2**	**1**	**2**	**3**	**4**
Words per text	303	6036	215	276	320	327	354	294	276	288	301
Sentences per text	16	217	21	18	18	16	15	11	12	13	23
Syllables per word	1.36	1.38	1.40	1.39	1.43	1.37	1.40	1.51	1.47	1.37	1.23
Words per sentence	17	25	10	15	17	20	23	26	23	23	14
Ratio of unique words	0.62	0.31	0.51	0.53	0.52	0.52	0.51	0.56	0.56	0.54	0.49
Flesch-Kincaid	7.13	10.71	4.83	6.66	7.91	8.38	9.90	12.5	10.71	9.63	4.64

Table 3.1: Statistics on the datasets considered in our assessment, where S and C stand for Simple and Complex, respectively. When datasets are multilingual, texts from all languages are considered for computing average. * Given that ground truth scores for MTDE are continuous, for illustration purposes we reported statistics grouped in 4 levels, i.e., 0-25, 26-50, 51-75,76-100 (original values preserved in the experiments).

3.4.3 Compared Strategies

We describe below strategies considered in our assessment, including traditional formulas, state-of-the-art tools based on extensive feature engineering, and neural network structures intended for an ablation study on major components of Vec2Read.

Traditional Strategies

Flesch. Even if simple, Flesch [88] remains one of the most used readability formulas and is therefore treated as a baseline by authors of publications pertaining to readability estimation. In addition to the traditional version for English texts, we consider language-specific adaptations [131, 165, 83, 61]. We followed the framework used in [168], which maps the Flesch score of a given text t into a binary value (simple or complex) based on its distance with the average Flesch score computed using the training documents for the respective classes.

State-of-the-Art Strategies

S1. The system proposed by De Clercq and Hoste [54] is the only one designed for readability assessment for more than one language: Dutch and English. Its design consists of a Support Vector Machine that uses ad-hoc features to capture varied linguistic characteristics of texts, e.g., syntax or semantics. Given that the algorithm implementation is not publicly available, comparisons against this strategy are based on results reported in [54].

S2. A multilevel Basque readability assessment strategy that relies on Random Forest and linguistic features with a major emphasis on morphology and syntax [179]. The authors provided their dataset (including cross-validation folds) for comparison purposes. Due to lack of implementation availability, comparisons against S2 are limited to the Basque language.

S3. Similar to S2, the strategy introduced in [168] also relies on a Random Forest and linguistic features. Given implementation availability, we adapted it to run on all discrete and continuous prediction tasks by changing its linguistic annotation tools. For fairness in the comparison, we used the same linguistic annotation tools used by Vec2Read (described in Section 3.4.1).

S1, S2, and S3 are treated as examples of feature engineered state-of-the-art strategies.

Ablation Study Strategies

To determine the utility of each feature incorporated in the architecture of Vec2Read, we consider several variations of Vec2Read in the assessment.

FC. A two layer fully connected neural network with 256 hidden units, taking as input the average of the word embeddings of all words in a text.

¬Attention. Basic architecture of Vec2Read. It maintains Vec2Read's hierarchical and recurrent structure, but overrides the output of its attention generation mechanism by assigning each word and sentence a uniformly-distributed attention score.

¬Word, **¬Sent**, **¬Sem**, **¬Syn** and **¬Morph**. Vec2Read architecture without word level, sentence level, semantic, syntactic, and morphological attention, respectively.

3.4.4 Experimental Setup

We followed a 10-cross-fold validation framework for measuring the performance of each strategy considered. A disjoint stratified 10% of data in SimpleWiki (includes both simple and complex) was excluded from the experiments and used for developmental and hyper-parameter tuning purposes. Note that to abide by the adaptability premise intended for our model, we only tuned hyper-parameters for English. Doing so allows us

Dataset	Lang.	Flesch	State of the art			Ablation							Vec2Read
			S1	S2	S3	FC	¬Attention	¬Word	¬Sent	¬Sem	¬Syn	¬Morph	
Binary Prediction (Accuracy)													
SimpleWiki	en	.724	-	-	.822	.722	.877	.893	.896	.887	.897	.915	**.918***
VikiWiki	en	.720	-	-	.827	.721	.852	.860	.862	.859	.868	.876	**.879***
	es	.687	-	-	.792	.719	.816	.823	.831	.828	.835	.839	**.847***
	fr	.670	-	-	.842	.756	.864	.869	.870	.869	.870	.872	**.884***
	it	.653	-	-	.755	.766	.783	.797	.802	.793	.801	.805	**.814***
	eu	-	-	-	**.693**	.648	.682	.683	.686	.684	.684	.685	.687
	ca	-	-	-	.733	.677	.715	.725	.737	.728	.732	.734	**.742**
Multilevel Prediction (Accuracy)													
Ikasbil	eu	-	-	.625	.622	.617	.679	.685	.689	.681	.684	.686	**.692***
Newsela	en	-	-	-	.464	.447	.489	.501	.517	.498	.502	.525	**.527***
	es	-	-	-	.467	.452	.487	.494	.510	.504	.509	.503	**.519***
Wizenoze	en	-	-	-	.649	.631	.665	.678	.685	.682	.685	.700	**.701***
	du	-	-	-	.652	.636	.668	.679	.687	.681	6.85	.683	**.695***
Continuous Prediction (RMSE)													
MTDE	du	-	**.0003***	-	-	.0171	.0068	.0064	.0064	.0066	.0062	.0059	.0059
	en	-	.0060	-	-	.0184	.0054	.0052	.0051	.0051	.0053	.0051	**.0051**

Table 3.2: Performance comparison among traditional, state-of-the-art, ablation strategies, and Vec2Read on different datasets. '*' denotes a statistically significant improvement was found using Vec2Read compared with its counterparts (Flesch, S1, S2, S3), using a paired T-test ($p < 0.05$). Accuracy (higher is better) is reported for all datasets except for MTDE, where RMSE (lower is better) is used in order to be able to compare with S1. Cells marked with '-' denote that the strategy is not applicable to the dataset.

to understand to what extent the model can directly transfer to other languages without language-specific tuning, thus simulating a real-world scenario for tool adaptation.

To conduct fair comparisons, we used the same cross-validation folds across experiments (when possible, we used the folds made publicly available; otherwise we re-run strategies using our data and folds). The only exception are experiments related to S1, for which we could only access the original dataset. Consequently, we compare our results with respect to those published in [54].

3.4.5 Overall Performance

As mentioned by De Clercq and Hoste [54], each work in the readability area interprets the readability estimation task in a different manner–using different languages and datasets–often making the community unable to compare proposed tools with each other. In order to best contextualize the performance of Vec2Read, we consider a broad set of tasks using datasets of varied (i) **size**, that go from 105 documents to 262,918, (ii) **language**, considering seven languages, and (iii) **prediction type**, i.e., binary, multilevel, and continuous predictions.

To quantify performance of different readability estimation alternatives, we use *accuracy* for classification tasks and *Root Mean Square Error* (RMSE) for regression tasks. Table 3.2 summarizes the results obtained by Vec2Read and its counterparts on the aforementioned datasets. As we followed a 10-cross-fold validation framework, scores in Table 3.2 correspond to the averages over the 10 folds. Statistical significance improvement for Vec2Read with respect to its counterparts was tested using a paired T-test with a confidence interval of $p < 0.05$.

General discussion. As anticipated, we observe that traditional formulas (Flesch) yield the lowest performance, followed by the general-purpose neural network approach (FC). This validates our hypothesis that a neural network that simply considers words without considering text structure or other linguistic features is not enough for readability assessment. Further, models that consider richer traits of text, such as Vec2Read and its attention-less version ($\neg$*Attention*), are consistently comparable or outperform state-of-the-art strategies (S1,S2,S3) demonstrating the validity of the proposed architecture. Vec2Read achieved a statistically-lower rate only for 1 out of 14 tasks (defined as a dataset-language pair) in our evaluation. We attribute this to the size of the dataset, which is only comprised of 105 texts. It is anticipated for a strategy based on feature engineering such as S1, which has been specifically designed for Dutch, to outperform a neural network based counterpart (such as

Vec2Read), as the latter is known to need large amounts of data for best performance.

Dataset	Lang	Words	Part Of Speech	Morphological
SimpleWiki	en	unincorporated, reside, inhabitants	CCONJ, SCONJ, DET	Relative (pronoun), Past, Infinitive
VikiWiki	en	belonged, abolished, comprising	SCONJ, CCONJ, DET	Relative (pronoun), Infinitive, Past
	es	recae, mantiene, consiste	SCONJ, ADJ , AUX	Participle, Subjunctive, Past
	fr	circonscriptions, associer, comporter	CCONJ, ADV, SCONJ	Reflexive, Subjunctive, Passive
	it	comprende, risiede, rivelato	CCONJ, SCONJ, VERB	Past, Subjunctive, Relative (pronoun)
	eu	aldarrikapen, gizarte, eskumen	NOUN, ADJ, DET	Subjunctive, Inessive, Dative
	ca	acreditat, mantenir, contribuint	ADJ, NOUN, CCONJ	Subjunctive, Relative (pronoun), Participle
Ikasbil	eu	hedatu, irudikatu, biltzartu	CCONJ, VERB, SCONJ	Subjunctive, Genitive, Inessive
MTDE	du	geregeld, omvat, stemhebbend	NOUN, ADJ, CCONJ	Past, Participle, Infinitive
	en	handled, retained, consisting	NOUN, SCONJ, ADJ	3rd Person, Relative (pronoun), Past
Newsela	en	aquaponics, government, unwavering	CCONJ, SCONJ, ADJ	Infinitive, Relative (pronoun), Past
	es	postularse, extintos, realizacion	CCONJ, SCONJ, AUX	Subjunctive, 3rd Person, Participle
Wizenoze	en	controversy, transition, equality	SCONJ, CONJ, NOUN	Relative (pronoun), 3rd Person, Past
	du	vervaardiging, afgezette, bijgevolg	CCONJ, NOUN, ADJ	Participle, Past, Infinitive

Table 3.3: Words, POS tags, and Morphological tags that receive highest attention from Vec2Read.

Dataset size. The number of instances used for training has a strong effect on the overall performance of Vec2Read. All the analyzed strategies generate lower scores for smaller datasets; performance drop is more prominent among the strategies based on deep learning (Vec2Read and all the ablation strategies). We attribute this behavior to the higher variance of deep learning models, needing more data than feature engineered models to achieve good generalization. In addition, we also note that the attention mechanism becomes more useful the larger the dataset and its effect is negligible in small datasets such as MTDE.

Language and task type. We observe no emerging patterns in terms of performance induced by the language or the type of task. One could argue that results for English are in general higher, however, we attribute these differences to dataset size (English datasets are in general larger) rather than to the language itself. Accuracy scores for multilevel estimation are lower than for binary, which is expected, as it is harder for a model to learn readability predictions for scales that go beyond just simple or complex.

Ablation study. By comparing Vec2Read with its attention-less counterpart ($\neg$*Attention*) we can conclude that the proposed multiattentive mechanism has indeed a positive effect for readability prediction. In 11 out of 14 tasks the multiattentive mechanism achieved statistically significant improvements over $\neg$*Attention*; for the remaining 3 tasks (VikiWiki-EU, MTDE-EN, MTDE-DU) there was no statistically relevant difference. The usefulness of the attention mechanism is influenced by the size of the dataset, as the larger the dataset, the more prominent the improvement obtained by the model using the attention mechanism. We also notice that the difference of using the morphological attention for certain languages such as English is insignificant while it is more prominent in other languages, a fact we attribute to the low morphological diversity of English.

3.4.6 Attention Mechanism

As outlined in Section 3.4.5, the attention mechanism of Vec2Read leads to improvement in prediction performance. In this section, we aim to shed light on what the attention mechanism is actually learning to do and whether this information could be used for explaining the estimated reading levels from a more *qualitative* standpoint. Even if attention mechanisms are used in manifold applications, there exists no defined framework for evaluating their behavior. Instead, researchers focus on finding explanations of what the mechanism is learning [117, 262]. For this reason, the following discussion is not intended to be conclusive but instead provide initial results meant to be inspirational for future work on readability prediction explainability.

[2] For definition of POS tags refer to https://universaldependencies.org/u/pos/

In order to illustrate the parts of a text that receive the most attention from Vec2Read, we show in Table 3.3 the top-3 words, POS, and morphological tags, that score the highest attention level for each individual task. We observe that words that receive most attention are in general words that are not frequently used by an average speaker, and therefore can present a challenge for the reader. We also observe that conjunctions (used for making sentences longer) are consistently among the most influential POS tags and that subjunctive mood, passive voice, and specific verb forms, such as infinite or participle, are considered important by our model. Both the use of conjunctions and passive voice align with features already found positive in the readability literature [93, 101], leading us to infer that the attention mechanism is learning valid assumptions for detecting which parts of a text are most influential for readability prediction.

One of the benefits of using a multiattentive mechanism, compared to a traditional attention mechanism that considers all features at once, is that the model can adapt and give more importance to specific datapoints depending on the task. In order to illustrate how Vec2Read takes advantage of this functionality, we show in Table 3.4 the weights[3] assigned by the attention mechanism for each task, i.e., z_{norm}. We observe that higher weight is assigned to semantics when the dataset is large, while syntax is more relevant for smaller datasets. This behavior depicts the adaptability of our model, using more generalizable information, such as POS tags, when data is scarce and taking advantage of more fine grained information, such as words, when data is abundant. Weights for morphological and syntactic attention are similar for most of the tasks with the exception of English, where morphology receives a lower weight compared to other languages. We attribute this

[3] Weights averaged across 10 folds, see Section 3.4.5.

phenomena to English being a morphologically-poor language.

	SimpleWiki	**VikiWiki**						**Ikasbil**	**MTDE**		**Newsela**		**Wizenoze**	
	en	**en**	**es**	**fr**	**it**	**eu**	**ca**	**eu**	**en**	**du**	**en**	**es**	**en**	**du**
Semantic	.72	.67	.53	.55	.62	.43	.39	.68	.38	.31	.62	.41	.39	.32
Syntactic	.20	.26	.26	.19	.21	.28	.41	.15	.32	.37	.30	.35	.32	.40
Morphological	.08	.07	.21	.26	.17	.29	.20	.17	.30	.32	.08	.24	.29	.28

Table 3.4: Weights learned by Vec2Read for each of the datasets considered.

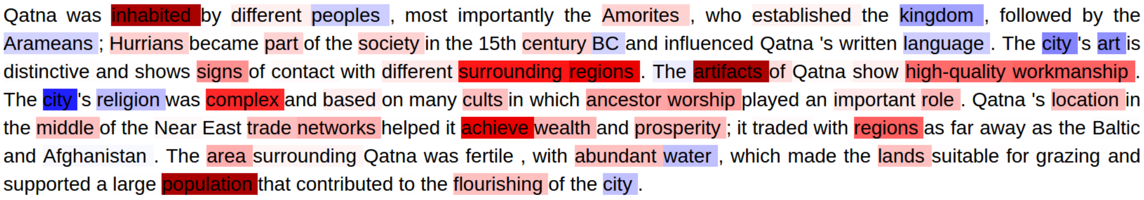
Consider Figures 3.3 and 3.4, two examples of attentions generated by Vec2Read. Figure 3.3 showcases the combined attention scores a_{ij} predicted by Vec2Read for a text snippet extracted from the English Wikipedia document about Qatna. The model used for predicting the attentions was trained using the SimpleWiki dataset. In this example, we see that Vec2Read mostly focuses on complex nouns and adjectives, and tends to ignore less informative words, such as determiners.

Figure 3.4 shows the attentions generated for a sentence in Spanish by Vec2Read trained using Spanish VikiWiki. This example is meant to illustrate the “extra" information that can be obtained from a multiattentive mechanism, not only by showing which of the words are important for estimating text difficulty, but also hinting about why they influence the process. As captured in Figure 3.4, the connector *Consequentemente* (Consequently) is most important from a syntactic perspective, while the sequence *fue cerrado* (was closed) is more important from a morphological standpoint.

Manual analysis of the attention scores lead us to identify which parts of a text the model is focusing on. This initial examination reveals that the model is indeed learning about linguistic patterns known to be important for defining the difficulty of a text as opposed to stylistic biases caused by how the datasets were generated. This also serves as an indication for the validity of using crowd-sourced datasets, such as SimpleWiki and VikiWiki, for training purposes.

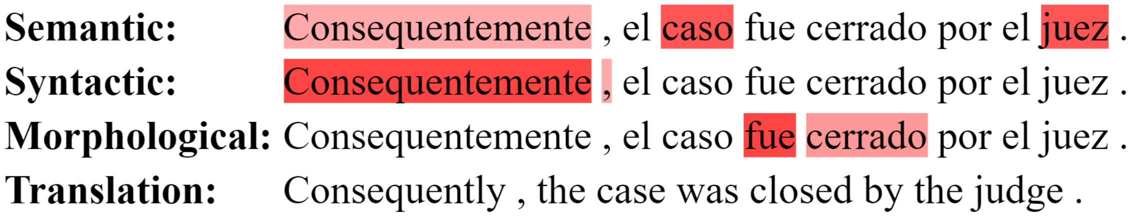
We found many examples where the multiattentive mechanism yielded interesting outputs, however, we also found some deficiencies we would like to highlight. Even if connectors, like *Consequentemente*, were detected correctly by Vec2Read, other commonly used connectors, such as *sin embargo* (nevertheless) or *a pesar de ello* (nonetheless), were not detected correctly given their multi-word structure. This indicates that word level attentions might not be enough for some languages, thus, demonstrating the need to consider more sophisticated structures such as dependency trees, as well as other syntactic and morphological features of the text, in the future.

Figure 3.3: Attention scores generated by Vec2Read for a snippet of a Wikipedia article about Qatna. Color saturation indicates magnitude of the attention score, while hue indicates polarity (blue for simple, red for complex). Magnitudes are provided by the attention mechanism while the polarities are determined by the readability prediction generated when using each word as input to Vec2Read.



Qatna was inhabited by different peoples , most importantly the Amorites , who established the kingdom , followed by the Arameans ; Hurrians became part of the society in the 15th century BC and influenced Qatna 's written language . The city 's art is distinctive and shows signs of contact with different surrounding regions . The artifacts of Qatna show high-quality workmanship . The city 's religion was complex and based on many cults in which ancestor worship played an important role . Qatna 's location in the middle of the Near East trade networks helped it achieve wealth and prosperity ; it traded with regions as far away as the Baltic and Afghanistan . The area surrounding Qatna was fertile , with abundant water , which made the lands suitable for grazing and supported a large population that contributed to the flourishing of the city .

Figure 3.4: Scores generated using individual attention mechanisms by Vec2Read for a sentence in Spanish; saturation indicates the magnitude of the attention score.



Semantic: Consequentemente , el caso fue cerrado por el juez .
Syntactic: Consequentemente , el caso fue cerrado por el juez .
Morphological: Consequentemente , el caso fue cerrado por el juez .
Translation: Consequently , the case was closed by the judge .

3.5 Conclusion

In this chapter, we have introduced Vec2Read, a multiattentive recurrent neural network architecture designed for automatic multilingual readability assessment. Vec2Read takes advantage of deep learning techniques by incorporating a multiattentive mechanism that allows the system to consider words and sentences that most influence the reading level of a text. We demonstrated the validity of our proposed architecture by conducting an exhaustive analysis using datasets in seven different languages and comparing Vec2Read to traditional, state-of-the-art, and other neural network architectures. Moreover, we outlined the benefits of this type of architecture for readability assessment, including the interpretability of the predictions using the attention scores.

This research work sets the foundations for language agnostic readability assessment, demonstrating that it is indeed possible to design a readability assessment strategy that works regardless of the language. This is achieved by disregarding hand-engineered features, historically known to be tedious to create and test, in favour of using simple tokens as input. We anticipate that given the magnitude and the diversity of the evaluation conducted, we have set a new baseline in the readability area, considerably harder to beat than the popularly-used Flesch. This is supported by (i) the use of datasets in multiple languages that can, for the most part, be easily obtained and (ii) the release of our algorithm, so that other researchers can run it for comparison purposes. We expect this will make an area that is currently crowded with hard-to-compare systems, finally progress towards more precise, usable, and comparable tools.

In the future, our research will be focused on generating more valuable explanations on what influences the readability of a text, as well as enhancing our model so that it can be trained jointly for multiple languages or can obtain benefit of cross-lingual data in order to improve the performance in languages with small corpora. We also plan on experimenting with character based models, which could potentially take advantage of morphological information of texts without the need of a morphological tagger.

CHAPTER 4

HIERARCHICAL MAPPING FOR CROSS-LINGUAL WORD EMBEDDING ALIGNMENT

Abstract

The alignment of word embedding spaces in different languages into a common cross-lingual space has recently been in vogue. Strategies that do so compute pairwise alignments and then map multiple languages to a single pivot language (most often English). These strategies, however, are biased towards the choice of the pivot language, given that language proximity and the linguistic characteristics of the target language can strongly impact the resultant cross-lingual space in detriment of topologically distant languages. We present a strategy that eliminates the need for a pivot language by learning the mappings across languages in a hierarchical way. Experiments demonstrate that our strategy significantly improves vocabulary induction scores in all existing benchmarks, as well as in a new non-English centered benchmark we built, which we make publicly available.

4.1 Introduction

Word embeddings have changed how we build text processing applications, given their capabilities for representing the meaning of words [187, 201, 30]. Traditional embedding-generation strategies create different embeddings for the same word depending on the language. Even if the embeddings themselves are different across languages, their distributions tend to be consistent—the relative distances across word embeddings are preserved regardless of the language [188]. This behaviour has been exploited for cross-lingual embedding generation by aligning any two monolingual embeddings spaces into one [58, 259, 14].

Alignment techniques have been successful in generating bilingual embedding spaces that can later be merged into a cross-lingual space using a pivoting language, being English the most common choice. Unfortunately, mapping one language into another suffers from a neutrality problem, as the resultant bilingual space is impacted by language-specific phenomena and corpus-specific biases of the target language [62]. To address this issue, Doval et al. [62] propose mapping any two languages into a different *middle* space. This mapping, however, precludes the use of a pivot language for merging multiple bilingual spaces into a cross-lingual one, limiting the solution to a bilingual scenario. Additionally, the pivoting strategy suffers from a generalized bias problem, as languages that are the most similar to the pivot obtain a better alignment and are therefore better represented in the cross-lingual space. This is because language proximity is a key factor when learning alignments. This is evidenced by the results in [15] which indicate that when using English (Indo-European) as a pivot, the vocabulary induction results for Finnish (Uralic) are about 10-points below the rest of the Indo-European languages under study.

If we want to incorporate *all languages* into the same cross-lingual space regardless of their characteristics, we need to go beyond the *train-bilingual/merge-by-pivoting* (**TB/MP**) model, and instead seek solutions that can directly generate cross-lingual spaces without requiring a bilingual step. This motivates the design of **HCEG** (Hierarchical Cross-lingual Embedding Generation), the hierarchical pivotless approach for generating cross-lingual embedding spaces that we present in this manuscript. HCEG addresses both the language proximity and target-space bias problems by learning a compositional mapping across multiple languages in a hierarchical fashion. This is accomplished by taking advantage of a language family tree for aggregating multiple languages into a single cross-lingual space. What distinguishes HCEG from TB/MP strategies, is that it does not need to include the pivot language in all mapping functions. This enables the option to learn mappings between typologically similar languages, known to yield better quality mappings [15].

The main contributions of our work include:

- A strategy that leverages a *language family tree* for learning mapping matrices that are composed hierarchically to yield cross-lingual embedding spaces for language families.
- An enhanced unsupervised initialization technique that considers the frequency of occurrence of words in a language, yielding superior initialization word-pair sets that lead to better initial solutions and reduce the number of iterations needed for convergence.
- An analysis of the benefits of hierarchically generating mappings across multiple languages compared to traditional unsupervised and supervised TB/MP alignment strategies.

- A dataset[1] of train/test cross-lingual word pairs for combinations across languages, that can be used for evaluating word embeddings mapping techniques for languages that do not rely on English as a target or source.

4.2 Related Work

Recent interest in cross-lingual word embedding generation has lead to manifold strategies that can be classified into four groups [212]: 1) **Mapping** techniques that rely on a bilingual lexicon for mapping an already trained monolingual space into another [188, 15, 62]; 2) **Pseudo-cross-lingual** techniques that generate synthetic cross-lingual corpora that are then used in a traditional monolingual strategy, by randomly replacing words of a text with their translations [103, 69] or by combining texts in various languages into one [248]; 3) Approaches that only optimize for a **Cross-lingual objective** function, which require parallel corpora in the form of aligned sentences [115, 152] or texts [224]; and 4) Approaches using a **Joint objective** function that optimizes both mono- and cross-lingual loss, that rely on a parallel corpora aligned at the word [273, 167] or sentence level [104, 49].

A key factor for cross-lingual embedding generation techniques is the amount of supervised signal needed. Parallel corpora is a scarce resource; even nonexistent for some isolated or low-resource languages. Thus, we focus on Mapping-based strategies that can go from requiring just a bilingual lexicon [188] to absolutely no supervised signal [17]. This aligns with one of the premises for our research to enable the generation of a single cross-lingual embedding space for as many languages as possible.

[1] Code and datasets are available at ionmadrazo.github.io/

Mikolov et al. [188] first introduced a mapping strategy for aligning two monolingual spaces that learns a linear transformation from source to target space using stochastic gradient descend. This approach was later enhanced with the use of least squares for finding the optimal solution, L2-normalizing the word embedding, or constraining the mapping matrix to be orthogonal [58, 222, 259, 14, 223]; enhancements that soon became standard in the area. These models, however, are affected by **hubness**, where some words tend to be in the surrounding of an exceptionally large amount of other words causing problems when using Nearest-Neighbour as the retrieval algorithm, and **neutrality** where the resultant cross-lingual space is highly conditioned by the characteristics of the language used as target. Hubness was addressed by a correction applied to Nearest-Neighbour retrieval whether using a inverted softmax [223] or a cross-domain similarity local scaling [48] later incorporated as part of the training loss [129]. Neutrality was noticed by Doval et al. [62], for which they proposed using two independent linear transformations so that the resulting cross-lingual space is in a *middle* point between the two languages rather than just on the target language, and therefore, not biased towards either language.

Recent approaches strive to avoid any sort of supervision to learn mapping functions [48, 17]. This is particularly advantageous when dealing with low-resource languages for which supervised signal can be hard to obtain.

Our work is inspired by Doval et al. [62], in the sense that it focuses on obtaining a non-biased or neutral cross-lingual space that does not need to be centered in English (or any other pivot language) as the primary source. This neutrality is obtained by a compositional mapping strategy that hierarchically combines mapping functions in order to generate a single, non-language-centered cross-lingual space, enabling a better mapping for languages that are distant or non-typologically related to English.

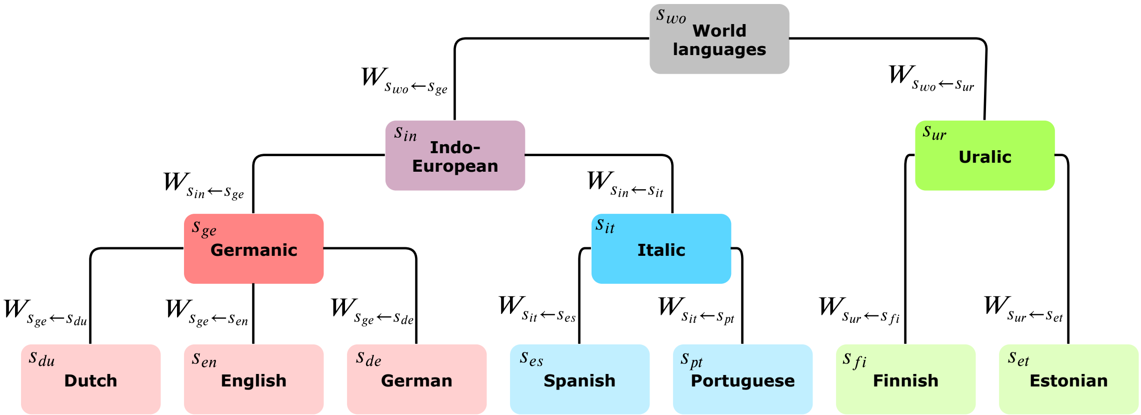
4.3 Proposed Strategy

A language family tree is a natural categorization of languages that has historically been used by linguistics as a reference that encodes similarities and differences across languages [47]. For example, based on the relative distances among languages in the tree illustrated in Figure 4.1, we infer that both Spanish and Portuguese are relatively similar to each other given that they are part of the same Italic family. At the same time, both languages are farther apart from English than each other, and are radically different with respect to Finnish.

A language family tree offers a natural organization that can be exploited when building cross-lingual spaces that integrate typologically diverse languages. We leverage this structure in HCEG, in order to generate a hierarchically compositional cross-lingual word embedding space. Unlike traditional TB/MP strategies that generate a single cross-lingual space, the result of HCEG is a set of transformation matrices that can be used to hierarchically compose the space required in each use-case. This maximizes the typological intra-similarity among languages used for generating the embedding space, while minimizing the differences across languages that can hinder the quality of the cross-lingual embedding space. Thus, if an external application only considers languages that are Germanic, then it can just use the Germanic cross-lingual space generated by HCEG, while if it needs languages beyond Germanic it can utilize a higher level family, such as the Indo-european. This can not be done with the traditional TB/MP model. In this case, if an application is, for example, using only Uralic languages, then it would be forced to use an English centered cross-lingual space; resulting in a decrease in the quality of the cross-lingual space used due to the potential bad quality of mappings between typologically different languages, such as Uralic and Indo-European languages [15].

4.3.1 Definitions

Figure 4.1: Sample language tree representation simplified for illustration purposes [157].


s_{wo} World languages
$W_{s_{wo} \leftarrow s_{ge}}$
$W_{s_{wo} \leftarrow s_{ur}}$
s_{in} Indo-European
s_{ur} Uralic
$W_{s_{in} \leftarrow s_{ge}}$
$W_{s_{in} \leftarrow s_{it}}$
s_{ge} Germanic
s_{it} Italic
$W_{s_{ge} \leftarrow s_{du}}$
$W_{s_{ge} \leftarrow s_{en}}$
$W_{s_{ge} \leftarrow s_{de}}$
$W_{s_{it} \leftarrow s_{es}}$
$W_{s_{it} \leftarrow s_{pt}}$
$W_{s_{ur} \leftarrow s_{fi}}$
$W_{s_{ur} \leftarrow s_{et}}$
s_{du} Dutch
s_{en} English
s_{de} German
s_{es} Spanish
s_{pt} Portuguese
s_{fi} Finnish
s_{et} Estonian

Let $L = \{l_1, \ldots, l_{|L|}\}$ be a set of languages considered, $F = \{f_1, \ldots, f_{|F|}\}$ a set of language families, and $S = L \cup F = \{s_1, \ldots, s_{|F|+|L|}\}$ a set of possible language spaces. Let $X_l \in \mathbb{R}^{V_l \times d}$ be the set of word embeddings in language l, where V_l is the vocabulary of l and d is the number of dimensions of each embedding. Consider T is a language family tree (exemplified in Figure 4.1). The nodes in T represent language spaces in S, while each edge represents a transformation between the two nodes attached to it, i.e., $W_{s_a \leftarrow s_b} \in \mathbb{R}^{d \times d}$ refers to the transformation from space s_b to space s_a. For notation ease, we refer to $W_{s_a \overset{*}{\leftarrow} s_b}$ as the transformation that results from aggregating all transformations in the path from s_b to s_a, using the dot product:

$$W_{s_a \xleftarrow{*} s_b} = W_{s_a \leftarrow s_{t_1}} W_{s_{t_1} \leftarrow s_{t_2}} W_{s_{t_2} \leftarrow s_b} \tag{4.1}$$

where the path from s_a to s_b is $s_a, s_{t_1}, s_{t_2}, s_b$; s_{t_1} and s_{t_2} are intermediate spaces between s_a and s_b.

Finally, P is a set of bilingual lexicons, where $P_{l_1,l_2} \in \{0,1\}^{V_{l_1} \times V_{l_2}}$ is a bilingual lexicon with word pairs in languages l_1 and l_2. $P_{l_1,l_2}(i,j) = 1$ if the i^{th} word of V_{l_1} and the j^{th} word of V_{l_2} are aligned, $P_{l_1,l_2}(i,j) = 0$ otherwise.

Example. Consider the set of embeddings for English X_{en}, the transformation that converts embeddings in the English space to the Germanic language family space $W_{s_{ge} \xleftarrow{*} s_{en}}$, and the English embeddings transformed to the Germanic space $W_{s_{ge} \xleftarrow{*} s_{en}} X_{en}$. HCEG makes it so that $W_{s_{ge} \xleftarrow{*} s_{en}} X_{en}$ and $W_{s_{ge} \xleftarrow{*} s_{de}} X_{de}$ (the transformed embeddings of English and German) are in the same *Germanic* embedding space, while $W_{s_{in} \xleftarrow{*} s_{en}} X_{en}$ and $W_{s_{in} \xleftarrow{*} s_{es}} X_{es}$ (the transformed embeddings of English and Spanish) are in the same *Indo-european* embedding space.

In the rest of this section we describe HCEG in detail. Values given to each hyper-parameter mentioned in this section are defined in Section 4.4.4.

4.3.2 Embedding Normalization

When dealing with embeddings generated from different sources and languages it is important to normalize them. For doing so, HCEG follows a normalization sequence shown to be beneficial [17], which consists of length normalization, mean centering, and a second length normalization. The last length normalization allows computing cosine similarity between embeddings in a more efficient manner, simplifying the computation of cosine similarity to a dot product given that the embeddings are of unit-length.

4.3.3 Word Pairs

In order to generate a cross-lingual embedding space, HCEG requires a set P of aligned words across different languages. When using HCEG in a **supervised** way, P can be any existing resource consisting of bilingual lexicons, such as the ones described in Section 4.4.1. However, best advantage of the proposed strategy is taken when using **unsupervised** lexicon induction techniques, as they enable generating input lexicons for any pair of languages needed. Unlike TB/MP strategies that can only take advantage of signal that involves the pivot language, HCEG can use signal across all combinations of languages. For example, a TB/MP model where English is the pivot can only use lexicons comprised of English words. Instead, HCEG can exploit bilingual lexicons from other languages, such as *Spanish-Portuguese* or *Spanish-Dutch*, that if using the language tree in Figure 4.1 would reinforce the training of $W_{s_{it} \leftarrow s_{es}}$, $W_{s_{it} \leftarrow s_{pt}}$ and $W_{s_{it} \leftarrow s_{es}}$, $W_{s_{in} \leftarrow s_{it}}$, $W_{s_{in} \leftarrow s_{ge}}$, $W_{s_{ge} \leftarrow s_{du}}$, respectively.

When using HCEG in unsupervised mode, P needs to be automatically inferred. Yet, computing each $P_{l_1,l_2} \in P$ given two monolingual embedding matrices X_{l_1} and X_{l_2} is not a trivial task, as X_{l_1} and X_{l_2} are not aligned in vocabulary or dimension axes. Artetxe et al. [17] leverages the fact that the relative distances among words are maintained across languages [188], and thus propose using a language agnostic representation M_l for generating an initial alignment P_{l_1,l_2}:

$$M_l = sorted(X_l X_l^\top) \tag{4.2}$$

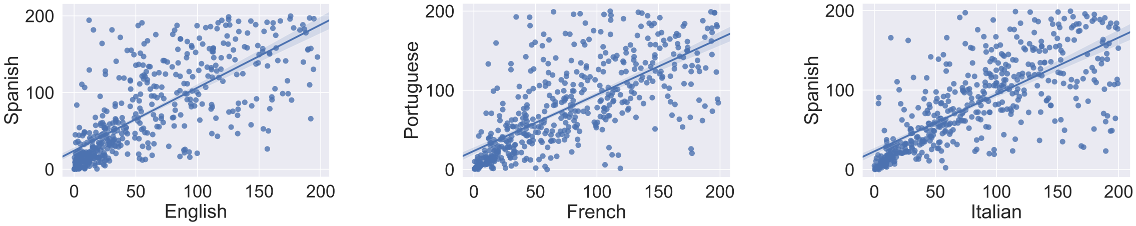
where given that X_l is length normalized, $X_l X_l^\top$ computes a matrix of dimensions $V_l \times V_l$ containing in each row the cosine similarities of the corresponding word embedding with respect to all other word embeddings. The values in each row are then *sorted* to generate a distribution representation of each word that in a ideal case where the isometry assumption holds perfectly would be language agnostic. Using the embedding representations M_{l_1} and M_{l_2}, P_{l_1,l_2} can be computed by assigning each word its most similar representation as its pair, i.e., $P_{l_1,l_2}(i,j) = 1$ if:

$$j = \arg\max_{1 \leq j \leq V_l} M_{l_1}(i,*) M_{l_2}(j,*)^\top \tag{4.3}$$

where $M_{l_1}(i,*)$ is the i^{th} row of M_{l_1} and $M_{l_2}(j,*)$ is the j^{th} row of M_{l_2}.

Artetxe et al. [17] showed that this assumption is strong enough to generate an initial alignment across languages. However, as we demonstrate in Section 4.3.3, the quality of this initial alignment is dependant on the languages used, making this initialization not applicable for languages that are typologically too distant from each other. To ensure a more robust initialization we enhance the strategy in [17] by introducing a new signal based on the frequency of use of words. Lin et al. [161] found that the top-2 most frequent words tend to be consistent across different languages. Motivated by this result, we measure to what extent does the frequency rankings of words correlate across languages. As shown in Figure 4.2, the word-frequency rankings are strongly correlated across languages, meaning that popular words tend to be popular regardless of the language. We exploit this behaviour in order to reduce the search space of Equation 4.3 as follows:

Figure 4.2: Distributions of word rankings across languages. The coordinates of each dot (representing a word pair) are determined by the position in the frequency ranking the word pair in each of the languages. Numbers are written in thousands. Scores computed using FastText embedding rankings [107] and MUSE cross-lingual pairs [48]. Pearson's correlation (ρ) computed using the full set of word pairs, figures generated using a random sample of 500 word pairs for illustration purposes.



(a) $\rho = 0.684$ (b) $\rho = 0.709$ (c) $\rho = 0.737$

$$j = \arg \max_{j-t \leq j \leq j+t} M_{l_1}(i,*) M_{l_2}(j,*)^\top \tag{4.4}$$

where t is a value used to determine the search window. Note that we assume the embeddings in any matrix X_l are sorted in ascending order of frequency, i.e., the embedding in the first row represents the most frequent word of language l. Apart from improving the overall quality of the inferred lexicons (see Section 4.5.1), incorporating a frequency ranking based search as part of the initialization reduces the computation time needed as the search space is considerably reduced.

4.3.4 Objective Function

Unlike traditional objective functions that optimize a transformation matrix for two languages at a time, the goal of HCEG is to simultaneously optimize the set of all transformation matrices W, such that the loss function $\mathcal{L}$ is minimized:

$$\arg\min_{W} \mathcal{L} \tag{4.5}$$

$\mathcal{L}$ is a linear combination of three different losses:

$$\mathcal{L} = \beta_1 \times \mathcal{L}_{align} + \beta_2 \times \mathcal{L}_{orth} + \beta_3 \times \mathcal{L}_{reg} \tag{4.6}$$

where $\mathcal{L}_{align}$, $\mathcal{L}_{orth}$, $\mathcal{L}_{reg}$, represent the alignment, orthogonality and regularization losses, and β_1, β_2, β_3 are their weights.

$\mathcal{L}_{align}$ gauges the extent to which training word pairs align. This is done by computing the sum of the cosine similarity among all word pairs in P:

$$\begin{aligned} \mathcal{L}_{align} = - \sum_{P_{l_1,l_2} \in P} P_{l_1,l_2}(W_{s_{\widehat{l_1,l_2}} \xleftarrow{*} s_{l1}} X_{l_1} \cdot \\ W_{s_{\widehat{l_1,l_2}} \xleftarrow{*} s_{l2}} X_{l_2}) \end{aligned} \tag{4.7}$$

where $s_{\widehat{l_1,l_2}}$ refers to the space in the lowest common parent node for s_{l_1} and s_{l_2} in T (e.g., $s_{\widehat{es,en}} = s_{in}$ in Figure 4.1). We found that using $s_{\widehat{l_1,l_2}}$ instead of the space in the root node of T improves the overall performance of HCEG, apart from reducing the time taken for training (see Section 4.5.3).

Several researchers have found beneficial to enforce orthogonality in the transformation matrices W [259, 14, 223]. This constraint ensures that the original quality of the embeddings is not degraded when transforming them to a cross-lingual space. For this reason, we incorporate an orthogonality constraint $\mathcal{L}_{orth}$ into our loss function in Equation 4.8, with I being the identity matrix.

$$\mathcal{L}_{orth} = \sum_{W_{s_1 \leftarrow s_2} \in W} ||I - W_{s_1 \leftarrow s_2} W^{\top}_{s_1 \leftarrow s_2}|| \tag{4.8}$$

We also find beneficial to include a regularization term in $\mathcal{L}$:

$$\mathcal{L}_{reg} = \sum_{W_{s_1 \leftarrow s_2} \in W} ||W_{s_1 \leftarrow s_2}||_2 \tag{4.9}$$

4.3.5 Learning the Parameters

HCEG utilizes stochastic gradient descent for tuning the parameters in W with respect to the training word pairs in P. In each iteration, $\mathcal{L}$ is computed and backtracked in order to tune each transformation matrix in W such that $\mathcal{L}$ is minimized. Batching is used to reduce the computational load in each iteration. A batch of word pairs $\hat{P}$ is sampled from P by randomly selecting α_{lpairs} language pairs as well as α_{wpairs} word pairs in each $\hat{P}_{l_1,l_2} \in \hat{P}$, e.g., a batch might consist of 10 $\hat{P}_{l_1,l_2}$ matrices each containing 500 aligned words.

Iterations are grouped into epochs of α_{iter} iterations at the end of which $\mathcal{L}$ is computed for the whole P. We take a conservative approach as convergence criterion. If no improvement is found in $\mathcal{L}$ in the last α_{conv} epochs, the training loop stops.

We achieve best convergence time initializing each $W_{s_1 \leftarrow s_2} \in W$ to be orthogonal. We tried several methods for orthogonal initialization, such as simply initializing to the identity matrix. However, we obtained most consistent results using the random semi-orthogonal initialization introduced by Saxe et al. [216].

4.3.6 Iterative Refinement

As shown by Artetxe et al. [15], the initial lexicon P is iteratively improved by using the generated cross-lingual space for inferring a new lexicon P' at the end of each learning phase described in 4.3.5. More specifically, when computing each $P'_{l_1,l_2} \in P'$, $P'_{l_1,l_2}(i,j)$ is 1 (0 otherwise) if

$$\begin{aligned} j = \arg\max_{j} \ & W_{s_{\widehat{l_1,l_2}} \overset{*}{\leftarrow} s_{l1}} X_{l_1}(i,*) \cdot \\ & (W_{s_{\widehat{l_1,l_2}} \overset{*}{\leftarrow} s_{l2}} X_{l_2}(j,*))^\top \end{aligned} \tag{4.10}$$

Potentially, any new bilingual lexicon P'_{l_1,l_2} can be inferred and included in P' at the end of each learning phase. However, as the cardinally of L grows, this process can take a prohibitive amount of time given combinatorial explosion. Therefore, in practice, we only infer P'_{l_1,l_2} following a criterion intended to maximize lexicon quality. P'_{l_1,l_2} is inferred for languages l_1 and l_2 only if l_1 and l_2 are siblings in T (they share the same parent node) or l_1 and l_2 are the best representatives of their corresponding family. A language is deemed the *best representative* of its family if it is the most frequently-spoken[2] language in its subtree. For example, in Figure 4.1, Spanish is the *best* representative for the Italic family, but not for Indo-European, for which English is used.

The set criterion not only reduces the amount of time required to infer P' but also improves overall HCEG performance. This is due to a better utilization of the hierarchical characteristics of our cross-lingual space, only inferring bilingual lexicons from typologically-related languages or their best representatives in terms of resource quality.

[2] Based on numbers reported by Lewis and Gary [157].

4.3.7 Retrieval criterion

As discussed in Section 4.2, one of the issues effecting Nearest-Neighbour retrieval is hubness Dinu et al. [58], where certain words are in the surrounding of an abnormally large amount of other words, causing the Nearest-Neighbour algorithm to incorrectly prioritize hub words. To address this issue, we use Cross-domain Similarity Local Scaling (CSLS) [48] as the retrieval algorithm during both training and prediction time. CSLS is a rectification for Nearest-Neighbour retrieval that avoids hubness by counterbalancing the cosine similarity between two embeddings by a factor consisting of the average similarity of each embeddings with its k closest neighbours. Following the criteria in [48], we set the number of neighbours used by CSLS to $k = 10$.

4.4 Evaluation Framework

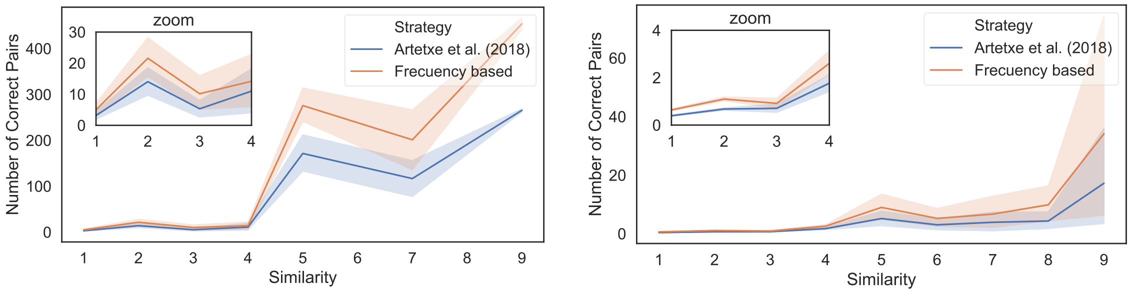
We describe below the evaluation set up used for conducting the experiments presented in Section 4.5.

4.4.1 Word Pair Datasets

Dinu-Artetxe. The Dinu-Artetxe dataset, presented by Dinu et al. [58] and enhanced by Artetxe et al. [14], is the most widely used benchmark for evaluating cross-lingual embeddings. It is comprised of English-centered bilingual lexicons for Italian, Spanish, German, and Finnish.

MUSE. The MUSE dataset [48] contains bilingual lexicons for all combinations of German, English, Spanish, French, Italian, and Portuguese. In addition, it includes word

Figure 4.3: Number of correct word pairs inferred using the unsupervised initialization technique presented by *Artetxe et al. [17]* and the *Frequency based* technique described in Section 4.3.3.



(a) MUSE dataset

(b) Panlex dataset

pairs for 44 languages with respect to English.

Panlex. Dinu-Artetxe and MUSE are both English centered datasets, given that most (if not all) of their word pairs have English as their source or target language. This makes the datasets suboptimal for our purpose of generating and evaluating a non-language centered cross-lingual space. For this reason, we generated a dataset using Panlex [130], a panlingual lexical database. This dataset (made public in our repository) includes bilingual lexicons for all combinations of 157 languages FastText is available, totalling 24,492 bilingual lexicons. Each of the lexicons was generated by randomly sampling 5k words from the top-200k words in the embedding set for the source language, and translating them to the target language using the Panlex database.

4.4.2 Language Selection and Family Tree

As previously stated, we aim to generate a single cross-lingual space for as many languages as possible. We started with the 157 languages for which FastText embeddings are available

[107]. We then removed languages that did not meet both of the following criteria: 1) there must exist a bilingual lexicon with at least 500 word pairs for the language in any of the datasets described in Section 4.4.1, and 2) the embedding set provided by FastText must contain at least 20k words. The first criterion is a minimal condition for evaluation, while the second one is necessary for the unsupervised initialization strategy. The criteria is met by languages, which are the ones used in our experiments. Their corresponding ISO-639 codes can be seen in Table 4.3.We use the language family tree defined by Lewis and Gary [157].

4.4.3 Framework

For experimental purposes, each dataset described in Section 4.4.1 is split into training and testing sets. We use the original train-test splits for Dinu-Artetxe and MUSE. For Panlex, we generate a split randomly sampling word pairs–keeping 80% for the training and the remaining 20% for testing. For development and parameter tuning purposes, we use a disjoint set of word pairs specifically created for this purpose based on the Panlex lexical database. This development set contains 10 different languages with varied popularity. None of the word pairs present in this development set are part of either the train or test sets.

4.4.4 Hyper-Parameters

The following hyper-parameters were manually tuned using the development set described in Section 4.4.3: $\beta_1 = 0.98$, $\beta_2 = 0.01$, $\beta_3 = 0.01$, $t = 1000$, $\alpha_{lpairs} = 128$, $\alpha_{wpairs} = 2048$, $\alpha_{iter} = 5000$, $\alpha_{conv} = 25$.

4.5 Evaluation

We discuss below the results of the study conducted over languages to assess HCEG.

4.5.1 Unsupervised Initialization

We first evaluate the performance of the unsupervised initialisation strategy described in Section 4.3.3, and compare it to the state-of-the-art strategy proposed by Artetxe et al. [17]. In this case, we run both initialisation strategies using the top-20k FastText embeddings [107] for all pairwise combinations of the languages we study. For each language pair, we measure how many of the inferred word pairs are present in the corresponding lexicons in the MUSE and Panlex datasets. For MUSE, our proposed initialization strategy (*Frequency based*) obtains an average of 48.09 correct pairs, an improvement with respect to the 29.62 obtained by the strategy proposed by Artetxe et al. [17]. For Panlex, the respective average correct pair counts are 1.05 and 0.55. Both differences are statistically significant ($p < 0.01$) using a paired T-test. The noticeable difference across datasets is due to how the sampling was done for generating the datasets: MUSE contains a considerably higher amount of frequent words in comparison to Panlex, making the latter a relatively harder dataset for vocabulary induction. In Figure 4.3 we illustrate the results of each strategy grouped by language-pair similarity. This similarity is based on the number of common parents the two languages share. For example, in Figure 4.1, Spanish has a similarity of 3, 2, and 1 with Portuguese, English, and Finnish, respectively. As we see in Figure 4.3, similarity is a factor that strongly determines the quality of the alignment generated by the unsupervised initialization. Even if this phenomenon affects both analyzed strategies, our proposed frequency based initialization strategy consistently obtains a few more correct word pairs for the least similar language pairs, which as we show in Table 4.4 are key for generating a correct mapping for those languages.

4.5.2 State-of-the-Art Comparison

In order to contextualize the performance of HCEG with respect to the state-of-the-art (listed in Tables 4.1 and 4.2), we measure vocabulary induction accuracy. We report results for both the supervised (HCEG-S) and unsupervised (HCEG-U) versions of HCEG when applicable. In the supervised mode, we train one single model per dataset using all the training word pairs available. We then use this model for computing all pairwise scores. In the unsupervised mode, we train a single model regardless of the dataset used for testing. We found unfair to train a supervised model using the Dinu-Artetxe dataset given that it only contains 4 bilingual lexicons, not enough for training our tree structure. Thus, only unsupervised results are shown for that dataset.

As shown in Table 4.1 the unsupervised version of HCEG achieves, in most cases, the best performance among all the strategies, even improving over state-of-the-art supervised models. The improvement is most noticeable for Finnish and Spanish, where HCEG-U obtains an improvement of 2 and 5 points, respectively. A similar behaviour can be seen in Table 4.2, where we describe the results on the MUSE dataset. Spanish, alongside with Catalan, Italian, and Portuguese, obtain a substantially larger improvement compared to other languages. We attribute this to the fact that Spanish is the second most resourceful language in terms of corpora after English. This makes the quality of Spanish word embeddings comparably better to other languages, which as a result improves the mapping quality of typologically-related languages, such as Portuguese, Italian, or Catalan.

	Method	en-it	en-de	en-fi	en-es
Supervised	Mikolov (2013b)	34.93*	35.00*	25.91*	27.73*
	Faruqui (2014)	38.40*	37.13*	27.60*	26.80*
	Shigeto (2015)	41.53*	43.07*	31.04*	33.73*
	Dinu (2014)	37.7	38.93*	29.14*	30.40*
	Lazaridou (2015)	40.2	-	-	-
	Xing (2015)	36.87*	41.27*	28.23*	31.20*
	Zhang (2016)	36.73*	40.80*	28.16*	31.07*
	Artetxe (2016)	39.27	41.87*	30.62*	31.40*
	Artetxe (2017)	39.67	40.87	28.72	-
	Smith (2017)	43.1	43.33*	29.42*	35.13*
	Artetxe (2018a)	45.27	44.13	32.94	36.60
	Jouling (2018)	45.5	-	-	-
Semi.	Artetxe (2017) 25	37.27	39.60	28.16	-
	Smith (2017) cog	39.9	-	-	-
	Artetxe (2017), num	39.40	40.27	26.47	-
Unsupervised	Zhang (2017), $\lambda = 1$	0.00*	0.00*	0.00*	0.00*
	Zhang (2017) $\lambda = 10$	0.00*	0.00*	0.01*	0.01*
	Conneau (2017) code	45.15*	46.83*	0.38*	35.38*
	Conneau (2017) paper	45.1	0.01*	0.01*	35.44*
	Artetxe (2018)	48.13	**48.19**	32.63	37.33
	HCEG-U	**49.02**	48.18	**34.82**	**42.15**

Table 4.1: Results using the Dinu-Artetxe dataset. Scores marked with (*) were reported by Artetxe et al. [17]. All the other scores were reported in the original papers.

	Conneau (2017)	Joulin (2018)	Artetxe (2018)	HCEG-S	HCEG-U
bg	57.5	63.9	65.8	64.1	**67.5**
ca	70.9	73.8	76.3	73.1	**77.7**
cs	64.5	68.2	70.2	68.2	**71.7**
da	67.4	71.1	70.3	68.8	**72.7**
de	72.7	76.9	**79.1**	75.8	79.0
el	58.5	62.7	67.8	65.3	**68.5**
es	83.5	86.4	88.6	86.8	**90.4**
et	45.7	49.5	55.8	53.5	**57.3**
fi	59.5	65.8	68.1	65.2	**68.3**
fr	82.4	84.7	87.6	85.4	**88.3**
he	54.1	57.8	61.1	59.5	**63.0**
hr	52.2	55.6	57.6	54.8	**58.2**
hu	64.9	69.3	69.6	66.8	**70.1**
id	67.9	69.7	75.5	73.2	**75.6**
it	77.9	81.5	83.3	81.3	**85.6**
mk	54.6	59.9	63.5	62.3	**64.9**
nl	75.3	79.7	79.9	79.4	**81.9**
no	67.4	71.2	69.9	69.5	**71.9**
pl	66.9	70.5	72.0	70.7	**72.8**
pt	80.3	82.9	85.5	83.8	**87.8**
ro	68.1	74.0	75.4	72.8	**76.0**
ru	63.7	67.1	69.5	68.1	**69.8**
sk	55.3	59.0	62.0	59.6	**62.4**
sl	50.4	54.2	60.1	57.7	**61.1**
sv	60.0	63.7	66.2	65.0	**68.0**
tr	59.2	61.9	68.7	66.3	**70.0**
uk	49.3	51.5	**56.4**	53.8	**56.4**
vi	55.8	55.8	3.9	55.5	**58.3**
Avg.	63.8	67.4	68.2	68.1	**71.2**

Table 4.2: Results on the MUSE dataset. Scores of Artetxe et al. [17] were obtained using the code share by the authors. All other scores were reported by Joulin et al. [129].

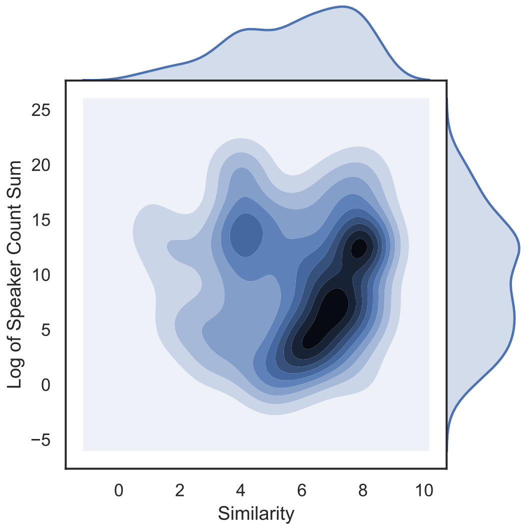
The importance of explicitly considering topological connections among languages to enhance mappings gets more evident when analyzing the data in Table 4.3. Here we include the pairing that yielded the best and worst mapping for each language, as well as the position of English in the quality ranking. English and Spanish have a strong quality mapping with respect to each other, being Spanish the language that English obtains the best mapping with, while English is the second best mapped language for Spanish. Additionally, Spanish is the language with which Italian, Portuguese, and Catalan obtain the best mapping quality. On the other side of the spectrum, the worst mappings are dominated by two languages, Georgian and Vietnamise, with 40 languages having these two language as worst. Followed by Maltese, Albanian, and Finnish, with 8 occurrences each. This is not unexpected, as these languages are relatively isolated in the language family tree, and also have a low number of speakers. We also see that English is usually on the top side of the ranking for most languages. For languages that are completely isolated, such as Basque and Yoruba, English tends to be their best mapped language. From this we surmise that when typological relations are lacking, quality of the embedding space is the only aspect the mapping strategy can rely on.

L	B,W,E	L	B,W,E	L	B,W,E	L	B,W,E	L	B,W,E	L	B,W,E	L	B,W,E
af	nl,fi,4	ceb	tl,li,22	ga	gd,tt,12	jv	id,scn,34	my	zh,mk,19	sco	en,mt,1	tr	tk,ka,13
als	en,vi,1	ckb	tg,tr,19	gd	ga,vi,2	ka	en,bs,1	nds	nl,vi,3	sd	bn,tl,5	tt	ba,sa,9
am	arz,de,80	cs	sk,vi,12	gl	pt,ka,16	kk	ky,vi,51	nl	af,ka,4	si	dv,ka,5	ug	tr,vls,4
an	es,ka,17	cv	tr,sq,2	gom	mr,fi,10	km	vi,nl,4	no	sv,vi,3	sk	cs,vi,5	uk	ru,fi,19
arz	mt,ja,3	cy	br,fi,2	gu	pa,ka,3	kn	ta,lt,55	oc	es,my,3	sl	sr,vi,6	ur	hi,eo,10
as	bn,vi,4	da	sv,fi,4	he	arz,mk,10	ko	en,af,1	pa	gu,vi,6	so	arz,sq,73	vec	pms,tr,2
ast	es,ja,20	de	lb,mt,5	hi	ur,ka,5	ky	kk,af,17	pam	id,sr,18	sq	en,tt,1	vi	km,vls,3
ba	tt,sq,34	dv	si,ka,3	hr	sr,tt,5	la	es,mt,3	pl	cs,vi,4	sr	hr,vi,4	vls	nl,eo,8
bar	de,fi,6	el	en,eo,1	hsb	pl,am,3	lb	de,ka,2	pms	vec,sah,7	su	id,mk,37	wa	fr,fi,7
be	ru,vi,4	en	es,gv,-	hu	fi,ckb,9	li	nl,ka,7	pt	es,mt,5	sv	da,vi,5	yo	en,lt,1
bg	mk,ka,9	eo	en,sq,1	hy	en,fi,1	lt	ru,mt,5	qu	en,bn,1	ta	ml,mt,3	zh	my,de,10
bn	as,vi,6	es	pt,vi,2	id	jv,vi,3	mg	id,sq,44	ro	es,vi,6	te	ta,mk,15		
br	cy,ka,18	eu	en,lt,1	ilo	id,sq,6	mk	bg,vi,4	ru	uk,su,20	tg	ckb,ka,13		
bs	sr,ka,2	fi	hu,als,24	is	sv,ka,3	ml	ta,sq,29	sa	hi,ka,2	th	en,vls,1		
ca	es,mt,5	fr	it,vi,5	it	es,mt,5	mr	si,ka,21	sah	tr,ka,2	tk	tr,lt,7		
ce	en,sq,1	fy	en,eo,1	ja	en,vi,1	mt	arz,tt,70	scn	it,ka,21	tl	ceb,ru,47		

Table 4.3: Best (B), worst (W) and English mapping ranking (E) for each language (L).

Given space constraints, we cannot show the vocabulary induction scores for the 24,492 language pairs in the Panlex dataset. Instead, we group the results using two variables: the sum of number of speakers for each of the two languages, and the minimum similarity (as defined in Section 4.5.1) for each language with respect to English. We rely on these

Figure 4.4: Improvement over the strategy proposed by Artetxe et al. [17] in Panlex, in terms of language similarity and number of speakers. Darker denotes larger improvement.



variables for grouping purposes as they align with two of our objectives for designing HCEG: 1) remove the bias towards the pivot language (English), and 2) improve the performance of low-resource languages by taking advantage of typologically similar languages.

Figure 4.4 captures the improvement of HCEG-U over the strategy in [17] (the best-performing benchmark), grouped by the aforementioned variables. We excluded Hindi and Chinese from the figure, as they made any pattern hard to observe given their high number of speakers. The sum of number of speakers axis was also logarithmically scaled to facilitate visualization. The figure captures an evident trend in the similarity axis. The lower the similarity of the language with respect to English, the higher the improvement achieved by HCEG-U. This can be attributed to the manner in which TB/MP models generated the space using English as primary resource, hindering the potential quality of languages that are distant from it. Additionally, we see a less-prominent but existing trend in the speaker

	Description	Dinu-Artetxe	MUSE	Panlex
Supervised	¬Hierarchy	-	66.7	32.0
	¬Orthogonal Init.	-	67.8	36.5
	¬Iterative Refinement	-	65.4	35.1
	All vs All Inference	-	66.3	36.6
	World langs. as root	-	67.5	35.7
	HCEG-S	-	**68.1**	**37.3**
Unsupervised	¬Hierarchy	40.2	67.9	28.1
	¬Orthogonal Init.	43.2	71.0	34.7
	¬Iterative Refinement	0.09	0.08	0.02
	All vs All Inference	39.3	69.4	34.6
	World langs. as root	42.8	70.2	33.8
	¬Freq. based Init.	41.2	68.0	31.1
	HCEG-U	**43.5**	**71.2**	**35.8**

Table 4.4: Ablation study.

sum axis. Despite some exceptions, HCEG-U obtains higher differences with respect to Artetxe et al. [17] the less spoken a language is. A behaviour that is similar in essence to a Pareto front can also be depicted from the figure. Even if both variables contribute to the difference in improvement of HCEG-U, one variable needs to compensate for the other in order to maximize accuracy. In other words, the improvement is higher the less speakers the language pair has or the more distant the two languages are from English, but when both variables go to the extreme, the improvement decreases. The aforementioned trends serve as evidence that the hierarchical structure is indeed important when building a cross-lingual space that considers typologically diverse languages, validating our premises for designing HCEG.

4.5.3 Ablation study

In order to assess the validity of each functionality included as part of HCEG, we conducted an ablation study. We summarize the results of this study in Table 4.4, where the symbol

$\neg$ indicates that the subsequent feature is ablated in the model. For example, $\neg$Hierarchy indicates that the Hierarchy structure is removed, replacing it by a structure where each language needs just one transformation matrix to reach the *World languages* space.

As indicated by the ablation results, the hierarchical structure is indeed a key part of HCEG, considerably reducing its performance when removed, and having its strongest effect in the dataset with the highest number of languages, i.e., Panlex. It is also noticeable the importance of the Iterative Refinement strategy, making the unsupervised version of HCEG useless when removed. The Frequency-based initialization is also a characteristic that considerably improves the results of HCEG-U. Looking deeper into the data, we found 2,198 language pairs (about 9% of all pairs) that obtained a vocabulary induction accuracy close to 0 (<0.05) without using this initialization, but were able to produce enough signal to yield more substantial accuracy values (>10.0) when using the Frequency based initialization. Finally, the design decisions that we initially took for reducing training time–(i) the orthogonal initialization, (ii) the heuristic based inference, and (iii) using the lowest common root for computing the loss function–also have a positive effect on the performance of the HCEG.

4.6 Conclusion and Future Work

In this chapter, we have introduced HCEG, a cross-lingual space learning strategy that does not depend on a pivot language, as instead, it takes advantage of the natural hierarchy existing among languages. Results from extensive studies on languages demonstrate that the proposed strategy outperforms existing cross-lingual space generation techniques, in terms of vocabulary induction, for both popular and not so popular languages. HCEG improves the mapping quality of many low-resource languages. Yet, we noticed this improvement mostly happens when a language has more typologically-related counterparts. Therefore, as future work, we intend to investigate other techniques that can help improve the quality of mapping for typologically-isolated low-resource languages. Additionally, we plan to explore a closed form solution for HCEG, in order to reduce its training time.

CHAPTER 5

A FRAMEWORK FOR HIERARCHICAL MULTILINGUAL MACHINE TRANSLATION

Abstract

Multilingual machine translation has recently been in vogue given its potential for improving machine translation performance for low-resource languages via transfer learning. Empirical examinations demonstrating the success of existing multilingual machine translation strategies, however, are limited to experiments in specific language groups. In this paper, we present a hierarchical framework for building multilingual machine translation strategies that takes advantage of a typological language family tree for enabling transfer among similar languages while avoiding the negative effects that result from incorporating languages that are too different to each other. Exhaustive experimentation on a dataset with 41 languages demonstrates the validity of the proposed framework, especially when it comes to improving the performance of low-resource languages via the use of typologically related families for which richer sets of resources are available.

5.1 Introduction

The explosive growth of text-based Web resources has been well-documented [261]. This, in theory, translates into a wealth of resources that are available to the masses. In practice, however, these resources are off-limit for populations that cannot read and understand the language in which each resource was originally written. Consequently, valuable resources written in popularly-spoken languages are out of reach for individuals who speak less-common ones, whereas resources written in minority languages rarely find their way into larger populations. Indeed, translators can fill this gap, but it is unfeasible for them to manually take ownership of this laborious and costly work on a large scale. This evidenced the need for machine translation: the task of taking a text in one language and translating it to another language in an automatic fashion [230].

While as an area of study machine translation exists since the 1950s [252], it was not till the 1990s-early 2000s when statistical approaches for machine translation showed prominence [34, 146]. It took some time for computing capabilities and text availability to converge into a new era of high-quality machine translation based on neural networks [21, 166]. Nowadays, research pertaining to machine translation is categorized in three groups: rule based strategies that rely on hand written translation rules [89], statistical techniques that learn rules based on parallel corpora [146], and neural machine translation strategies based on a encoder-decoder architecture [230], being the latter the most popular nowadays given its performance.

Most machine translation strategies build an encoder architecture that maps a text sequence in a source language to a vector representation and a decoder architecture that maps the vector representation to the same text sequence but in the target language, reducing the machine translation task to a purely bilingual task. The performance of this bilingual task is conditioned by two main factors: (1) quality/quantity of available parallel corpora, and (2) the similarity between the source and target language, defined in terms of the amount of linguistic patterns that both languages have in common. The more and better quality corpora it is available between the two languages, the better the translation quality it is expected to be. This is the reason why low-resource languages that tend to have less corpora available yield worse translation models. Additionally, the more similar two languages are the better the translation will be. It is not the same to translate from Spanish to Portuguese, two languages that are closely related as they follow similar linguistic patterns, as it is to translate from English to Chinese, that share little to no similarity in lexical or grammatical patterns.

Several approaches have been proposed in an attempt to address the two aforementioned problems, among which the translation via triangulation framework is the most prominent [44, 99]. In this framework, a translation is decomposed into multiple sub-translations in order to maximize corpora quality/quantity in each of the sub-translations in order to improve the performance of the final translation. As an example, instead of translating from Portuguese to Catalan, which might have reduced corpora available, a translation is first done from Portuguese to Spanish and then from Spanish to Catalan, improving the final translation performance given that both language pairs used (Portuguese-Spanish and Spanish-Catalan) have a considerable larger amount of parallel corpora available. This framework, however, has its own drawbacks, as a higher of sub-translations means a higher computational cost and a more prominent cumulative error (introduced at each sub-translation level).

Multilingual machine translation, derived from multi-task training techniques, is a more recent framework that intends to address the corpora availability problem. In this case the task of machine translation is no longer considered as a bilingual task, but as a multilingual task where multiple source and target languages can be simultaneously considered [166]. The objective of multilingual machine translation is to take advantage of knowledge in language pairs with large corpora availability and transfer it to lower resourced pairs by training them as part of the same model. For example, a single model can be trained to translate from Spanish to English and Catalan to English, with the expectancy that the performance of Catalan-English translations will get improved given that it has been trained together with a language with richer resources like Spanish. Examples of multilingual machine translation models include those based on strategies that use a single encoder for all languages but multiple decoders [60], or strategies that treat all languages as part of a single unified encoder-decoder structure [112].

Even if existing multilingual machine translation strategies achieve language transfer to a degree, this transference only takes place when using specific language sets. Furthermore, these strategies ignore possible negative side-effects of including languages that are considerably different into a single model, i.e., training languages like Catalan and Spanish might be beneficial for performance, however, including a distant language like Chinese might decrease the overall performance of the same model. As a result, a state-of-the-art model such as the one described by Ha et al. [112] that includes all languages as part of a unified encoder-decoder structure would be sub-optimal when including language groups with strong differences. Madrazo Azpiazu and Pera [171] observed a similar behavior in the area of cross-lingual word embedding and concluded that putting all languages into a single space could act in detriment of the general model if it is not done in an organized fashion.

Inspired by the idea of building a single model that can translate from multiple to multiple languages [112] and the need of organization of languages when building multilingual strategies [171], we propose a Hierarchical Framework for Neural Machine Translation (HNMT). HNMT is a multilingual machine translation encoder-decoder framework that explicitly considers the inherent hierarchical structure in languages. For doing so, HNMT exploits a typological language family tree, which is a hierarchical representation of languages organized by their linguistic similarity, in terms of grammar, vocabulary, and syntax, to name a few. In other words, HNMT follows this natural connection among languages to encode and decode word sequences, in our case sentences. The hierarchical nature of languages allows HNMT to only combine knowledge across languages with similar nature, while avoiding any negative knowledge transfer across distant languages.

The main contributions of this work include:

- A novel hierarchical encoder-decoder framework that can be applied to any of the popular state-of-the-at machine translation strategies to improve translation performance for low-resource languages.

- A comprehensive evaluation over 41 languages and 758 tasks to examine the extent to which language transfer is achieved.

- An analysis of the implications emerging from using the proposed framework for machine translation of low-resource languages.

5.2 Related Work

Machine translation techniques have been built using a variety of strategies, including rule-based systems [89], statistical machine translation [146], or neural machine translation strategies [230]. In this work, we dedicate research efforts to neural machine translation strategies (NMT). More specifically, we focus on the enhancement of encoder-decoder strategies from a multilingual perspective. For this reason, we describe below related literature in the area of NMT and multilingual approaches for NMT.

Encoder-decoders strategies for NMT. Encoder-decoder strategies were first proposed by Sutskever et al. [230] as a solution for the inability of traditional neural networks for learning sequence-to-sequence mappings. This strategy was soon found lacking when translating long sentences given its need to compress all the sentence information into a low dimensional vector [42]. Several researchers tried to address this problem by allowing the decoder to have access to a larger amount of information, such as the previously generated word and the encoded sentence at any time step [42] or to the whole set of hidden states produced by the decoder via an attention mechanism [21]. In order to obtain further training speed and translation quality, approaches presented later on tried remove the recurrent layers of the models, known to hinder parallelization of models. With this purpose in mind, Gehring et al. [96] proposed a model based on Convolutional Neural Networks, while Vaswani et al. [246] focused on just using layers purely based on attention.

Multilingual NMT. Multilingual NMT strategies can be categorized by the degree to which they can share part of the architecture across different languages. Dong et al. [60] use a single encoder regardless of the language and rely on separate decoders for translation. Luong et al. [166] introduce a strategy that uses one single encoder and decoder per language among all translation pairs. Firat et al. [86] maintain the different encoders and decoders but share the attention mechanism across all translation pairs. Ha et al. [112] propose to use one universal encoder and decoder that can handle any source and target language. This is achieved by providing the model with information of the language as an embedded parameter.

Even if existing Multilingual NMT models can obtain varied ranges of transfer learning across languages, none of the strategies we discussed takes advantage of the inherent hierarchical structure of the languages, that can beneficial to generate a more reliable language transfer among typologically similar languages, avoiding hindering the performance across distant languages.

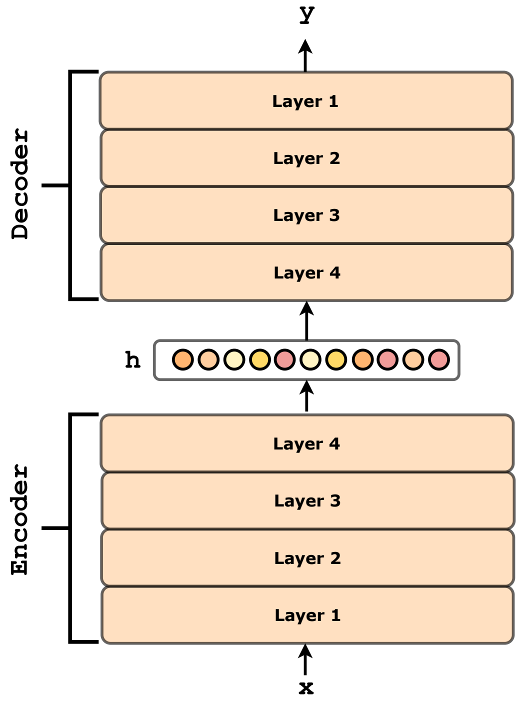
5.3 Method

In this section, we describe the proposed framework for hierarchical multilingual machine translation, i.e., HNMT. We first present a general sequence-to-sequence architecture used for neural machine translation, which we illustrate in Figure 5.1. Then we explain how this general structure can be extended for multilingual machine translation. Lastly, we describe our proposed hierarchical framework illustrated in Figure 5.2.

5.3.1 Neural Machine Translation

State-of-the-art neural machine translation takes advantage of sequence-to-sequence models for translating a sequence x (usually a sentence) in the source language L_s to a sequence y in the target language L_t. For doing so, the model is generally separated into an encoder

Figure 5.1: General bilingual machine translation architecture with 4 layers in both the encoder and the decoder.



module (ENC_{L_s}) capable for encoding x into a vector representation h of size $|h|$ and a decoder (DEC_{L_t}) that aims at generating y from h.

Both the encoder and the decoder modules contain an equal amount of N repeated layers that are used in a sequential way, as illustrated in Figure 5.1. Starting from an input representation x each encoding layer $ENC_{L_s}(i)$, is responsible for taking the representation $h_{enc_{i-1}}$ and generating h_{enc_i}, until it produces h. Once h is generated each decoder layer $DEC_{L_s}(i)$, will take h_{dec_i} and generate $h_{dec_{i-1}}$ until y is generated. Following our naming convention: $h_{enc_0} = x$, $h_{enc_N} = h = h_{dec_N}$, and $h_{dec_0} = y$. If the model consists of 3 layers ($N = 3$), the process of translating x to y requires $N * 2 = 6$ steps:

$$\begin{aligned} x \xrightarrow{ENC_{L_s}(1)} h_{enc_1} \xrightarrow{ENC_{L_s}(2)} h_{enc_2} \xrightarrow{ENC_{L_s}(3)} h \\ h \xrightarrow{DEC_{L_s}(3)} h_{dec_2} \xrightarrow{DEC_{L_s}(2)} h_{dec_1} \xrightarrow{DEC_{L_s}(1)} y \end{aligned} \tag{5.1}$$

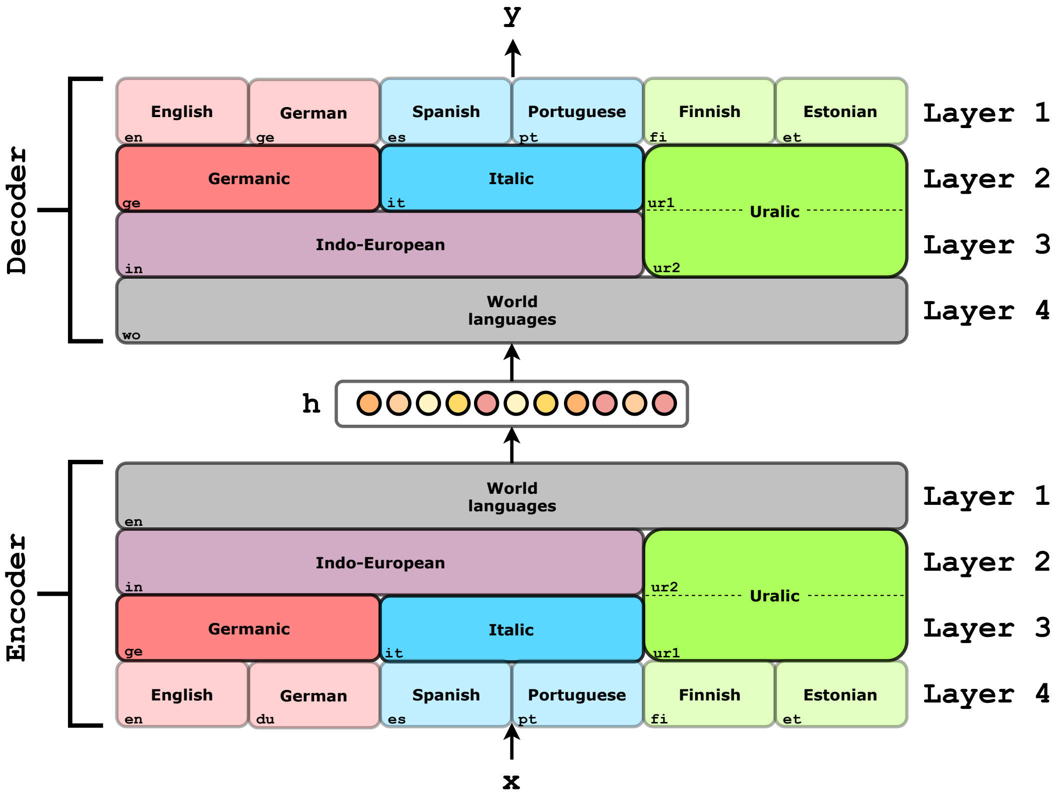
Architectures for building each of the layers $ENC_{L_s}(i)$ and $DECL_s(i)$ are manifold in the literature, being recurrent neural networks [121], convolutional neural networks [96], and transformers [246] the most widely accepted approaches. As previously stated, our proposed strategy is designed so that it can be applied to all of these architectures. However, for simplicity, we only showcase and discuss the practical application of HNMT on a single architecture (see Section 5.4). We use a Long Short Term Memory (LSTM) recurrent neural network [121] given that it is an architecture with well-studied benefits and limitations. This enables us to isolate any phenomena introduced by this architecture from our framework so that we really focus our analysis on the advantages and disadvantages of our framework on its own.

5.3.2 Multilingual Machine Translation

In traditional (bilingual) neural machine translation, encoder and decoder modules are language specific, as captured by the subscripts in ENC_{L_s} and DEC_{L_t}. This means that an encoder for English ($ENC_{L_{en}}$) can be neither substituted by an encoder for Spanish ($ENC_{L_{es}}$) nor used to encode any x that is not in English. Additionally, there is no guarantee that the representation h is equivalent in any translation task, i.e., the representation h that $ENC_{L_{en}}$ generates after being trained for English-Spanish translation is different from the representation generated by $ENC_{L_{en}}$ for English-Portuguese. Even inverse translation tasks, e.g., English-Spanish and Spanish-English, are considered to be separate tasks as there is no knowledge sharing across the two translations, resulting in different performance for each of the two. This poses a strong limitation to any language transfer in the translation task as every encoder/decoder is not only language specific but also task specific.

The goal of multilingual machine translation is indeed to address this limitation by generating models that can transfer language knowledge across tasks. This is achieved by frameworks from the multi-task learning area, such as jointly training two models for several tasks sharing part of the model weights [86, 112]. For example, for generating a model that can translate from both Spanish and Portuguese to English the model would be trained using pairs from both tasks, separate Spanish and Portuguese encoders but a single English decoder. This training strategy enables training the English decoder using data from both tasks (Spanish-English and Portuguese-English), benefiting from a larger aligned corpora and therefore achieving better decoding and translation performance. As described in Section 5.2, different strategies have been proposed in literature for multilingual machine translation. However, to the best of our knowledge, all of them consider the encoders and decoders as a atomic unit that cannot be separated any further, just differentiating from each other by how many full decoders or encoders the model uses for multilingual translation purposes, i.e., one-to-many, many-to-one, or one-to-one [86, 112]. One of the limitations of these models is that they consider all languages to be of same nature, meaning that all languages are combined into a single encoder/decoder without any organization, ignoring the fact that some languages might indeed benefit each other while others would hinder the final performance of the translation task. This has demonstrated to be the case in related areas such as cross-lingual word embedding generation [171]. Instead, HNMT is capable of incorporating further divisions inside the encoder/decoder allowing a more fine grained possibilities for determining which languages should share weights or not as we describe in the following section.

Figure 5.2: Description of the general architecture of HNMT



5.3.3 Hierarchical Multilingual Machine Translation

For HNMT, we define the multilingual machine translation task as a one-to-one task, meaning that HNMT only contains a single encoder and a single decoder. However, each layer $ENC_{L_t}(i)$ or $DEC_{L_t}(i)$ is shared or not across languages depending on their similarity. This enables similar languages to share a larger amount of layers, fostering further language transfer among them, while different language share less layers, avoiding hindering the model's overall performance by forcefully combining language that are too distant. To delineate this inter-connectivity across languages HNMT takes into account the hierarchical nature of languages by taking advantage of a typological language tree. As illustrated in Figure 5.2 each family in the language tree corresponds to a layer of the encoder and the decoder. Each language always has a unique, non-shared-layer, which correspond to the very first layer of the encoder and very last layer in the decoder. This is due to the fact that we consider that each language to be different from any other even if it is to a small degree. This layer enables the model to capture these language-specific characteristics in both the encoder and the decoder. Additionally, HNMT also incorporates one layer that is shared across all languages. This layer is located directly before and after the h vector representation of the sentence, and its purpose is to unify how the model generates this vector regardless of the language used. The remaining intermediate layers are directly determined by the language tree.

For illustration purposes consider the translation task from Spanish to English versus the same task but from Spanish to Finnish. Based on the structure described in Figure 5.2, Spanish to English translating requires the following 8 steps:

$$\begin{gathered} x \xrightarrow{ENC_{es}} h_{enc_1} \xrightarrow{ENC_{it}} h_{enc_2} \xrightarrow{ENC_{in}} h_{enc_3} \xrightarrow{ENC_{wo}} h \\ h \xrightarrow{DEC_{wo}} h_{dec_1} \xrightarrow{DEC_{in}} h_{dec_2} \xrightarrow{DEC_{ge}} h_{dec_3} \xrightarrow{DEC_{en}} y \end{gathered} \tag{5.2}$$

where ENC_l and DEC_l refer to the encoder and decoder layers of language l respectively.

Spanish to Finnish translation, on the other hand, requires the following steps:

$$\begin{aligned} x \xrightarrow{ENC_{es}} h_{enc_1} \xrightarrow{ENC_{it}} h_{enc_2} \xrightarrow{ENC_{in}} h_{enc_3} \xrightarrow{ENC_{wo}} h \\ h \xrightarrow{DEC_{wo}} h_{dec_1} \xrightarrow{DEC_{ur2}} h_{dec_2} \xrightarrow{DEC_{ur1}} h_{dec_3} \xrightarrow{DEC_{fi}} y \end{aligned} \tag{5.3}$$

It is important to note that, as reflected in Figure 5.2, the language tree is not an equally balanced tree, meaning that the number of families from any language to the root node is different. In the example, the number of families from Spanish to the root is two, while the number of families from Finnish to the root is just one. This characteristic of the tree directly conflicts with the requirement of most existing sequence-to-sequence models to contain a same amount of layers in the encoder and the decoder. Additionally, some language might contain more families than layers are used in the model, i.e., if the layer number is chosen to be $N = 3$ the families of English, German, Spanish, and Portuguese would not fit into the model. In order to address both of these concerns, we conduct a two-step preprocessing of the tree. First, we limit the tree to have $N - 2$ layers, pruning any family that does not meet this constraint. Thereafter, we duplicate any leaf node that is not in the layer $N - 2$ of the tree, e.g., in the sample tree in Figure 5.2, the Uralic family is duplicated to adhere to this constraint.

5.3.4 Training HNMT for Sentence Translation

HNMT takes advantage of stochastic gradient descend for learning the weights of its model. Different from a traditional machine translation strategy, HNMT utilizes multiple datasets with different languages for training. Training is conducted in a round-robin fashion with respect to the datasets, i.e., one epoch of each dataset is trained in a sequential manner. Each dataset epoch is divided into batches of sentence pairs β_{batch}, for which the loss function is computed, backtracked and parameters tuned. We use cross entropy as loss function and Adaptive Movement Estimation [142] as optimizer with a learning rate of β_{lr}. The training will continue until no improvement is found on the training set for the average loss across all datasets in the last 10 epochs. In order to avoid overfitting, the model selected for testing is the model that achieves best performance in a separated validation set. Refer to Section 5.4.2 for further details on how we split the datasets and tune hyper-parameters.

5.4 Evaluation Framework

In this section, we describe the evaluation framework used for examining the performance of HNMT and showcasing its advantages with respect to existing baselines.

5.4.1 Data

We use the GlobalVoices parallel corpora [236] for training and evaluation purposes. This dataset is comprised of bilingual corpora for most combinations across 41 languages, totaling 758 different tasks, i.e., pairs of languages for translation. Each task contains a varying amount of parallel sentence pairs that go from less than 10k sentences (in the case of Catalan-English) to more than half a million (in the case of Spanish-English). The strong variation of corpora available for each task mimics a real world scenario where few languages are very rich in resources while many barely have resources associated with them, making this dataset ideal for our experiments.

In order to input words to a neural machine translation model they first need to be converted into a numerical vector representation. For doing so, we take advantage of the cross-lingual word embeddings generated by Madrazo Azpiazu and Pera [171], tailored to low-resource scenarios, a case we consider of specific interest in our experiments.

Finally, we use the language tree described in Lewis and Gary [157] on our experiments. This tree can sometimes be overly detailed, containing too many names describing nearly the same family of languages. For this reason, we prune the original tree to remove family names that can be treated as redundant for translation purposes. For example, having both Central Iberian and Castilian as family for the Spanish language is redundant. For pruning, we define the following criteria: Any family that contains exactly the same amount of languages as its parent is removed.

5.4.2 Validation and Hyper-parameter Tuning

Each of the 758 task specific datasets considered in this study is randomly separated into 3 splits using 70%, 10%, and 20% of the sentence pairs for training, validation, and testing, respectively. The training set used for learning the weights of the model. The validation portion is used for selecting the best model among the ones generated during training. Finally, the testing set is only used for measuring the performance of the final model. Disjoint from these 3 sets, we held-out a development set of 20k Spanish-English sentence pairs. This development set is only used for verifying the correctness of the implementation and tuning hyper-parameters. No sentence pair in this held-out set is ever included in any of the train, validation, or testing sets.

Hyper-parameters where manually selected, meaning that no exhaustive/automatic hyper-parameter tuning strategy was applied. The final hyper-parameters used in the experiments are: $\beta_{batch} = 768$, $\beta_{lr} = 0.01$, $|h| = 512$, $N = 5$, Layer-type=LSTM. It is true that the number of weights selected for our strategy is comparably smaller than what most state-of-the-art strategies currently use[127]. In fact, for HNMT we use $|h| = 512$ and $N = 5$, whereas for current strategies $|h| = 1024$ and $N = 8$ are customary. This was a compromise we had to take in order to balance for the large number of tasks we consider in the study, i.e., 758, compared with the *less than a dozen* tasks most current studies consider [16, 127].

5.4.3 Baselines

To contextualize the performance of HNMT, we compare its performance to that obtained by four baselines: a traditional bilingual baseline (Many-to-many) and three multilingual baselines (One-to-many, Many-to-one, One-to-one).

1. **Many-to-many**. This model resembles the traditional bilingual machine translation strategy where each task has it own encoder a decoders.

2. **One-to-many**. In this multilingual machine translation model one encoder is used regardless of the language and a different decoder for each language.

3. **Many-to-one**. Opposite to the previous model this one uses a single decoder for all he language but multiple encoders, one per language.

4. **One-to-one**. This is a universal machine translation model that utilizes just one encoder and one decoder for all the languages considered.

Models that have a single decoder require some explicit information of the output language in order to enable the model to know the language in which it needs to generate the output sentence. For these models, we prepend the input of the model with a special token representing the language that needs to be generated similar to what is done by Johnson et al. [127]. It is also important to note that unlike aforementioned models HNMT does not require this token to operate, given that the last layer of the decoder is specific to the target language.

5.4.4 Metric

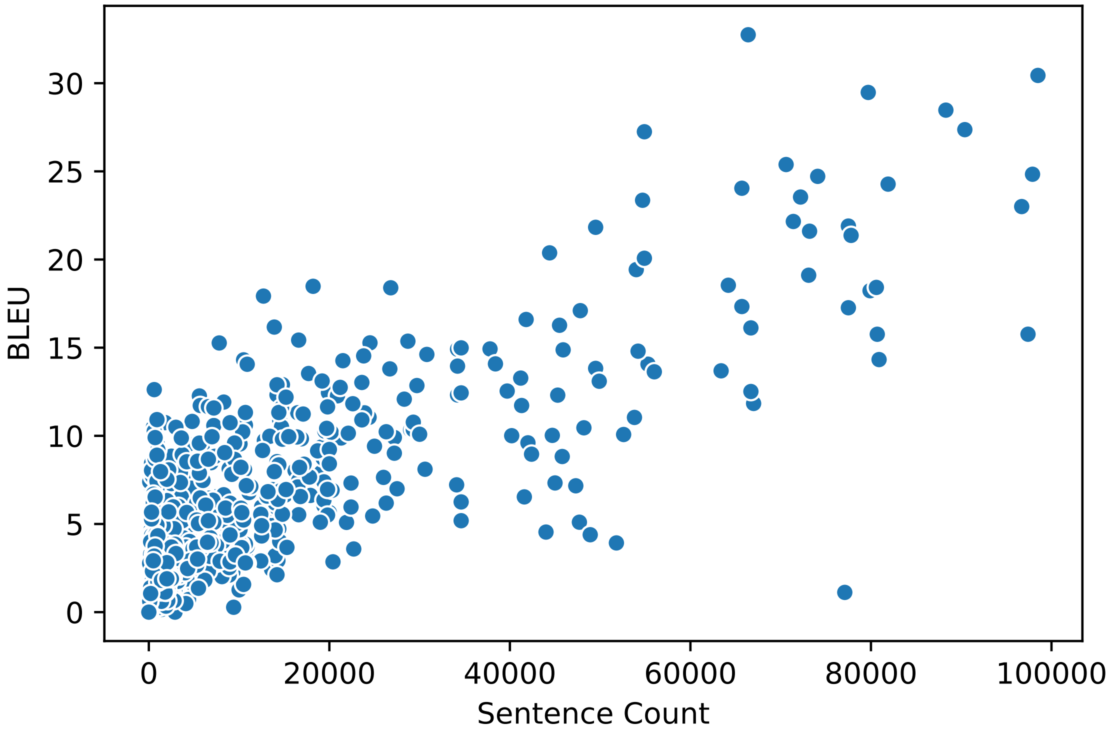
For measuring the performance of each task we take advantage of a traditional metric to the machine translation area: Bilingual Evaluation Under Study (BLEU) [199].

5.5 Results and Discussion

In order to fully analyze the performance of HNMT, it is important to first understand what traditional (bilingual) machine translation strategies can achieve. Therefore, we start by analyzing the performance of the many-to-many model from different perspectives. For doing so, we train and test this model for each of the tasks defined by a language pair in the GlobalVoices dataset. In Figure 5.3, we illustrate the performance of each task organized by the number of sentences available for the task. Emerging from the figure is a pattern that is inherent to the machine translation area: the more sentence pairs available, the better the performance of the model. This leads to an uneven scenario, one where strategies are better performing, i.e., are more effective, for resource-rich languages than for low-resource ones.

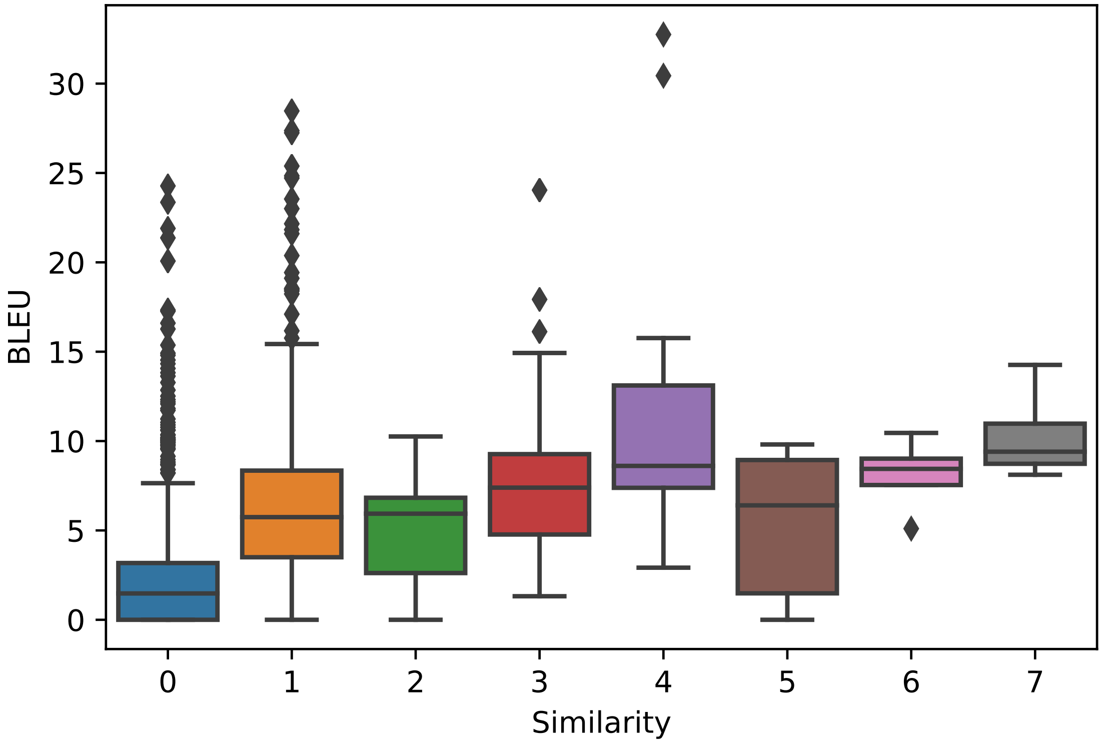
Another issue affecting machine translation related to the direct connection that exists between performance and language similarity. While this is a fact that has been pointed out by several researchers [44, 99], to the best of our knowledge, this has never been thoroughly

Figure 5.3: BLEU score obtained for each of the bilingual tasks using a traditional machine translation baseline (many-to-many), organized by the number of sentence pairs available for the task. Results for tasks with more than 100k sentence are omitted for visualization purposes.



studied. For demonstrating how this dependency behaves in our experiments, we define the similarity between two language as the number of parent family nodes they share. As an example, if we refer back to Figure 5.2, the similarity between English and German is 2 as they both have the Germanic and Indo-European families as parents, while the similarity between Finnish and English is 0 as they do not have any common parent family. We depict in Figure 5.4 the performance of each of the tasks grouped by the similarity between the source and the target languages. From the figure, it is evident that the results follow a

Figure 5.4: BLEU score obtained for each of the bilingual tasks using a traditional machine translation baseline (many-to-many), grouped by the similarity between the source and target language for the task.



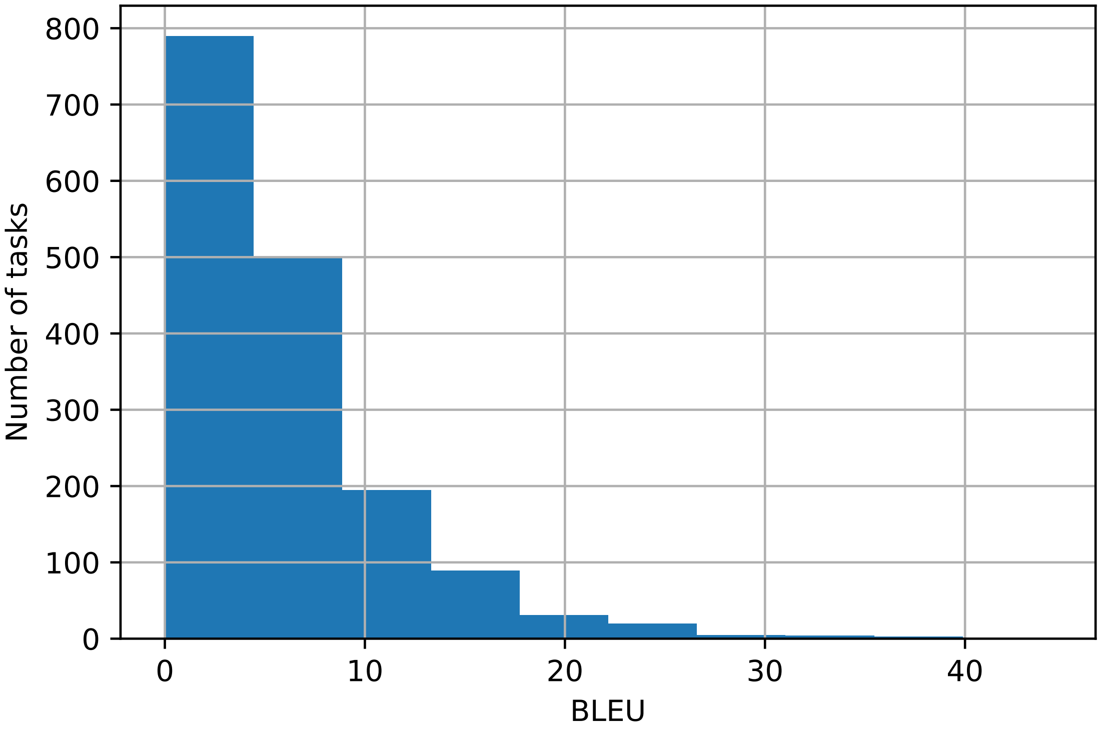
pattern where the more similar any two languages are, the better the quality of machine translation achieved for them.

These two patterns demonstrate (1) a real need in the area of machine translation for designing transfer learning strategy that can improve the performance of machine translation for low-resource languages, and (2) a possibility to achieve valuable language transfer by using a proper organization of the languages, that makes is possible to take advantage of synergies among similar languages. These two patterns, further validate the premises that leaded us to design HNMT.

In Table 5.1 we present the results of 5 different machine translation models for each of the tasks in GlobalVoices dataset, grouped by source language. We include the traditional machine translation model (many-to-many), 3 frameworks for building multilingual machine translation that are representative of the current state-of-the-art, and our HNMT framework. As shown in the table, our proposed strategy yields an average gain of 1.07 BLEU points over the traditional many-to-many strategy. This difference is statistically significant under paired T-test with a confidence interval of $p < 0.05$. Largest improvements are found in languages such as Catalan, Portuguese, Italian, or Spanish, which we find not to be a coincidence but the result of HNMT correctly integrating languages that share similarities. The lowest improvement is obtained for Oriya, an Indo-Aryan language that shares little similarity with respect to any of the remaining languages in the dataset. Among multilingual machine translation models (one-to-one, one-to-many, many-to-one), only the one-to-many model achieves an improvement over the traditional bilingual baseline. This is not a surprising result as, even if multilingual machine translation models have shown to improve over traditional machine translation with specific language combinations, they are known to under-perform when simultaneously dealing with either too many or too different languages [127].

Table 5.1 captures average BLEU scores over all tasks that use the specified language as source. While average allow us to assess and compare performance across frameworks, it does not shine a light on translation pairs that greatly deviate from the average. To showcase the varied degrees of BLEU scores obtained by HNMT for each of the 758 translation tasks, we included an histogram in Figure 5.5. It can be appreciated in the figure that for most of the translation tasks performance ranges between 0 and 10 BLEU points. However, there are some cases for which BLUE is as high as 30-40. Not surprisingly, these cases align with popular, resource-rich languages like Spanish-English and Portuguese-English. At the opposite extreme, we see BLEU as low as 0.27. Once again, this is anticipated, as these low scores are the result of translation to/from low-resource (and often less recognized) languages, like Catalan to Oriya.

Figure 5.5: Distribution of BLEU scores yielded by HNMT for individual translation tasks.



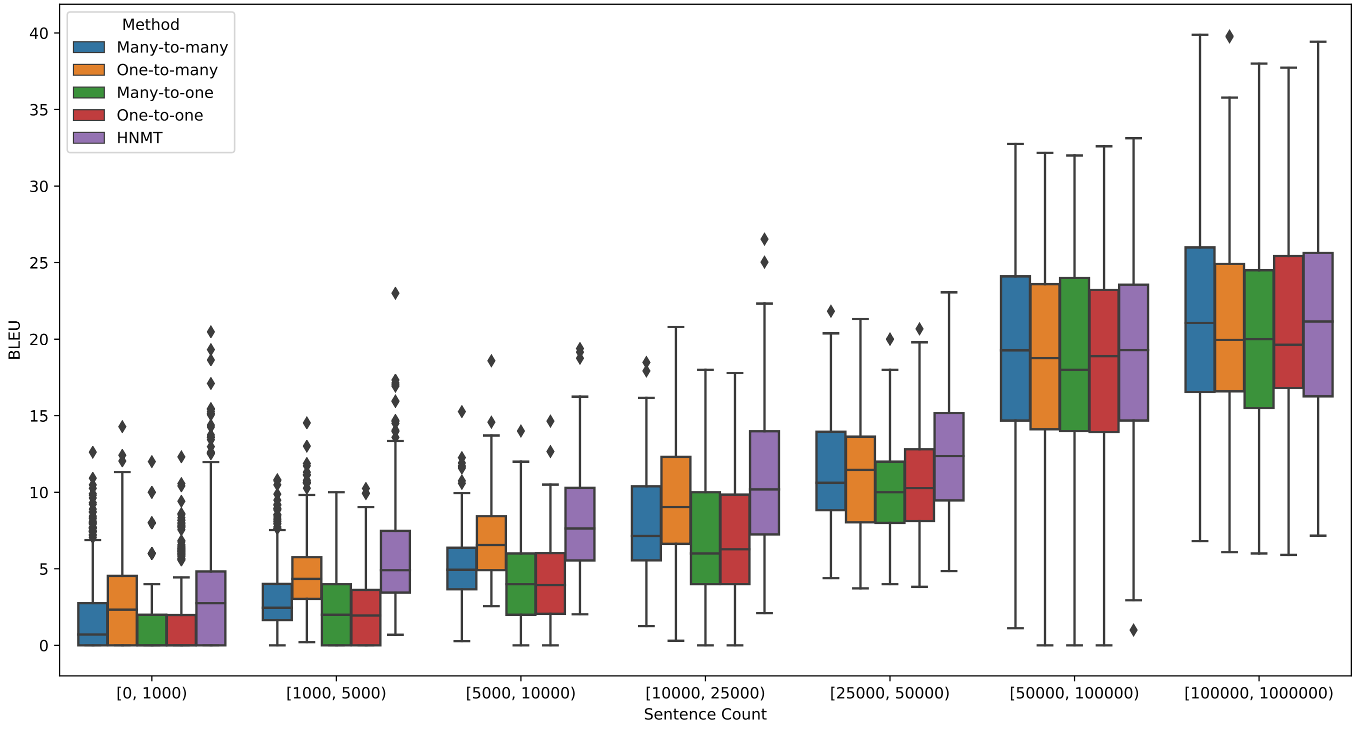
In order to gather further insights on the translation capabilities of HNMT and to better visualize in which cases does HNMT achieve performance improvements with respect to other baselines, we conduct further analysis using language similarity and corpora availability lenses.

We first explore model performance when corpora of different sizes is used for training purposes. To do so, we grouped each of the language pair tasks into seven different groups based on the amount of parallel sentence available. As depicted in Figure 5.6, corpora availability is a determinant factor for translation. The pattern we devised in Figure 5.3 is once again visible in Figure 5.6, the more sentences available for a task the better is its performance. However, differences with respect to the baseline are what make a model stand out in this case. Excluding the cases with high amount of corpora, where improvement is hardly possible from a language transfer perspective, we see that HNMT is the model that achieves the most improvement with respect to the bilingual baseline, followed by the one-to-many model. This behavior denotes that HNMT is indeed capable of improving the performance of machine translation in cases where resources are not abundant.

Table 5.1: BLEU scores obtained by each of the baselines and the proposed model for each language considered in the study. Improvement denotes the difference with respect to the bilingual machine translation baseline (Many-to-many). Note that values shown for each language correspond to the average BLEU obtained using the language as source with respect to all other languages used as target. Language names are described using ISO-639 notation.

	Many-to-many	One-to-many	Many-to-one	One-to-one	HNMT	Improvement
am	1.51	1.65	1.32	1.15	**2.02**	0.51
ar	6.08	6.43	5.41	4.90	**7.15**	1.06
aym	4.08	4.28	3.43	3.04	**5.07**	0.98
bg	4.22	4.38	3.55	3.23	**5.59**	1.36
bn	8.92	9.11	7.85	7.53	**10.28**	1.36
ca	5.62	5.96	4.73	4.43	**7.33**	1.71
cs	4.76	4.90	3.99	3.89	**6.04**	1.28
da	4.49	4.74	3.79	3.64	**5.57**	1.08
de	7.57	7.76	6.08	6.20	**8.98**	1.42
el	7.54	7.78	6.43	6.06	**8.78**	1.25
en	11.26	11.43	9.92	9.73	**12.51**	1.25
eo	2.33	2.66	1.75	1.75	**3.12**	0.78
es	11.61	11.82	10.45	10.22	**13.35**	1.74
fa	3.74	3.99	3.14	2.85	**4.64**	0.90
fil	2.60	2.91	2.09	1.97	**3.31**	0.71
fr	8.72	8.90	7.70	7.29	**10.27**	1.55
he	1.21	1.36	0.92	0.95	**1.65**	0.43
hi	1.71	1.75	1.44	1.30	**2.32**	0.61
hu	4.50	4.79	3.61	3.40	**5.46**	0.97
id	4.05	4.21	3.04	3.02	**4.87**	0.82
it	7.84	8.04	6.73	6.47	**9.35**	1.51
jp	6.53	6.65	5.48	5.12	**7.47**	0.95
km	0.81	0.96	0.64	0.57	**1.22**	0.41
ko	3.67	4.05	2.99	2.79	**4.58**	0.90
mg	8.67	8.90	7.64	7.24	**9.55**	0.88
mk	6.14	6.37	5.45	4.87	**7.67**	1.52
my	1.51	1.74	1.31	1.09	**2.15**	0.64
nl	6.21	6.42	5.28	5.05	**7.57**	1.35
or	0.40	0.43	0.29	0.29	**0.48**	0.09
pl	6.57	6.77	5.39	5.28	**7.86**	1.29
pt	6.95	7.18	5.86	5.46	**8.71**	1.77
ro	3.33	3.53	2.78	2.58	**4.33**	1.00
ru	8.16	8.34	7.00	6.81	**9.61**	1.45
sq	4.03	4.31	3.54	3.25	**5.26**	1.22
sr	5.39	5.62	4.38	4.22	**6.63**	1.24
sv	4.88	5.06	3.93	3.75	**6.07**	1.19
sw	5.22	5.47	4.43	3.98	**6.17**	0.95
tr	3.54	3.68	2.99	2.66	**4.29**	0.74
ur	3.61	3.76	2.98	2.77	**4.48**	0.87
zhs	6.22	6.43	5.10	4.90	**7.19**	0.97
zht	6.41	6.58	5.38	5.20	**7.39**	0.98
Average	5.19	5.39	4.40	4.17	**6.25**	1.07

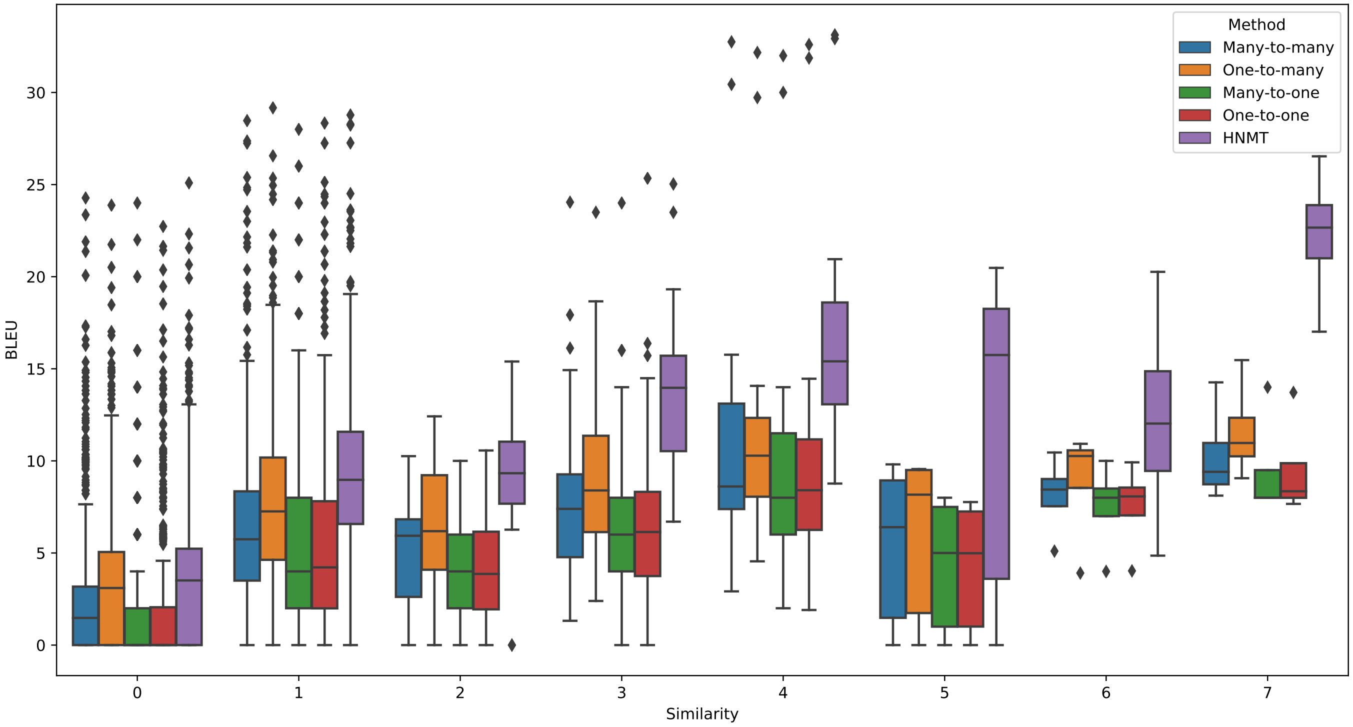
Figure 5.6: BLEU score obtained for each of the bilingual translations tasks using several translation models, grouped by the number of sentence pairs available for the task. Results for tasks with more than 100k sentences are omitted to ease visualization.



We are also interested in exploring the effect language similarity has on translation, which is why we examine model performance for languages pairs with different degrees of similarity between them. Results from this experiment are summarized in Figure 5.7.

In general, we observe similar patterns to the ones we previously described: BLEU scores computed for machine translation task are higher for languages that are similar. This pattern occurs regardless of the language, however, it is considerably more pronounced in the case of HNMT, leading to a higher improvement with respect to the bilingual baseline the more similar the languages are. We also notice from Figure 5.7 that none of the other multilingual machine translation models takes advantage of this behavior, maintaining a similar difference with respect to the baseline regardless of the degree of similarity between the languages in pairs considered from analysis. These results serve as indication that the hierarchical organization used in HNMT is indeed useful for explicitly taking advantage of similarities across languages, validating our premises for the design of HNMT.

Figure 5.7: BLEU score obtained for the models considered in our analysis, grouped by the similarity between the source and target language in the task.



5.6 Conclusion and Future Work

In this chapter, we presented HNMT, an hierarchical framework for machine translation that can be applied to any multilingual neural machine translation strategy, for achieving a higher degree of transfer learning across languages. We conducted several experiments using 758 language pairs including languages with varied resource availability and similarity. Our empirical analysis reveals that highest improvements take place when the languages are typologically related and aligned corpora is not abundant, achieving an improvement of about 5 BLEU points in specific cases. These results validate our premise that machine translation for low-resource languages can be enhanced by means of language transfer if an appropriate organization of languages is used, such as the one we utilize as part of HNMT. As a natural part of its encoding-decoding process for translation, HNMT generates a language-agnostic vector representation of sentences. While we did not evaluated the quality of this by-product of our work, given that it was out of scope, exploratory examinations lead us to believe that these language-agnostic representations could be leveraged for supporting a multilingual applications in related text processing areas.

We are aware of some limitations of this work. First, even if the strategy is shown to improve low-resource scenarios where the source and target language are typologically related, this effect is not as prominent when the languages are different from each other. Consequently, the applicability of HNMT for isolated languages such as Basque is limited. Second, given the high amount of tasks and languages considered, the size of the machine translation models we used for experimentation is small compared to current state-of-the-art systems. For example, we set $|h| = 512$ and $N = 5$, when current strategies use $|h| = 1024$ and $N = 8$. In the future, we plan on leveraging other types of signals, such as the use of sub-word embeddings, for enabling further language transfer. Additionally, we will extend our empirical analysis to explore the performance effect of using larger and more varied machine translation models, such as Convolutional Neural Networks or Transformers.

CHAPTER 6

AN ANALYSIS OF TRANSFER LEARNING METHODS FOR MULTILINGUAL READABILITY ASSESSMENT

Abstract

Recent advances in readability assessment have lead to the introduction of multilingual strategies that can predict the reading-level of a text regardless of its language. These strategies, however, tend to be limited to just operating in different languages rather than taking any explicit advantage of the multilingual corpora they utilize. In this manuscript, we discuss the results of the in-depth empirical analysis we conducted to assess the language transfer capabilities of four different strategies for readability assessment with increasing multilingual power. Results showcase that transfer learning is a valid option for improving the performance of readability assessment, particularly in the case of typologically similar languages and when training corpora availability is limited.

6.1 Introduction

Readability assessment has historically been used by different stakeholders —from educators to public institutions— for measuring the complexity of texts [26]. Traditionally, readability assessment relied on formulas based on a linear combination of superficial text features, such as the amount of words per sentence or the average number of letters in a word [88, 51]. In time, the area evolved to incorporate features that describe the text from more diverse linguistic perspectives, e.g., syntax, morphology, pragmatics, and more sophisticated supervised machine learning strategies [93, 101, 135]. The increasing amount of hand-engineered features made state-of-the-art readability assessment techniques strongly dependant on the language they where designed for, making adaptation for low resource languages difficult.

Recent advances in the area have focused on addressing the language-dependency problem by either: (1) finding a combination of features that can work for a set of multiple languages [54, 172], or (2) building strategies that do not depend of hand-engineered features [170]. The former option still has the adaptability limitation of single language strategies, as they are tuned to work in a fixed set of languages, and new features need to be designed in order to incorporate extra languages. The latter avoids using features, taking advantage of deep-learning techniques that solely rely on words as input, meaning that they can be utilized in any language without specific tuning. However, given its dependency on deep-learning techniques that require larger amounts of data than feature-based counterparts, it has been shown to underperform in low-resource scenarios [170].

Regardless of the alternative considered, the amount of leveled corpora available plays a strong role in the quality of the predictions readability assessment strategies can provide.

This creates a scenario where the quality of readability assessment for popular languages, such as English, is drastically higher than for other low-resource languages, such as Catalan [172]. Additionally, despite the fact that recent strategies can work in different languages, they do not enable any cross-lingual transfer that could increase the quality for low-resource language readability assessment.

In this manuscript, we conduct an empirical study in order to analyze the extent to which existing transfer learning techniques can be used as part of a readability assessment in order to improve the performance for low-resource languages. For doing so, we compare four multilingual readability assessment strategies —three of which are created specifically for our experiments— in terms of their language transfer capabilities. The study is driven by two main research questions: (1) Can existing transfer learning techniques be applied to readability assessment? and (2) In which circumstances (languages or models) is transfer learning maximized?. To answer these questions, we conduct transfer learning experiments over six different languages; we combine different languages in a pairwise fashion and we also explore the effects of using varying amounts of training documents. Results indicate that transference across languages is indeed a viable option for improving the performance of low-resource languages that are typologically[1] related to other languages for which more abundant resources exist. Typologically isolated languages, such as Basque, also obtain improvements in certain cases. Unfortunately, this isolation is what causes them to act in detriment of the effectiveness in readability prediction of other languages that used as supporting language.

[1] Typology is the area of linguistic that categorizes languages given their lexical, grammatical, or other linguistic patterns. The degree of similarity between any two languages is defined by the linguistic patterns they share.

6.2 Related Work

State-of-the-art in readability assessment has mostly been focused on the design of features for different languages with varied linguistic perspectives: Aluisio et al. [7] propose a strategy for predicting whether English texts are simple or complex using elaborated semantic features, such as term ambiguity. Anula [12] propose using features focused on the frequency of words for Spanish. Gonzalez-Dios et al. [101] put more emphasis on morphological features, known to be important in morphologically rich languages like Basque or French [93]. Forsyth [91] analyze the importance of using discourse features for Arabic readability assessment. Syntactic features have demonstrated to be useful in multiple languages, such as Italian [56] and Russian Karpov et al. [135], but they are not considered as important for languages like Chinese, in which case lexical features are more prevalent [39].

Even if literature regarding single language readability assessment is extensive, little has been done regarding multilingual readability assessment. De Clercq and Hoste [54] analyzed the importance of several existing features for Dutch and English readability assessment, determining that a single set of features could be used for both languages at the same time. Madrazo Azpiazu and Pera [172] analyzed the possibility of building a single readability assessment tool for six different languages at the same time. While effective, the tool is limited to the use of hand-engineered features in multilingual readability assessment given their strong language dependency. This is the reason why the authors dedicated research efforts to the design of the first feature-less strategy Madrazo Azpiazu and Pera [170]. The proposed strategy solely relies on words, thus enabling further multilingual adaptability, which previous strategies lacked.

Even if there exist readability assessment strategies that can work regardless of the language, none of them actually benefits from the multilingual nature of corpora they are using. This prompted our examination of diverse strategies that incrementally enable language transfer in multilingual readability assessment strategies.

6.3 Evaluation Framework

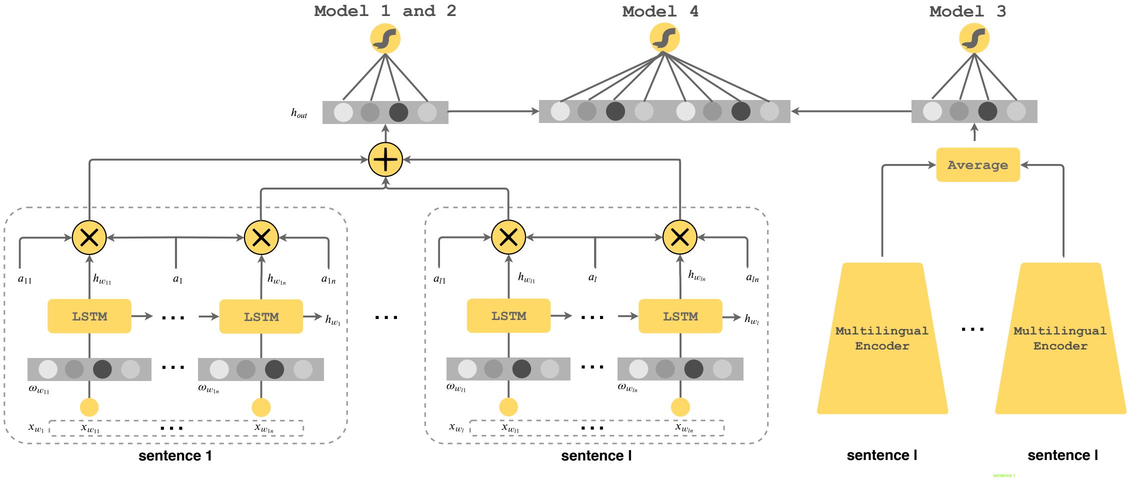
In this section, we describe the set up, models, languages, datasets, and metrics considered in our empirical study.

6.3.1 Experiment Set Up

Let experiment E be expressed by the tuple $<M, D, L_{train}, N, L_{test}>$, where M is a readability prediction model (i.e., any of the models described Section 6.3.2), L_{train} is an ordered pair of languages used for training, D is the dataset considered in the experiment, N is an ordered pair denoting the number of documents used for each language in L_{train}, and L_{train} is the language the model is being tested on for readability assessment. As an example, $<M1, Newsela, (en, es), (50, 1000), (es)>$ represents an experiment where M1 is trained in the Newsela dataset, for English and Spanish with 50 and 100 documents respectively, and tested only in Spanish.

We use 10-cross-fold for validating our results, where the counts of N are sampled from the respective 9-fold training set in each iteration, using uniform random sampling with duplicates. Results are averaged across all iterations unless otherwise indicated.

Figure 6.1: Description of the general architecture of the four considered strategies.



6.3.2 Compared Strategies

We describe below the four models used for transfer learning analysis (illustrated in Figure 6.1).

M1: Baseline Model

In order to compare strategies that can be leveraged for achieving language transfer learning, we first need to analyze the degree to which existing models can transfer knowledge across languages. For doing so, we use an existing state-of-the-art strategy as a starting point into our experiments.

As we described in Section 6.2, there are two common alternatives for designing strategies for multilingual readability assessment. On the one hand, depend on a single set of features that works in all the languages considered. This mean that even if a strategy can be designed to work in n languages, adding an additional one will always will always require building new language-specific features. On the other hand, adopt a feature-less approach that can work on any language. Given that the latter alternative does not rely on any hand-engineered features, it works independently of the languages considered. Therefore, we find this latter strategy to be the most promising to be used as a baseline. Specifically, we choose the strategy proposed by Madrazo Azpiazu and Pera [170] to serve as a baseline model M1. Given that models M2, M3, and M4 are incrementally built based on this model, we include a brief overview of M1 below.

M1 is based on a architecture that is both recurrent and hierarchical Madrazo Azpiazu and Pera [170]. The input to the model is a text that is split into sentences and converted into word embeddings. We follow the authors' choice of using the FastText word embeddings [107]. These word embeddings are combined using a LSTM recurrent neural network per sentence. Finally, the hidden states generated by the recurrent layer are combined into a text representation using word and sentence level attention mechanisms.

For experimentation, we distinguish between two variants of this model: M1-Mono, which refers to a monolingual variant which only takes input data in one language, and M1 which takes input texts in various languages for training.

M2: Cross-lingual Word Embeddings

One of the limitations of M1, in terms of transfer learning across languages, is that the input representations, i.e. the word embeddings, are different for each language. For example, the word *house* in English and its counterpart in Spanish *casa*, will be represented by different word embeddings. M2 addresses this issue by taking advantage of cross-lingual word embeddings. This allows the model to learn about simple or complex words in one language and directly transfer this knowledge to other languages given that same words will always have same embedding regardless of the language. For building M2, we use the cross-lingual word embeddings proposed by Madrazo Azpiazu and Pera [171].

M3: Cross-lingual Sentence Embeddings

Readability assessment literature has strongly highlighted the importance of syntax when determining the complexity of a text [135, 93]. Even if M2 can learn about words being simple or complex regardless of the language, the order in which these words occur in the text can drastically change the meaning of the sentences in the text and, as a result, the overall complexity of the text. In order to enable syntax-level language transfer, we design M3 to take advantage of cross-lingual sentence embeddings. These sentence embeddings are generated following the procedure proposed by Madrazo Azpiazu and Pera [171], based on an enconder-decoder architecture. Similar to M2, the input text is converted into cross-lingual word embbedings which are fed to a cross-lingual sentence enconder. The generated sentence encodings are averaged in order to generate a text representation. Finally, a fully-connected layer is used to generate a prediction.

M4: Hybrid Model

The last model considered in our study is M4, which is the result of the combination of M2 and M3. In this case, M4 is able to simultaneously take advantage of both the word-level and sentence-level transfer capabilities of M2 and M3. For doing so, M4 combines the text level encodings generated by both M2 and M3 via concatenation in order to generate a new enconding that combines both word and sentence level information. Similar to the other models, a final fully-connected layer is the one responsible of generating a prediction.

6.3.3 Languages

In this study, we consider six different languages which we name following their ISO 6391 codes: English=en, Spanish=es, Italian=it, Catalan=ca, Basque=eu, and French=fr.

6.3.4 Datasets

For evaluation purposes, we use the VikiWiki dataset. This publicly available dataset[2] was first introduced in [170] and is comprised of simple/complex documents extracted from Vikidia(.org) and Wikipedia(.org). The dataset consists of 70,514 documents in 6 different languages: 23,648 in French, 9,470 in Italian, 8,390 in Spanish, 3,534 in English, 924 in Catalan, and 898 in Basque, which are evenly distributed among both reading-levels. To control size of text samples available per language, we reduce the amount of documents in each language to the language with least documents. As a result, each language contains a total of 898 documents in our experiments, yielding a training set of 808 documents and a testing set of 90 in each iteration of the 10-cross-fold validation.

6.3.5 Hyper-parameter Tuning

We use a disjoint dataset sampled from VikiWiki for Spanish and English for developmental purposes, composed of 100 documents for each of the two languages. Documents in this development dataset were solely used for code testing purposes, as well as hyper-parameter

[2] https://github.com/ionmadrazo/VikiWiki

tuning. No document in the development set is part of any of the training or testing datasets used for evaluation. Note that we did not conduct any experiment specific hyper-parameter tuning; instead, all the hyper-parameters where manually tuned using the developmental set.

Hyper-parameters used in the experiments discussed in Section 6.4 are described as follows: $d_{word} = 300, d_{sent} = 512$; for M1 and the sentence encoder in M2 and M3 we use the hyper-parameters reported by the authors, see [170, 171].

6.3.6 Performance Metric

In order to asses the language transfer capabilities of each model we use ACC@K (Accuracy at K). In this case, accuracy is defined as the ratio of texts for which their readability levels were correctly predicted over the size of the text collection used for testing, when K documents in a supporting language were used during the training process. For illustration purposes, consider the following sample experiment E=<$M, D, (l_1, l_2), (n_1, k), (l_1)$>, where model M is trained using n_1 resources in language l_1 and k resources in language l_2 in order to predict the readability of resources in l_1. ACC@k for this experiment would reflect the ratio of documents in the corresponding testing set in l_1 for which M is able to estimate their corresponding reading level when trained using k documents from D in supporting language l_2.

6.4 Results and Analysis

In order to evaluate the capabilities of the aforementioned multilingual readability-assessment models for language transfer, we conduct three different experiments, which we discuss in the rest of this section.

6.4.1 Overall Results

For comparing the performance of the four models introduced in Section 6.3.2, we measure the accuracy they obtain for each of the languages in VikiWiki. More specifically, we measure the accuracy gain obtained for each *prediction* language, when trained alongside any of the other *supporting* languages. For example, for computing the accuracy of English, we run 5 different experiments where we train the model using all documents (reduced to 808 as described in Section 6.3.4) in English and each of the other supporting languages, and then average the prediction results as the overall accuracy score for English. This measure provides an estimate of the extent to which the prediction for each language can be improved, as well as a means of comparison across the 4 models.

In order to contextualize the gain obtained as a result of transfer learning by each of the models, we compare the accuracy of each model with respect to that obtained by M1-Mono, the multilingual strategy presented by Madrazo Azpiazu and Pera [170]. Recall that M1-Mono does not incorporate any language transfer functionality. As shown in Table 6.1, simply training an existing model with documents in more than one language, such as M1, does not lead to improvements in terms of the accuracy of the model. Using cross-lingual word embeddings (i.e., M2) has a positive effect regarding language transfer, achieving an average improvement of 0.03 in accuracy. From the results in Table 6.1, it also becomes apparent that, by themselves, cross-lingual sentence embeddings are not enough for providing accurate predictions (i.e., M3). Instead, combining embeddings at word and sentence levels (i.e., M4) yields a more positive outcome, with an average gain of 0.04 .

Overall, M4 is the model that achieves the best performance over its counterparts, obtaining a statistically significant difference in 5 out of 6 cases. Exploring performance from a language perspective, we see that Basque and Italian are the languages that obtain the most benefit from language transfer, as evidenced by their 0.05 gain over M1-Mono when using M4.

Table 6.1: Improvement obtained for each language for ACC@808 averaged for all other languages in VikiWiki. For example the value for M4 and English, is obtained by averaging the improvement over M1-Mono of $<M4, VikiWiki, (en, l), (808, 808), (en)$ for each $l \in (es, fr, it, ca, eu)>$. (*) denotes statistically significant difference under a paired T-test versus other counterparts ($p < 0.05$).

Prediction Lang.	Model				
	M1-Mono	M1	M2	M3	M4
English	.879	-0.05	+0.02	-0.22	+0.03*
French	.884	-0.06	+0.02	-0.27	+0.02
Spanish	.847	-0.04	+0.03	-0.19	+0.04*
Italian	.814	-0.03	+0.03	-0.23	+0.05*
Catalan	.742	-0.08	+0.02	-0.21	+0.03*
Basque	.687	-0.04	+0.04	-0.15	+0.05*
Average	.809	-0.05	+0.03	-0.21	+0.04*

6.4.2 Synergies across Languages

Upon deeper analysis on the results summarized in Table 6.1, we noticed considerable differences in performance improvement depending on which language is used as a supporting language. Given that patterns we found are consistent across the four models, we focus our discussion on the best performing model.

In Table 6.2, we summarize the accuracy obtained by M4 for all language pairs considered in the study. Similar to the previous experiment, our conclusions are based on gain values with respect to M1-Mono, for which baseline values are shown in the diagonal. Based on the reported accuracy scores, Italian is the language that obtains most benefit of of language transfer, specifically when being supported by Spanish or Catalan. Regarding supporting capabilities, Spanish is the language that best supports others followed by Catalan. We attribute this to the typological similarity across some of the languages considered in our analysis, where Catalan and Spanish are probably the two most similar ones. In contrast, Basque is the worst supporting language, among the ones considered. This is anticipated, as this outcome further aligns with our typological similarity hypothesis. It is important to note that while Basque does not lead to improvement when serving as a supporting language to its counterparts, it is one of the languages that most benefit obtains from language transfer. This denotes a non-transitive relationship among languages when it comes to support for readability prediction. We attribute this to the overall complexity of the language, in terms of readability assessment (see M1-Mono results). Any additional document incorporated during the the training of a model for Basque readability assessment has a positive effect on the overall accuracy of the model, i.e., incorporating additional documents translates to overall improvement in readability assessment.

To further showcase how language transfer takes place for each of the language pairs we considered in our experiments, we also discuss the accuracy obtained by M4 when trained using different amounts of supporting documents per language. More specifically, we illustrate in Figure 6.2 the accuracy obtained for Catalan readability assessment when trained with K documents of each of the supporting languages, where K is a variable

Table 6.2: ACC@808 for pairwise combinations of all VikiWiki languages using M4. For each cell the accuracy of the following experiment is computed: $<M4, VikiWiki, (l_1, l_2), (808, 808), (l_1)>$ where l_1 is the predicted language (row) and l_2 is the supporting language (column). Improvements are calculated over the M1-Mono baseline show in the diagonal.

Predicted lang.	Supporting lang.						Average
	English	French	Spanish	Italian	Catalan	Basque	
English	**.879**	0.04	0.05	0.03	0.02	-0.01	0.03
French	0.03	**.884**	0.02	0.03	0.04	-0.02	0.02
Spanish	0.04	0.04	**.847**	0.06	0.06	-0.02	0.04
Italian	0.04	0.05	0.09	**.814**	0.08	-0.03	0.05
Catalan	0.03	0.03	0.07	0.04	**0.742**	-0.04	0.03
Basque	0.04	0.04	0.05	0.05	0.05	**.687**	0.05
Average	0.04	0.04	0.06	0.04	0.05	-0.02	

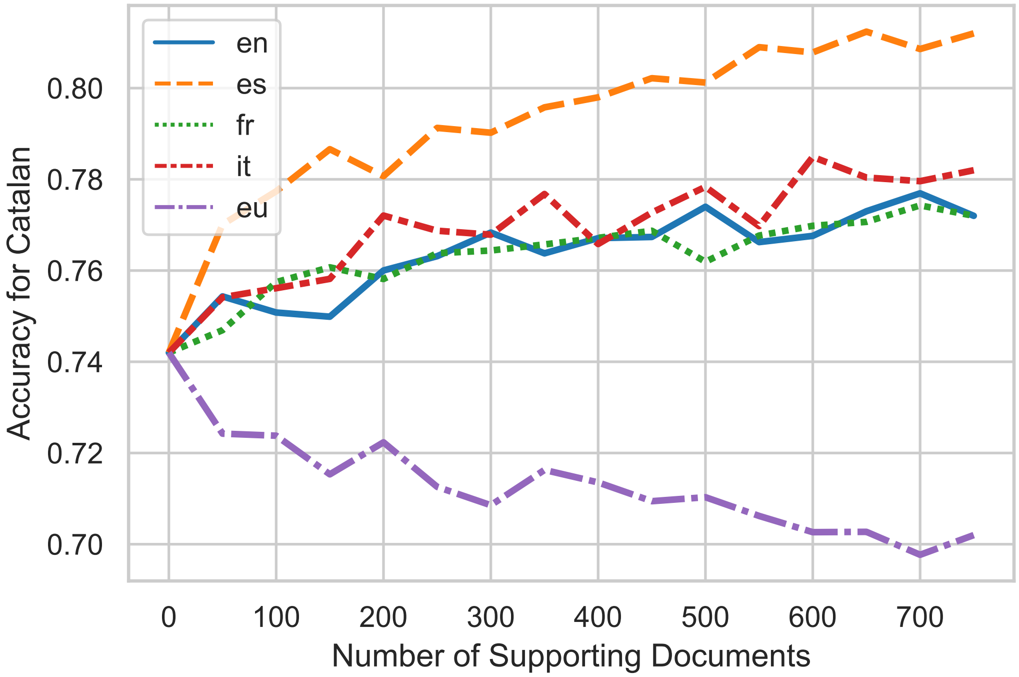
number noted in the x-axis. In addition, we illustrated the results obtained for the same experiment but using Basque for readability assessment in Figure 6.3.

In the case of Catalan, the difference among languages is considerable, specially when being supported by Spanish and Basque. Adding any amount of Spanish documents to the training has a positive effect, the opposite is true when aggregating supporting documents written in Basque during training. In the Basque prediction experiment, illustrated in Figure 6.3, we notice a steady increase in accuracy improvement by adding more supporting documents into the training regardless of the language in which the supporting documents are written.

6.4.3 Low-resource Scenario

An important aspect language transfer is that it can benefit minority languages, as resources for these languages tend to be scarce. In this experiment, we analyze the extent to which the

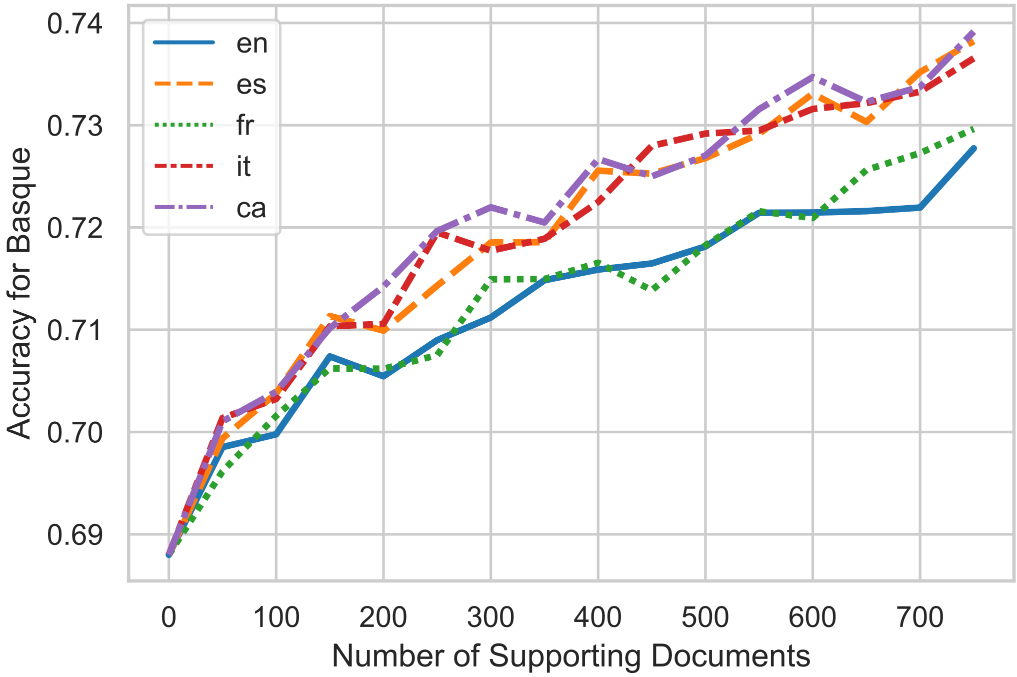
Figure 6.2: Accuracy achieved by the Catalan model using different amounts of documents (x-axis) from other languages, i.e., accuracy that results from experiment $<M4, VikiWiki, (ca, legend), (808, x - axis), (ca)>$.



models can benefit languages in a low-resource scenario. Similar to previous experiments, we only discuss results for M4.

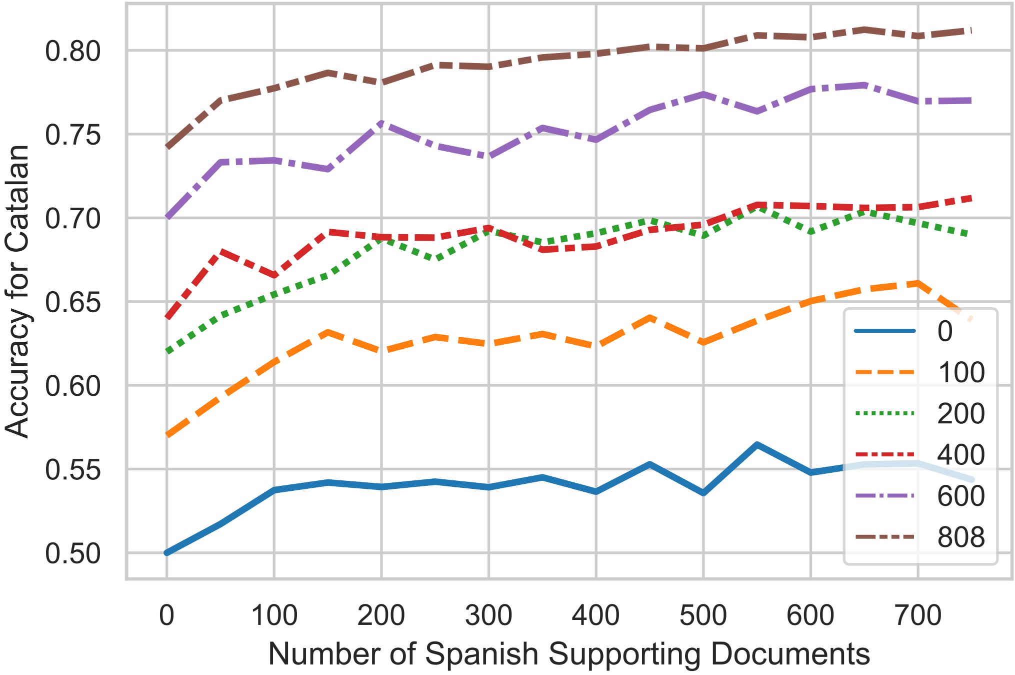
Consider Catalan, a minority language for which labeled resources tend to be scarce. We depict in Figure 6.4 the accuracy obtained for Catalan readability assessment prediction when supported by Spanish. Accuracy ratios are based on the amount of documents written in Catalan and Spanish used for training M4 for Catalan readability assessment. Results indicate that the prediction for Catalan improves regardless of the resource availability for this language by using documents written in Spanish. Even in the zero-shot scenario

Figure 6.3: Accuracy achieved by the Basque model using different amounts of documents (x-axis) from other languages, i.e., accuracy that results from experiment, i.e., $<M4, VikiWiki, (eu, legend), (808, x - axis), (eu)>$.



where we use no Catalan resources at all (with 0.5 accuracy as we are using a balanced dataset), readability prediction is improved by applying M4 trained with Spanish documents. This yields a similar performance to the one obtained when using 100 documents in Catalan. Upon further analysis, we noticed that this behaviour also occurs for other typologically similar languages, e.g., Italian or Spanish, but unfortunately, it is more limited for typologically isolated languages such as Basque.

Figure 6.4: Accuracy achieved by the Catalan model when using different amounts of Spanish supporting documents. Number of documents in Spanish shown in the x-axis and number of document in Catalan denoted by the legend. In other words: $<M4, VikiWiki, (ca, es), (legend, x\text{-}axis), (ca)>$.



6.5 Conclusion

In this chapter, we explored the performance of different strategies that can be applied to achieve cross-lingual language transfer for readability assessment. Best transference characteristics are achieved by a hybrid model (M4) that combines both word and sentence level cross-lingual embeddings. The highest language transfer is observed among languages that are of similar nature, such as in the case of languages of the Romance family. It is important to note that results of the experiments reveal that even in the case of typologically isolated languages, transfer benefits occur, in terms of overall accuracy improvement. In the future, we are interested in exploring the use of sub-word level embeddings, as these could inform language transfer at morphological level and as a result provide additional improvements for morphologically-rich languages, such as Basque or French. Additionally, we plan on replicating the experiments discussed in this manuscript using datasets that include resources written in more languages and labeled with different readability levels, i.e., go beyond binary resource labeling. This will allow us to validate our conclusions when the readability assessment prediction task presents an added layer of complexity.

CHAPTER 7

CONCLUSION

In this dissertation, we explored strategies that respond to the need to adapt Information Retrieval systems (IR) so that they can better serve non-traditional users. We focused on two particular characteristics that make a user deviate from the norm: language and reading abilities.

To address existing knowledge gaps regarding multilingualism for readability assessment, we introduced the first featureless readability assessment strategy that can work regardless of the language (**Chapter 3**). Results from in-depth empirical evaluations with diverse datasets (which we shared with the research community) allowed us to demonstrate that the feature dependency of existing strategies can be avoided by building models that directly rely on the core components of texts, i.e., words and sentences.

In our pursuit of enhancing the transfer learning capabilities of exiting IR strategies, we dedicated research efforts to the area of cross-lingual word embeddings (**Chapter 4**). We found that existing generation techniques are strongly biased towards popular languages, such as English, under-performing on less popular ones, such as Basque. To respond to this limitation, we proposed an hierarchical cross-lingual embedding generation strategy that removes the need to have a single pivot language (English in most cases). This strategy generates better democratized embedding spaces, taking advantage of the synergy among similar languages and avoiding the noise introduced by trying to put distant language together.

Ascending from word to sentence level cross lingual representations (**Chapter 5**), we focused on the area of machine translation—an area where performance is strongly determined by corpora availability and differences between source and target languages. To amend the dependency of existing machine translation strategies for large corpora, we proposed a multilingual hierarchical machine translation strategy that can transfer knowledge across languages for improving the performance of machine translation for languages with low-resources. Similar to our work in the area of cross-lingual word embedding, this strategy takes advantage of similarities across languages and avoids any noise introduced by simultaneously incorporating different languages into the same machine translation strategy.

In order to demonstrate how the three proposed strategies can work in-tandem for addressing the needs of non-traditional users, we conducted a study that enable assessment and comparison of four multilingual readability assessment strategies that are built upon our aforementioned models (**Chapter 6**). In our exploratory analysis, we measured the extent to which each model can achieve language transfer, finding that an hybrid strategy that combines all the models–presented in Chapters 3, 4, and 5–is the one that achieves the best performance, given that it can leverage both word and sentence level information for predicting the reading-level of a text.

The foundations of the work presented in this dissertation lay on how to best represent textual resources in a way that they can be leveraged in the adaptation of existing IR strategies. As a result of our research work, we have designed and deployed strategies for generating language-agnostic word and sentence representations. Furthermore, we have built architectures that support aggregation of words and sentence representations with semantic, syntactic, and morphological representations of text; it is this hybridization ability what makes it possible to perform multilingual readability assessment and machine translation, as we demonstrated throughout the main chapters of this work and the appendixes.

Despite knowledge and lessons learned that emerged from the work conducted so far, we see that still many problems remain open for the research community (even beyond IR) to tackle if the design of IR systems that respond to the needs and expectations of non-traditional users is to be achieved. Below we mention only some of the few questions that have emerged from our recent interactions and discoveries in this area (reported in Chapters 3-6, along with the **Appendixes** that follow this section).

1. Creating a standardized reading-level scale that allows unifying all leveled corpora under the same education criteria regardless of the source of the documents and the language.

2. Generating leveled corpora that contains resource in reading-levels beyond binary (simple/complex), enabling researchers to train and test novel models that can take of more fine grained readability information.

3. Designing strategies that enable more detailed readability assessment that go beyond simply generating a label for a text, including methods for detecting phrases that can pose specific challenges to the reader and providing reasoning on what the challenges involve.

4. Generating and testing existing cross-lingual embedding generating strategies on datasets that incorporate words that are not the most frequent in the vocabulary.

5. Finding novel strategies that can deal with the polysemy issue when mapping multiple word embeddings into the same space.

6. Researching novel approaches that can incorporate morphological information into word embeddings enabling better and more precise mapping across morphologically rich languages for both word embedding generation and machine translation.

7. Generating models that can take advantage of the fact that some languages follow a similar order when writing sentences.

8. Investigating strategies that can enable language transfer to typologically isolated languages such as Basque.

9. Designing novel strategies that enable better understanding and explanations of what internal neural network architectures are doing internally.

10. Improving the internal design of the proposed models in order to make them more scalable so that they can be used as part of production environments with larger data throughput requirements.

All the contributions we have presented in this dissertation were created based on one specific mindset we would like to transmit: *avoid having a prototype of the user when designing an application, think of the users as an group of different individuals with different capabilities and difficulties.* Research outcomes we described have set the foundations for IR for non-traditional users by avoiding language biases and enabling IR system adaptation so that it can respond to the reading ability of each user. However, there is still a long road to ahead in order to reach our goal of a democratized, unbiased, and accessible IR.

APPENDIX A

IS READABILITY A VALUABLE SIGNAL FOR HASHTAG RECOMMENDATIONS?

Abstract

We present an initial study examining the benefits of incorporating readability indicators in social network-related tasks. In order to do so, we introduce TweetRead, a readability assessment tool specifically designed for Twitter and use it to inform the hashtag prediction process, highlighting the importance of a readability signal in recommendation tasks.

Readability is a measure of the ease with which a text can be read. Usually represented by a number, it is an indicator used by teachers to classify and find appropriate resources for students. Several studies have demonstrated the benefits of using readability indicators in educational-related applications, such as book recommendation, text simplification, or automatic translation. However, applying readability indicators outside this environment remains relatively unexplored. Social networks could benefit from readability assessment. Twitter is a social network where users and texts are the main focus. For this reason, it is natural to think that for Twitter the ease with which a tweet can be understood by a user may affect his interest in it, and therefore influence actions taken, such as re-tweeting, giving a like or replying to the tweet.

The authors of [270] examined the degree to which the age of a user, a feature strongly correlated with readability, influences who people follow on Twitter, and demonstrated that Twitter users have a higher chance to follow people of similar age.

Using standard readability measures in text from Twitter, which constrains tweets to be of at most 140 characters in length, is not a trivial task. The lack of structure and shortness of those texts make standard natural language analysis techniques inefficient. With that in mind, we developed TweetRead, a novel readability assessment tool specifically designed for tweets. TweetRead takes advantage of social information, such as hashtags or mentions, for predicting the text complexity levels of tweets. Furthermore, in order to highlight the usefulness of such a tool in social networking environments, we developed a simple, yet effective, hashtag recommendation strategy that takes advantage of TweetRead-generated complexity levels of tweets to inform the hashtag recommendation process.

A.1 TweetRead

TweetRead's goal is to estimate readability of any given tweet T. TweetRead is based on a logistic regression technique[1] that fuses simple indicators describing T from different perspectives and determines its text complexity. The indicators considered by TweetRead include: (i) T's readability level, estimated using $Flesch$[2] [88], (ii) T's similarity with respect to word distributions generated from a large Twitter corpora C labeled by age groups, (iii) average readability of each hashtag h in T, computed based on the average readability levels estimated using Flesch of tweets in C that include h, (iv) average readability level of

[1] We empirically verified that among numerous supervised techniques, logistic regression was the most promising one.

[2] Flesch estimates the readability of a text/tweet t, by examining its length and the average length of terms in t.

users mentioned on T, estimated using Flesch on tweets written by mentioned users, and (v) frequency of mentions, emoticons, and hashtags in T.

Unlike traditional readability formulas that tend to map readability levels with school grades, to tailor TweetRead to the Twittersphere, we consider six levels of text complexity following Levinston's [156] adult development stages.

A.2 Hashtag Recommendation

Hashtags are character strings used to represent concepts on Twitter, starting with a # symbol. They are a core Twitter feature and serve classification and search purposes. Their unrestricted nature, however, creates difficulties, including the fact that the same concept can be represented by different hashtags, hindering the search process of a concept [269]. For example, tweets related to the Monaco Formula 1 Grand Prix can be searched using #monacoGP, #monacoF1GP or #monacoF1 retrieving different results. Hashtag recommendation aims at identifying suitable hashtags a user can include in his tweet to reduce the space of tags generated [269] and facilitate the ease with which he and other users can locate the corresponding tweet.

Given that (i) the scope of this paper is to validate the importance of considering a text complexity signal to enhance a recommendation task and (ii) multiple and increasingly complex systems have been developed for hashtag recommendation [98], we base our study on an existing framework for hashtag recommendation presented in [269]. Given a tweet T, the proposed framework identifies existing hashtags to recommend by following two major steps: (1) generate candidate hashtags by recommending hashtags present in similar tweets, using tf-idf based cosine similarity and (2) rank hashtags from retrieved candidate tweets using different strategies. The strategies presented in [269] include:

- **Similarity**. Prioritizes hashtags included on tweets that have the closes similarity to T, as estimated using the well-known tf-idf and cosine similarity measure.
- **Global popularity**. Prioritizes hashtags based on their respective frequency of occurrence on Twitter.
- **Local popularity**. Prioritizes hashtags based on their frequencies of occurrence among the tweets retrieved in response to T.

We enhance the proposed strategies by taking advantage of TweetRead, as follows:

- **TweetRead**. Prioritizes candidate hashtags that have the same or similar text complexity (estimated using TweetRead) with respect to T.
- **PopularityTweetRead**. Prioritizes hashtags based on their frequencies of occurrence among Twitter users whose reading abilities are estimated to match T's.
- **SimilarityTweetRead**. Prioritizes candidate hashtags based on their respective ranking scores computed using a linear combination of the scores yielded using Similarity and TweetRead.

A.3 Initial Assessment

In this section, we discuss an initial evaluation on TweetRead, as well as its applicability for suggesting hashtags.

Flesch	Spache	TweetRead
27%	31%	81%

Table A.1: Performance evaluation of TweetRead vs. baselines.

TweetRead. Given that readability of social content is an unexplored area, benchmark datasets that can be used for evaluation purposes are unavailable. For this reason, we built our own dataset. We initially gathered 172M tweets over an 8-month period using Twitter streaming API. We then eliminated tweets that did not include age references, which we needed to determine the age of each Twitter user in our dataset. In doing so we followed the framework presented in [270], which examines patterns such as “happy xth birthday". For the purpose of this experiment we assume that the age of people exactly corresponds to their readability level, and that every tweet written by a user will have the same readability level. As previously stated, we grouped labeled tweets into 6 age groups, which translates into a uniformly distributed dataset of 22k tweets with their corresponding readability levels. We followed a 10-cross-fold validation strategy and measured the accuracy of the predicted readability levels with respect to the ground truth. As shown in Table A.1, TweetRead significantly outperforms the baselines considered for this assessment: Flesch [88] and Spache [226], which are two well-known, traditional readability measures. The reported results demonstrate the need for readability strategies that examine information beyond standard text analysis, if they are meant to be successfully used in the social networking context.

Hashtag recommendation. For evaluating the strategies for hashtag recommendation presented in Section 3, we used the aforementioned dataset. We treated the hashtag of each corresponding tweet as the ground truth. In other words, for each tweet T, we generated the corresponding top-N hashtag recommendations and considered relevant the ones matching the hashtags in T. As in [269], we used the recall measure to evaluate performance and determine to which extend the correct hashtags were recommended within the top N generated suggestions. As shown in Figure A.1, even if readability on its own is not a sufficient factor to suggest hashtags, when combined in-tandem with other content-based and/or popularity strategies, it leads to the improvement of the overall hashtag recommendation process.

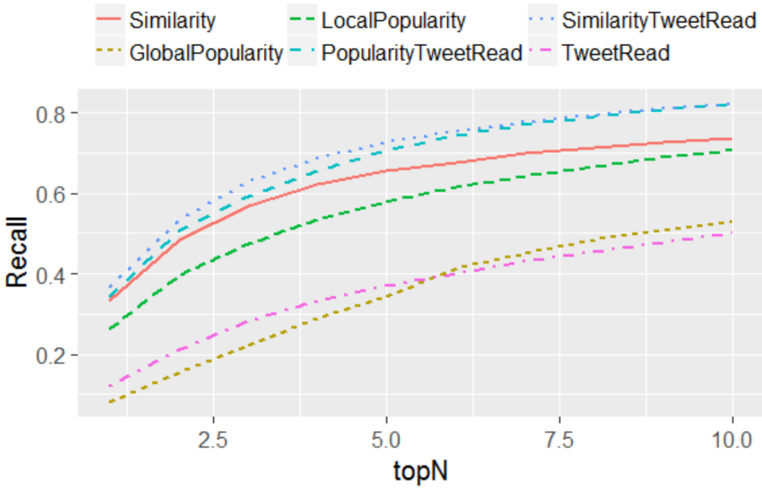


Figure A.1: Hashtag recommendation assessment.

A.4 Conclusion and Future Work

In this paper, we presented TweetRead, a novel readability assessment tool specifically designed to predict the readability of tweets. We also discussed the initial study conducted to demonstrate the benefit of using a readability signal in the hashtag recommendation task, which yielded promising results. In the future, we plan to explore other applications of readability in social networks, such as user recommendation, advertisement targeting or re-tweet prediction. We will also explore techniques to further enhance TweetRead and adapt it to other social networks beyond Twitter.

APPENDIX B

CAN READABILITY ENHANCE RECOMMENDATIONS ON COMMUNITY QUESTION ANSWERING SITES?

Abstract

We present an initial examination on the impact text complexity has when incorporated into the recommendation process in community question answering sites. We use Read2Vec, a readability assessment tool designed to measure the readability level of short documents, to inform a traditional content-based recommendation strategy. The results highlight the benefits of incorporating readability information in this process.

B.1 Introduction

Community question answering (CQA) sites allow users to submit questions on various domains so that they can be answered by the community. Sites like Yahoo! Answers, StackExchange, or StackOverflow, are becoming increasingly popular, with thousands of new questions posted daily. One of the main concerns of such sites, however, is the amount of time a user has to wait before his question is answered. For this reason, CQA sites depend upon knowledge already available and refer users to older answers, i.e., answers provided for previously-posted questions and archived on the site, so that users can get a more immediate response to their inquiries. This recommendation process has been extensively studied by researchers using a wide range of content similarity measures that go from the basic bag-of-words model to semantically related models, such as ranksLDA [214].

We argue that the recommendation process within CQA sites need to go beyond content matching and answer-feature analysis and consider that not every user has similar capabilities, in terms of both reading skills and domain expertise. User's reading skills can be measured by readability, which refers to the ease with which a reader can comprehend a given text. This information has been applied in the past with great success for informing tasks such as K-12 book recommendation [202], Twitter hashtag recommendation [169] and review rating prediction [79]. Yet, it has not made its way to CQA recommendations, where we hypothesize it can have a significant impact, given that whether the user understands the answer provided by a recommender can highly condition the value the user gives to the answer.

In this paper, we present an initial analysis that explores the influence of incorporating reading level information into the CQA recommendation process. With this objective in mind, we consider the *answer recommendation* task, where a user generates a query that needs to be matched with an existing question and its corresponding answer. We address this task by ranking question-answer pairs and selecting the top-ranked pair to recommend to the user. For doing so, we build upon a basic content-based recommendation strategy which we enhance using readability estimations. Using a recent Yahoo! Question-Answering dataset, we measure the performance of the basic recommender and the one informed by text complexity and demonstrate that readability has indeed an impact on user satisfaction.

B.2 Readability-based Recommendation

We describe below the strategy we use for conducting our analysis. Given a query q, generated by a user U, we locate each candidate answer C_a—along with the question Q_a

associated with C_a—that potentially addresses the needs of U expressed in q. Thereafter, the highest-ranked C_a-Q_a pair is recommended to U.

B.2.1 Examining Content

To perform content matching, we use an existing WordNet-based semantic similarity algorithm described by Li et al. in [160]. We use this strategy for computing the degree of similarity between q and C_{a}, denoted $Sim(q, C_{\mathrm{a}})$, and also the similarity between q and Q_a, denoted as $Sim(q, Q_a)$. We depend upon these similarity scores for ensuring that the recommended C_a-Q_a pair matches U's intent expressed in q. We use a semantic strategy, as opposed to the well known bag-of-words, to better capture sentence resemblance when sentences include similar, yet not exact-matching words, e.g. ice cream and frozen yogurt.

B.2.2 Estimating Text Complexity

To estimate the reading level of C_a and U (the latter inferred indirectly through q), we first considered traditional readability formulas, such as Flesch Kincaid [26]. However, we observed that these formulas were better suited for scoring long texts. Consequently, we instead use **Read2Vec**, which is a deep neural network-based readability model tailored to estimate complexity of short texts. The deep neural network is composed of two fully connected layers and a recurrent layer. Read2Vec was trained using documents from Wikipedia and Simple Wikipedia, and obtained a statistically significant improvement (72% for Flesch vs. 81% for Read2Vec) when predicting the readability level of short texts, compared to traditional formulas including Flesch, SMOG and Dale-Chall [26].

Given that the answer to be recommended to U should match U's reading ability to ensure comprehension, we compute the Euclidean distance between the corresponding estimations, using Equation B.1.

$$(q, C_a) = R2V(q) - R2V(C_{\mathrm{a}}) \tag{B.1}$$

where $R2V(q)$ and $R2V(C_{\mathrm{a}})$ are the readability level of q and C_a, respectively, estimated using Read2Vec.

B.2.3 Integrating Text Complexity with Content

We use a linear regression model[1] for combining the scores computed for each C_a–Q_a pair. This yields a score, $Rel(C_a, Q_a)$, which we use for ranking purposes i.e., the pair with the highest score is the one recommended to U.

$$Rel(C_a, Q_a) = \beta_0 + \beta_1 \mathit{Sim(q,\ C_a)} + \beta_2 \mathit{Sim(q,\ Q_a)} + \beta_3 \mathit{d(q,\ C_a)} \tag{B.2}$$

where β_0 is the bias weight, and β_1, β_2 and β_3 are the weights that capture the importance of the data points defined in Sections B.2.1 and B.2.2. This model was trained using least squares for optimization.

[1] We empirically verified that among well-known learning models, the one based on linear regression was the best suited to our task. We attribute this to its simplicity, which can better generalize over few training instances than most sophisticated models.

B.3 Initial Analysis

For analysis purposes, we use the L16 Yahoo! Answers Query to Questions **dataset** [253], which consists of 438 unique queries. Each query is associated with related question-answer pairs, as well as a user rating that reflects query-answer satisfaction on a [1-3] range, where 1 indicates “highly satisfied", i.e., the answer addresses the information needs of the corresponding query. This yields 1,571 instances, 15% of which we use for training purposes, and the remaining 1,326 instances we use for testing.

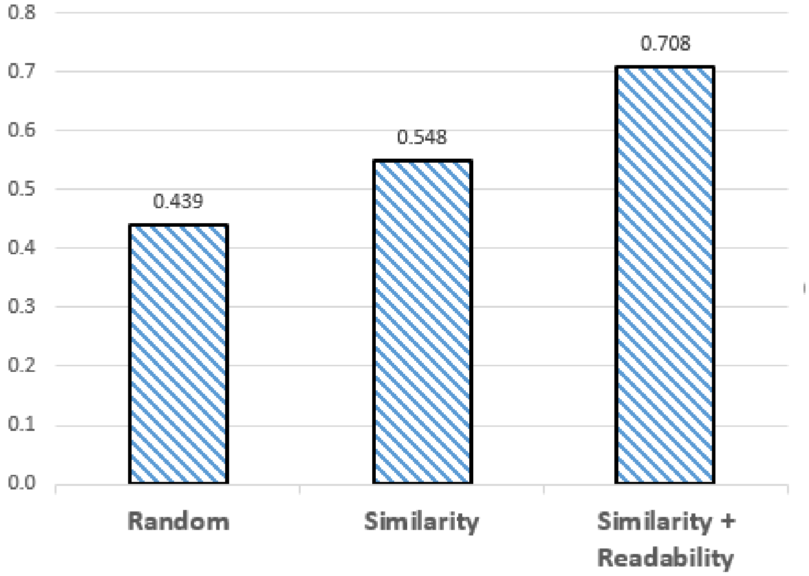
In addition to our *Similarity+Readability* recommendation strategy (presented in Section B.2), we consider two **baselines**: *Random*, which recommends question-answer pairs for each test query in an arbitrary manner; and *Similarity*, which recommends question-answer pairs for each test query based on the content similarity between the answer and the query, computed as in Section B.2.1.

An initial experiment revealed that regardless of the **metric**, i.e., Mean Reciprocal Rank (MRR) and Normalized Discounted Cumulative Gain (NDCG), the strategies exhibit similar behavior, thus we report our results using MRR.

As shown in Figure B.1, recommendations generated using the semantic similarity strategy discussed in Section B.2.1 yield a higher MRR than the one computed for the random strategy. This is anticipated, as *Similarity* explicitly captures the query-question and query-answer closeness. More importantly, as depicted in Figure B.1, integrating readability with a content-based approach for suggesting question-answer pairs in the CQA domain is effective, in terms of enhancing the overall recommendation process.[2] In fact, as per its reported MRR, *Similarity+Readability* positions suitable question-answer pairs high

[2] The weights learned by the model: $\beta_0, \beta_1, \beta_2, \beta_3$ = $2.26, 0.58, 0.20, 0.12$.

Figure B.1: Performance assessment based MRR using the Yahoo! Answers Query to Questions.



in the recommendation list, which is a non-trivial task, given that for the majority of the test queries (i.e., 83 %), there are between 5 and 23 candidate question-answer pairs.

B.4 Conclusions and Future Work

In this study, we analyzed the importance of incorporating readability level information into the recommendation process when it comes to the community based question answering domain. We treat the reading level as a personalization value and compare the readability level on an answer with respect to the reading abilities of a user, inferred through his query. We demonstrated that reading level can be an influential factor in terms of deciding the answer quality and can be used to improve user satisfaction in a recommendation process.

In the future, we plan to conduct a deeper study using other community question answering sites such as Quora or StackExchange. We also plan to analyze queries for additional factors, such as relative content-area expertise, to better predict a user's familiarity with content-specific vocabulary used on archived answers to be recommended. We suspect that readability and domain-knowledge expertise will be highly influential when the recommendation occurs on CQA sites like StackExchange, given the educational orientation of questions posted on the site.

APPENDIX C

LOOKING FOR THE MOVIE SEVEN OR SVEN FROM THE MOVIE FROZEN? A MULTI-PERSPECTIVE STRATEGY FOR RECOMMENDING QUERIES FOR CHILDREN

Abstract

Popular search engines are usually tuned to satisfy the information needs of a general audience. As a result, non-traditional, yet active groups of users, such as children, experience challenges composing queries that can lead them to the retrieval of adequate results. To aid young users in formulating keyword queries that can facilitate their information-seeking process, we introduce *ReQuIK*, a multi-perspective query suggestion system for children. *ReQuIK* informs its suggestion process by applying (i) a strategy based on search intent to capture the purpose of a query, (ii) a ranking strategy based on a wide and deep neural network that considers both raw text and traits commonly associated with kid-related queries, (iii) a filtering strategy based on the readability levels of documents potentially retrieved by a query to favor suggestions that trigger the retrieval of documents matching children's reading skills, and (iv) a content-similarity strategy to ensure diversity among suggestions. For assessing the quality of the system, we conducted initial offline and online experiments based on 591 queries written by 97 children, ages 6 to 13. The results of this assessment verified the correctness of *ReQuIK*'s recommendation strategy, the fact that it provides suggestions that appeal to children and *ReQuIK*'s ability to recommend queries that lead to the retrieval of materials with readability levels that correlate with children's reading skills.

C.1 Introduction

As one of the largest communities that search for online resources, children are introduced to the Web at increasingly young ages [63]. However, popular search tools are not explicitly designed with children in mind nor do their retrieved results explicitly target children. Consequently, many young users struggle in completing successful searches, especially since most search engines do not directly support, or offer weak support, for children's inquiry approaches [92]. As stated in [245], this is an important issue to address since early experiences influence skill development in making proper use of resources for personal and educational growth.

As described in the book Search Engine Society, "Children growing up in the 21^{st} century have only ever known a world in which search engines could be queried, and almost always provide some kind of an answer, even if it may not be the best one" [113]. Even though children, as inexperienced users, struggle with describing their information needs in a concise query [67], they still expect search engines to retrieve relevant information in response to their requirements, or at least suggest better choices for a successful search. As part of their capabilities, search engines often suggest[1] queries to aid users in better defining their information needs. In fact, a recent study conducted by Gossen et al. [102] shows that children pay more attention to suggested queries than adults. Unfortunately, these suggestions are not specifically tailored towards children and thus need improvement [243]. While multiple query suggestion modules have been developed to automatically generate queries that capture users' needs [10, 272], only a small number of them specifically target children. To address this problem, along with a need for more children-related tools,

[1] *Suggestion* and *recommendation* are used interchangeably in this manuscript.

we introduce *ReQuIK* (**Re**commendations based on **Qu**ery **I**ntention for **K**ids), a query suggestion module tailored towards 6-to-13 year old children.

The main goal of *ReQuIK* is to provide query recommendations that explicitly consider diverse and ambiguous users' information needs. Prior to generating recommendations for a given child-initiated query, *ReQuIK* takes advantage of the *search intent module* presented in [63] , which is used to capture the intended meaning of the query. In doing so, *ReQuIK* can deal with long natural language queries or queries that include common patterns children use when searching the web, which are difficult for search engines to process and properly handle. Even when the search intent of a query is identified, it is not enough to trigger the retrieval of suitable materials for each user, since the interests of children can vary depending of their age. To capture a wide range of potential suggestions, *ReQuIK* emulates a popular *query generation* strategy. Thereafter, *ReQuIK* identifies suitable suggestions by using a multi-perspective approach based on raw text analysis and a number of textual traits specifically associated with children content. These traits analyze usage of children words, popular culture terms, entities and diminutives in queries. By applying a multi-perspective approach based on deep learning, the proposed query suggestion module is able to learn distinctive characteristics that portray adults and children queries, and use that knowledge to predict which queries are the most child-friendly among the ones in a candidate set. To guarantee diversity among the recommended queries, *ReQuIK* uses a content similarity strategy that groups together queries that are topically similar and excludes suggestions that refer to the same topic, i.e., queries that would retrieve the same type of resources. We are aware that suggested queries could retrieve resources that children may not easily comprehend due to their high reading difficulty [198]. In order to minimize this situation, *ReQuIK* prioritizes query suggestions that will potentially lead to easier-to-read resources.

Due to the lack of datasets that capture children search activities, we dedicated research efforts to creating one. To do this, we deployed an ad-hoc search framework which interacts with the Bing *api* and facilitates the archival of search sessions. We conducted an experiment with 97 children, ages 6 to 13, who were given research/informational and factual search tasks by their teacher. As a result, we gathered close to 600 queries, which we used for development and evaluation purposes. Thereafter, we verified the validity of *ReQuIK* based on both offline and online experiments, using the aforementioned dataset, which demonstrate that not only does *ReQuIK* suggest queries that are children oriented, but it also leads to resources that are of the adequate reading level.

To the best of our knowledge, *ReQuIK* is the only available system that can be coupled with existing engines to generate query recommendations for children, favoring those that can lead to easier-to-read resources. *ReQuIK* suggests queries that initiate the retrieval of child-related topics and materials, which can lead to improving search engines' performance. The design of the proposed tool explicitly considers different patterns children use while searching the Web to adequately capture the intended meaning of their original queries. For example, if a child submits the query *"elsa"*, *ReQuIK* aims to prioritize query suggestions such as *"elsa coloring papers"* or *"elsa dress up games"* that correlate better with topics of interest to children rather than *"elsa pataky"*, as suggested by Google[2], which is more appealing to mature users. Other contributions of our work include (i) a novel ranking model inspired by a deep-and-wide architecture that, while successfully applied for ranking purposes [40, 268], has never been used in the query suggestions domain, (ii) a strategy to

[2] As verified on May 19, 2017.

overcome the lack of queries written by children by taking advantage of general purpose children-oriented phrases, and (iii) the aforementioned newly created dataset, which will be made available to the research community[3].

C.2 Related Work

Creating an appropriate query that leads to retrieving relevant information is challenging for young users. Previous studies state that the performance of a web search engine is poorer when retrieving documents in response to queries targeting information for children than for queries oriented to content of more traditional users [68]. Query recommendations can help children by providing queries that can be used to initiate a search.

Current research on children's query suggestions is limited, with a simple query to ACM Digital Library for “children query suggestions" or “children query recommendations" retrieving five distinct research works from among the top-20 results. Existing research includes the one conducted by Duarte et al. [67], who rely on a bipartite graph constructed using tags and URLs to suggest children queries. The authors enhanced their proposed strategy, as discussed in [68], by considering topical and language modeling features, such as a topic-sensitive Page Rank and a children-related vocabulary distribution. Besides examining tags assigned at Delicious to retrieve web pages, Eickhof et al. [72] consider high-level semantic categories (inferred from Wikipedia and the DMOZ.org taxonomy) associated with tags, and treat them as expansion terms. The aforementioned approaches, however, rely primarily on tags to make their suggestions, which can be poorly defined due to the lack of quality control on user tags which can be inherently noisy. Furthermore, these

[3] The dataset can be found in https://doi.org/10.18122/B2WQ5T.

tags are often provided by adults, instead of children, which explains why the vocabulary used to describe online resources for children might not correlate with the terms used by children. The work conducted by Vidinli and Ozcar [247] focuses on suggesting queries in within an educational search environment. The proposed strategy analyzes a number of features to determine the most suitable queries, among the candidates, that should be recommended, given a child-generated query. Unfortunately, the majority of the features are computed as a result of query-log examination, which is a constraint as query logs generated by K-12 students are rarely accessible.

To the best of our knowledge, the studies done in [134, 219, 257] are the only ones that do not use tags or query logs for generating children query suggestions. Instead, the authors in [219] use bigrams extracted from websites that contain text generated by children and Simple.Wikipedia.org, a collection of documents written for users whose second language is English. As opposed to the strategy in [219], which depends upon a pre-defined set of topical categories, *ReQuIK* relies on a dynamic clustering to ensure diversity of recommended queries. The module presented in [257] creates query suggestions that are semantically different but conceptually similar to a child-initiated query. To do so, the authors consider result set overlap generated by pairs of queries (a given query and a possible suggestion) and term overlap between the queries, and prioritize suggestions including n-grams in Simple Wikipedia or include terms in a pre-defined children vocabulary.

Even if with a different purpose, the work of Eickhoff et al. [71] is similar ours, given that they also aim at distinguishing between children and non children content. In their work, the authors use aesthetic features of websites as discriminators of children-related content, features that are not useful for classifying queries.

Similar to the approach in [134], *ReQuIK* emulates Ubersuggest's query generation strategy to create a set of queries to recommend. However, while the query suggestion strategy discussed in [134] depends on a regression model that combines multiple features, such as children vocabulary, phrasing patterns, popular culture terms related to children, and the popularity of the terms among children, *ReQuIK* adopts a multi-perspective suggestion approach that considers a different and larger set of traits to infer if a query is child-related, as well as content of the query itself. What distinguishes *ReQuIK* the most, among its counterparts, is its ability to simultaneously combine text pattern analysis as well as varied query traits to identify suitable child-related query suggestions.

C.3 ReQuIK

In this section, we describe the design of *ReQuIK* (see pseudocode in Algorithm 1). Along with the description of each strategy used in *ReQuIK*, we provide a step-by-step example (denoted **R.I.A.**, ReQuIK in action) using Q_E*: "I want the trol song"*, a query written by a child, which is also part of the sample introduced in Section C.4.1. This running example aims to further showcase the practical application of *ReQuIK*. Note that for Q_E, Google neither offers possible query suggestions nor leads to the retrieval of resources that explicitly target younger audiences, as illustrated in Figure C.1, which further aids our case in advocating for the existence of query recommendation strategies solely for children.

C.3.1 Search Intent

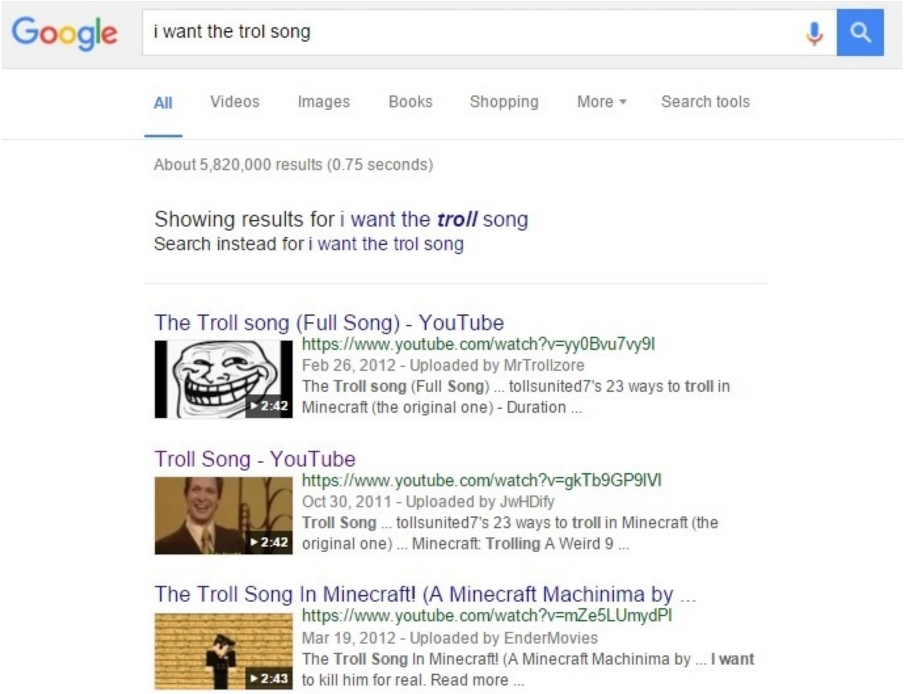
As described by Bilal et al. [29], to adequately serve children, search engines must address the fact that children are seldom successful in formulating succinct queries. In

Algorithm 1 Generating Query Suggestions

```
Input:  A query Q written by a child, a trained ranking model
RM, wordId dictionary WD, number of suggestions to generate k
candidates = empty set
scoredCandidates = empty list
suggestions = empty list
Q' = searchIntent(Q)
candidates = generateCandidates(Q')
for each candidate CQ in candidates do
   features = generateFeatures(CQ)
   wordIds = getWordIDs(CQ,WD)
   score = calculateScore(wordIds, features, RM)
   scoredCandidates = scoredCandidates + <CQ, score>
end for
sort(scoredCandidates)
for each candidate <CQ,score> in scoredCandidates do
   if suggestions is empty then
      suggestions = CQ
   else
      if CQ is not similar to any query in suggestions and readability(CQ)<8 then
         suggestions = suggestions + CQ
      end if
   end if
   if |suggestions| $\geq$ k then
      break
   end if
end for
```

fact, researchers have observed that children tend to use long (natural language) queries, as opposed to keyword queries, when performing online searches [65]. Unfortunately, the longer the query, the less likely a web search engine is to retrieve relevant resources in response to it, which can be very frustrating for young users [65]. Furthermore, children tend to misspell words and use writing patterns that differ from those of adults. For example, children can include the word “amazzzzing" instead of “amazing" in a query to emphasize

Figure C.1: Screenshot of documents retrieved by Google for the query *“I want the trol song"*



i want the trol song

About 5,820,000 results (0.75 seconds)

Showing results for i want the ***troll*** song
Search instead for i want the trol song

The Troll song (Full Song) - YouTube
https://www.youtube.com/watch?v=yy0Bvu7vy9I
Feb 26, 2012 - Uploaded by MrTrollzore
The **Troll song** (Full **Song**) ... tollsunited7's 23 ways to **troll** in Minecraft (the original one) - Duration ...
▸2:42

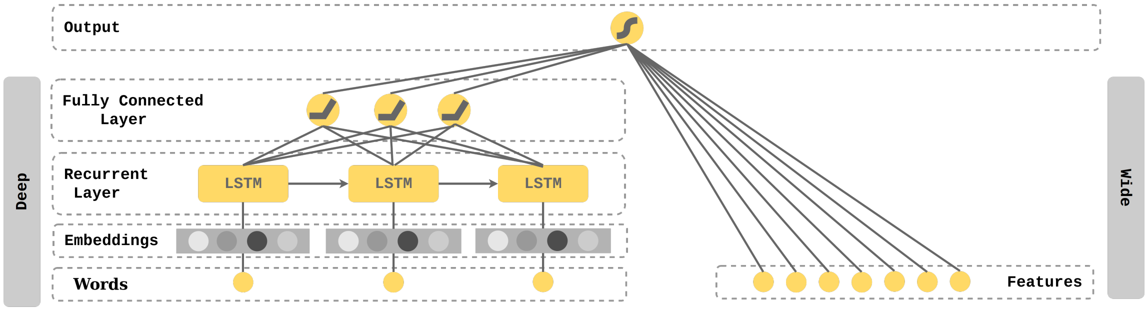
Troll Song - YouTube
https://www.youtube.com/watch?v=gkTb9GP9IVI
Oct 30, 2011 - Uploaded by JwHDify
Troll Song ... tollsunited7's 23 ways to **troll** in Minecraft (the original one) ... Minecraft: **Trolling** A Weird 9 ...
▸2:42

The Troll Song In Minecraft! (A Minecraft Machinima by ...
https://www.youtube.com/watch?v=mZe5LUmydPI
Mar 19, 2012 - Uploaded by EnderMovies
The **Troll Song** In Minecraft! (A Minecraft Machinima by ... **I want** to kill him for real. Read more ...
▸2:43

something is really amazing. To best satisfy children needs, *ReQuIK* relies on *QuIK*, the search intent module for children presented in [63], which addresses common patterns detected in children writing, including: use of diminutives, exaggerated and trendy terms as well as higher percentage of misspellings when compared to adult users. In doing so, it transforms an initial query Q into a simplified keyword query Q' that (i) captures the information need meant to be expressed by a child and (ii) can be adequately processed by search engines.

R.I.A. The search intent module employed by *ReQuIK* transforms Q_E into Q'_E “troll song", which suitably captures the intended meaning of Q_E, i.e., troll song from the popular movie Frozen. The search intent module solves two problems observed in Q_E: (i) removing terms that are superfluous for capturing the meaning of the query, i.e. *"I want the"*, and (ii) fixing the spelling error on *"trol"*.

Figure C.2: Ranking Model Architecture



C.3.2 Candidate Generation

Having identified the information need of a young user expressed in a query Q and created a shorter, more concise query *Q'*, *ReQuIK* generates a set of candidate queries, i.e., queries that could possibly be suggested to a user, by emulating the algorithm of Ubersuggest[4] [1], a popular query suggestion tool based on Google auto complete API. The advantage of adopting such a strategy is that it quickly finds keywords based on what users search for on the Internet, creating multiple possible queries [19]. Bypassing the use of static query logs or probabilistic models allows *ReQuIK* to offer up-to-date candidate queries, since the aforementioned auto-complete strategy is constantly updated by online search trends.

R.I.A. Submitting Q'_E to the candidate generator creates more than 200 candidate suggestions, including queries related to children, such as *"troll song from dora"* and *"troll*

[4] Ubersuggest queries Google autocomplete API multiple times with the initial query followed by each letter of the alphabet in order to retrieve multiple query candidates.

song frozen" as well as queries that seem intended for more mature users, such as *“troll song no copyright"* and *“troll song hitler"*.

C.3.3 Ranking model

Not all the candidate queries generated in Section 3.2 are necessarily targeted towards children’ needs, reading abilities, and interests. Consequently, to identify suitable suggestions among the candidates, *ReQuIK* takes advantage of a novel model (see Figure C.2) that we created by combining two architectures, a *deep model* and a *wide model*, inspired by the app recommendation model recently developed by Cheng et al. [40]. We discuss below insights and benefits of each architecture when applied to child-related tasks, as well as how we combine them into a single model for ranking purposes.

Wide Model: Learning what makes a query child-like

The wide model incorporates a set of manually-created features that are meant to capture traits often observed on child-related queries. These features are a result of an extensive analysis of children-related sentences (sampled from online sources for children). The wide model is composed of a vector $x_{feats} \in \mathbb{R}^f$, consisting of f features. Those features along with their computations are shown in Table C.1, while their descriptions are provided below. Due to space constraints, Table C.1 only includes the final set of features considered by *ReQuIK*. However, it is worth mentioning that we conducted an empirical analysis to identify the best subset of features to be used as a part of the ranking process. For example, features based on readability levels calculated by readability formulas such as Gunning FOG [6] and Dale-Chall [36] were overlooked in favor of the features based on an enhanced version of Spache [226] readability formula, which more adequately captures the level of difficulty of web resources.

Trendy Terms: Queries generated by young users contain a large amount of children-related trendy terms, such as names of movies, popular singers, computer games etc. By examining the presence of such terms in a candidate query Q, *ReQuIK* is able to determine if Q is appealing to children or not. (See Table C.1-A.)

Entities: Based on a conducted analysis on queries generated by children, we identified a considerable use of entities (e.g., person, location, organization, etc.). For this reason, *ReQuIK* examines the presence of entities as one of the criteria to help decide how likely Q is related to children's interest. (See Table C.1-B.)

Children Dictionary: In developing *ReQuIK*, we created a children dictionary using text collected from child-related websites. The use of such a collection of terms is of crucial importance since it best describes appropriate terms children use. Based on this, we concluded that young users use a narrower and unique vocabulary when expressing their needs. Therefore, we analyze the average frequency of terms in Q that are included in our children dictionary to enable *ReQuIK* to decide if the Q is tailored towards adults or children. Note that, this dictionary will be made available to the research community on this project's website upon the publication of this manuscript. (See Table C.1-C.)

Flesch-Kincaid: Another criteria that targets a simplicity of Q is based on the well-known readability formula, i.e., Flesch-Kincaid [88], which provides the grade level of Q and enables *ReQuIK* to decide if a young user can comprehend Q. (See Table C.1-D.)

Enhanced Spache: As a complement to the Flesch-Kincaid readability assessment, *ReQuIK* uses a feature based on an enhanced Spache formula. As an enhancement of a regular Spache [226] to obtain a greater accuracy, we expanded existing Spache dictionary of common words with children dictionary previously created. The combination of two dictionaries was necessary for considering more reliable and up-to-date trends of kids' vocabulary. This criteria increases the level of detection of words used by children in Q and enables *ReQuIK* to differentiate adult and child-related candidate queries. (See Table C.1-E.)

Difficult Terms: *ReQuIK* uses a criteria generated based on the frequency of non-children [5] terms in Q to further separate adult and child-like queries. (See Table C.1-F.)

Feature ID	Feature Name	Description
A	Trendy Terms	$tt(q) = \frac{\sum_{i=1}^{\lvert q\rvert} t\left(q_i\right)}{\lvert q\rvert}, t(q_i) = \begin{cases} 1, & \text{if } q_i \text{ is recognized as a trendy term} \\ 0, & \text{otherwise} \end{cases}$ To determine if q_i is a child-related trendy term, *ReQuIK* examines its existence in children related pages on well-known websites, such as Amazon.com and CommonSenseMedia.org.
B	Entities	$et(q) = \frac{\sum_{i=1}^{\lvert q\rvert} e\left(q_i\right)}{\lvert q\rvert}, e(q_i) = \begin{cases} 1, & \text{if } q_i \text{ is recognized as an entity by Stanford NER tool [178]} \\ 0, & \text{otherwise} \end{cases}$ To determine if q_i is an entity, *ReQuIK* uses well-known CoreNLP tools.
C	Children Dictionary	$ct(q) = \frac{\sum_{i=1}^{\lvert q\rvert} c\left(q_i\right)}{\lvert q\rvert}, c(q_i) = \begin{cases} 1, & \text{if } q_i \text{ is included in a children dictionary} \\ 0, & \text{otherwise} \end{cases}$ We created our own children dictionary, comprised of 100,000 non-stop lemmatized terms, extracted from texts retrieved from a sample of various children-related websites.
D	Flesch-Kincaid	$FK\big(q\big) = 0.39\left(\frac{totalwords}{totalsentences}\right) + 11.8\left(\frac{totalsyllables}{totalwords}\right) - 15.59$ In our case *total words* is the total number of words in q_i and *total sentences* is always equal to 1.
E	Enhanced Spache	$ES(q) = (0.121 \times AvgLengthOfq) + (0.082 \times NumberOfUniqueUnfamiliarWordsInq)$ We implemented Enhanced Spache formula by updating the existing dictionary of common words with the children dictionary in C.
F	Difficult terms	$dt(q) = \frac{\sum_{i=1}^{\lvert q\rvert} d\left(q_i\right)}{\lvert q\rvert}, d(q_i) = \begin{cases} 1, & \text{if } q_i \text{ is recognized as a difficult term} \\ 0, & \text{otherwise} \end{cases}$ *ReQuIK* treats q_i as a difficult term if it is not included in the children dictionary, or Spache dictionary of common words, or trendy terms list in A

Table C.1: Criteria description, where q represents a query, q_i is the i^{th} non-stop word, lemmatized term in q, and $|q|$ is length of non-stop-words in q. Each computed criteria score is normalized to fit a 1-5 scale

[5] We treat as “non-children" terms that do not appear or have low frequency on our children dictionary.

Deep Model: Learning from text

Manual analysis of queries can lead to identifying distinctive features such as the ones mentioned in Section C.3.3, however, some patterns can be impossible to detect upon simple observation or empirical analysis. Deep learning enables learning directly from raw data, so that new, unexpected features can be inferred automatically allowing the model to grasp patterns that humans are unable to find. The deep neural network is composed of one input layer and 3 hidden layers. The output layer is shared with the wide model and is therefore described in Section C.3.3. Each of the other layers are described as follows:

Input Layer The input of the neural network is represented as $x_{words} \in \mathbb{Z}^k$ where x_{words} is a vector k identifies representing a sequence of words. After analyzing the distribution of length of the queries we gathered, we fixed k at 15, a sufficient length to capture 95% of queries in our sample.

Embedding Layer The embedding layer's role is to convert each word identifier into a dense representation that will capture the semantics of the word. For doing so we define an embedding function $Q : \mathrm{wordId} \rightarrow \mathbb{R}^{\alpha}$, where α is the embedding size, that converts a word id to an embedding of length α. This function is based on a lookup table $S \in \mathbb{R}^{v \times \alpha}$, where v represents the vocabulary size. The embedding function Q returns a row from S that corresponds to the provided word id. This function is applied to all the word ids in the input sequence creating a new matrix $H_1 \in \mathbb{R}^{k \times \alpha}$ that will be the input of the next layer. The matrix S is initialized using a random uniform distribution within $[-1, 1]$ and will be trained together with the weights of the neural network.

Recurrent Layer Recurrent neural networks have been successfully used for processing sequential information [149]. A text document can be seen as a sequence of words, where each word depends on information provided by previous words, making it adequate for a recurrent neural network. The third layer of *ReQuIK*'s ranking deep neural network takes advantage of Long Short Term Memory cells (LSTM) [120], a recurrent cell specially suited for textual documents given its capability to remember long term information. Given a word embedding and a state vector $l_s \in \mathbb{R}^\beta$, where β refers to the number of LSTM units, each LSTM cell generates a output $l_{out} \in \mathbb{R}^\beta$. These outputs are concatenated to create a vector $h_2 \in \mathbb{R}^{\beta * k}$, which will be the input of the next layer.

Fully Connected Layer A fully connected layer is one of the most common layers in a neural network. This layer computes a weighted sum over all the outputs of the previous layer. More precisely, given the vector h_2 produced by the LSTM layer, this layer computes the following operation:

$$h_{deep} = \text{relu}(Wh_2 + b) \tag{C.1}$$

where $h_{deep} \in \mathbb{R}^\gamma$ is the output of this layer, $W \in \mathbb{R}^{(\beta * k) \times \gamma}$ is a matrix of weights, $b \in \mathbb{R}^\gamma$ a bias vector. γ is a parameter that determines the number of neurons in the layer and $relu$ refers to Rectified Linear Unit [148] which corresponds to the activation function that is applied to the result of the weighted sum.

Output Layer: Combining both models

The last layer of the ranking model is the one responsible for combining the aforementioned deep and the wide models. This enables *ReQuIK* to incorporate the benefits of both a wide and a deep model, being capable of learning patterns from text automatically, while also using human crafted features that consider traits related to children queries. For doing this we first concatenate h_{deep} and x_{feats} to create a new vector $h_{comb} \in \mathbb{R}^{\gamma+f}$. Similar to the fully connected layer in the deep model, a weighted sum of all the values is computed, to create an output:

$$y' = \text{sigmoid}(W_{comb} h_c comb + b_{comb}) \tag{C.2}$$

where $y' \in \mathbb{R}^c$ is the prediction of the neural network, $W_{comb} \in \mathbb{R}^{(\gamma+f)\times c}$ is a weight matrix, $b_{comb} \in \mathbb{R}^{\gamma+f}$ a bias vector, c the number of prediction classes (2 in our case) and a sigmoid as activation function. Note that the prediction vector y' is composed of real values, enabling to use the same model for both prediction and as a scoring function for ranking the candidate queries.

Training

To produce relevant predictions, a neural network needs to be trained. This process involves fitting several variables that include, weights, biases and embedding values. For fitting those values a loss function is minimized using input/output pairs from a training dataset. For doing so, we take advantage of Cross Entropy function as a loss function, defined as follows:

$$H_{y'}(y) = -\sum_{i} y'_i \log(y_i) \tag{C.3}$$

where y' is the prediction created by the neural network and y is the target ground truth using one-hot encoding. For minimizing the error, we took advantage of the Adaptive Movement Estimation (Adam) [142] optimization technique.

The performance of a neural network is affected by its parameters. To identify the optimal α, β, and γ, we sweep possible values and found that the best combination for this task is $\alpha = 128; \beta = 128; \gamma = 128$. Thus, this is the parameter set used in all the experiments reported in this paper.

R.I.A. Using its ranking strategy, *ReQuIK* assigns a rating to each of the candidate query recommendations generated for to Q'_E. Based on the predicted ratings, queries like *“troll song from dora"* and *“troll song frozen"* are prioritized over queries such as *“troll song no copyright"*, which more likely better capture topics of interest for more mature audiences.

C.3.4 Readability

We observed that among the list of top-N child-related queries, not all of them lead to the retrieval of documents matching the reading skills of 6 to 13 year old children. This is of high importance for the recommendation process, since children are not able to comprehend resources above their reading capabilities. To determine what queries will most likely retrieve documents with suitable reading levels, *ReQuIK* applies the Flesch-Kincaid formula [88]. Queries that lead to the retrieval of resources associated with readability levels that are greater than readability levels expected for a child are excluded from the list of queries to be suggested[6]. Note that we treat 8 as an appropriate children grade level since it corresponds to reading level of 13 year old user.

Relying on this filtering strategy enables *ReQuIK* to identify suitable suggestions based not only on query content itself, but also on the readability levels of documents that would potentially be retrieved using those queries to initiate the search process.

R.I.A. *ReQuIK* further filters candidate queries to ensure that recommendations shown to its users most likely trigger the retrieval of documents they can understand. Based on the average readability scores of the top-N documents retrieved in response to *"troll song frozen"* and *"troll song no copyright"*, which are 6.7 and 11.3, respectively, *ReQuIK* retains the former and excludes the latter from the set of possible query recommendations.

C.3.5 Diversity

To guarantee that generated suggestions cater to diverse user interests, *ReQuIK* excludes from the set of top-N query recommendations candidate suggestions that are, to a degree, similar to each other. *ReQuIK* applies the Semantic Similarity algorithm developed by Yuhua Li et al. [160], which provides WordNet-based scores that are used to determine if any two suggestions are semantically the same, i.e., would trigger the retrieval of similar resources. In this context, two suggestions are treated as similar if their similarity score is above a threshold. By applying this topical filtering strategy, we select top-k[7] diverse suggestions from the ranked list generated in Section C.3.4. For doing so, the first suggestion S_1 is always included in the final set of suggestions. Each subsequent candidate suggestion

[6] To determine which candidate query cq potentially retrieves resources with reading levels above the expected ones, we compute and average the reading levels of the top-3 resources retrieved in response to cq.

[7] We set $k = 4$ to emulate the number of suggestions often offered by search engines.

S_n is compared to the suggestions already in the final set. Thereafter, S_n is included only if it yields similarity scores of at most 0.7[8] with respect to the already-selected suggestions.

R.I.A. In the last step of its process, *ReQuIK* selects not only highest-rated and readability-level suitable queries, but also queries that offer topical diversity and thus target a wide range of users. For example, using its similarity-based filtering, *ReQuIK* treats *“troll song frozen"* and *“troll song frozen movie"* as highly similar and excludes the latter from the set of suggestions to be presented to the users. Lastly, the final set of suggestions generated by *ReQuIK* in response to the initial child-query Q_E includes *“troll song frozen"* and *“troll song from dora"*. These suggestions not only capture different information needs but do so in a keyword fashion that enable search engines to retrieve more relevant and suitable results for 6-13 year olds.

C.4 Experimental Results

In this section, we detail the results of the offline and online studies conducted to demonstrate the correctness of *ReQuIK*'s methodology and the relevance of its generated query recommendations.

C.4.1 Evaluation Framework

We discuss below the datasets and query suggestion strategies used for comparison purposes.

[8] We experimentally set 0.7 as the similarity threshold.

Dataset & Other Resources

Obtaining children-related data is not a simple task, especially due to children protection regulations that make sharing this type of data highly restrictive. While other researchers have used queries extracted from existing query logs like the popular AOL [66], the extraction strategy may be limited and not always identify queries written by children, as some writing patterns are common among both children and adult queries. To address this issue, given that neither datasets that can be used to evaluate query recommendations for children nor query logs comprised only of children queries are publicly available, we created our own dataset[9].

Search Environment. In order to construct a dataset, we developed an online search framework that emulates the behavior and appearance of Google, and enables us to gather search sessions[10] of children and archive information such as: query typed, selected query suggestions (if available), clicked URLs, and timestamps. We made the appearance of this framework similar to that of a popular search engine, given children's preference to use well-known engines, as opposed to the counterparts designed for them, e.g., KidzSearch. The search framework contains an initial page that a teacher can configure based on the grade of his/her class.

Gathering Queries. In order to get children to use the search framework, we collaborated with elementary schools in the Idaho (USA) area. We asked K-9 teachers to propose their students information discovery tasks, for which they used our framework to create queries. For determining the type of tasks we followed the strategy used by Gwizdka et al. [111]

[9] The process of gathering and archiving children queries was supervised by the Institutional Review Board at Boise State University in order to ensure children-related ethical and legal concerns were met.

[10] Information obtained from a child's search session is anonymous.

and included both: research/informational tasks such as *Find information about fire belly toads* or *Find information about tigers* and factual tasks such as *How long do toads live* or *When does summer start*. Each teacher started the class with specific questions, however, children were later allowed to find information about things of their interest. A total of 97 children between the ages of 6 and 13 participated in the study, generating 591 unique queries.

Even though *ReQuIK*'s objective is to recommend children queries, for development and assessment purposes we also required a set of non-children queries. Thus, the resulting dataset, denoted ReQ_{qs}, includes the aforementioned 591 queries labeled *children-queries* and 591 queries randomly selected among the ones in the Yahoo's query log dataset [263] labeled *adult-queries*.

Other Data Sources. Due to the limited amount of children query-logs in ReQ_{qs}, we gathered ReQ_{corp}, a sample of 1,061,666 sentences for development and training purposes. In creating ReQ_{corp}, we extracted sentences from websites oriented to children, including Dogo [59], Spaghetti [227], Toy Insider [238], Raising Children [208], Kidzvuz [140], Kids-in-mind [138], and Edutaining-kids [70]. We also included in ReQ_{corp} sample sentences from Wikipedia in order to provide negative, i.e., non-children, examples.

Comparison Strategies

In the following sections, we discuss the results of a number of experiments conducted to demonstrate the need for a query suggestion modules specifically designed with young users in mind and showcase the correctness and effectiveness of *ReQuIK*. To give context to such results, we compared the performance of *ReQuIK* with that of a number of baseline and state-of-the-art strategies. We examine (the queries suggested by) $Google$, $Yahoo$, and $Bing$, given that children favor well-known search engines when performing information discovery tasks [29]. Given that we argue for the need of techniques tailored for children, we also consider a number of search engines designed exclusively for children, including: $AskKids$[18], $Kidzsearch$[139], $Ipl2$[126], $KidRex$[137], and $SweetSearch$ [231]. We also include in our experiments CQS [219] (discussed in Section C.2), which is a state-of-the-art alternative for generating queries tailored to children.

C.4.2 Usefulness of the Search Intent Module

In addition to the results reported in [63], which demonstrate the performance of the search intent module, we conducted an experiment for measuring the impact this module has for query recommendation. For doing so, we submitted each of the child-written queries in $\textit{ReQ}_{qs}$ to a number of popular search engines. Thereafter, we determined for which of these queries, the corresponding search engine provided suggestions.

As shown in Table C.2, there is a consistent decrease in the number of children queries that can be handled by search engines, in terms of providing suggestions given a child-initiated query. This fact is evidenced in both commercial search engines and state-of-the-art systems such as CQS. Unlike its counterparts, *ReQuIK* is able to provide suggestions for 94% of the queries considered in this study, which is a clear indicator that taking advantage of a search intent module is beneficial not only to enhance the document retrieval process [63] but also the query recommendation process. While in Table C.2 we report results on search engines favored by children, it is important to mention that those percentages remain so, even on search engines designed specially for children. This is anticipated, since many of the search tools for children are powered by Google's safe search (e.g., Ask.forkid.co) or do not offer query suggestions (e.g., Cybersleuth-kids.com and Kiddle.co). CQS is only able to generate suggestions for 57% of the children-generated queries in ReQ_{qs}, due to its candidate generation strategy that struggles with identifying suggestions for queries longer than four-grams.

Google	Bing	Yahoo!	Kidsearch	CQS	ReQuIK
46%	36%	65%	76%	57 %	94%

Table C.2: Percentage of queries that trigger a recommendation in each compared system

C.4.3 Effectiveness of the Recommendations

As previously mentioned, there is a lack of datasets comprised of children queries and gathering and sharing children data to create such datasets is not trivial due to time and privacy concerns. Unfortunately, queries on ReQ_{qs} are not sufficient to train a deep neural network (i.e., ranking) model. In order to amend this issue, we train *ReQuIK*'s ranking model using children and non children oriented sentences in ReQ_{corp}. We hypothesize that these sentences are similar enough to a search query, permitting the modeled knowledge to be transferable to the query suggestion context. Therefore, to showcase the correctness of *ReQuIK*'s ranking strategy, we conducted two experiments. The first one measures the performance of the model in predicting whether a sentence is oriented for children or adults, and the second one measures how well this knowledge is transferred to the task of ranking children query suggestions.

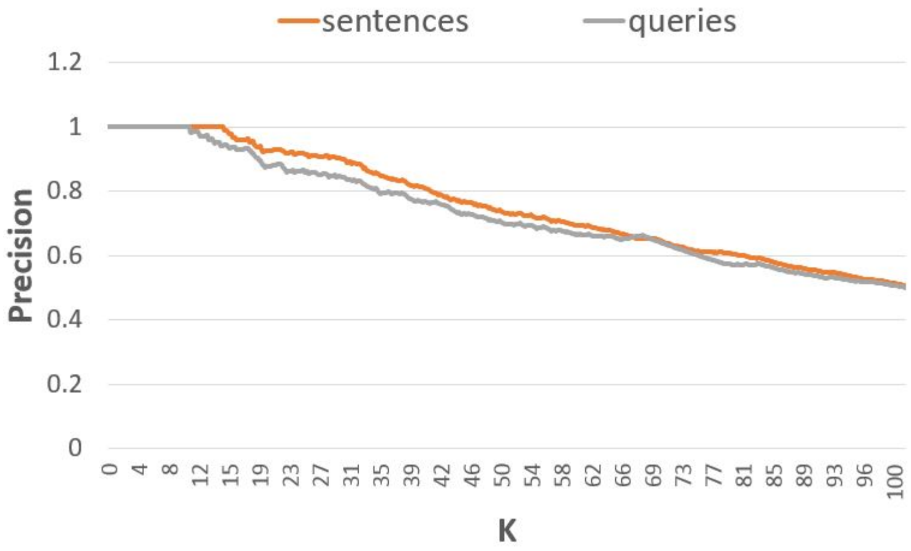
For the first task we trained the model described in Section C.3.3 using sentences from ReQ_{corp}. Note that sentences from Wikipedia were randomly sampled to make it

	Wide	**Deep**	**Wide and Deep**
Accuracy	0.68	0.92	0.94

Table C.3: Performance of diverse ranking strategies

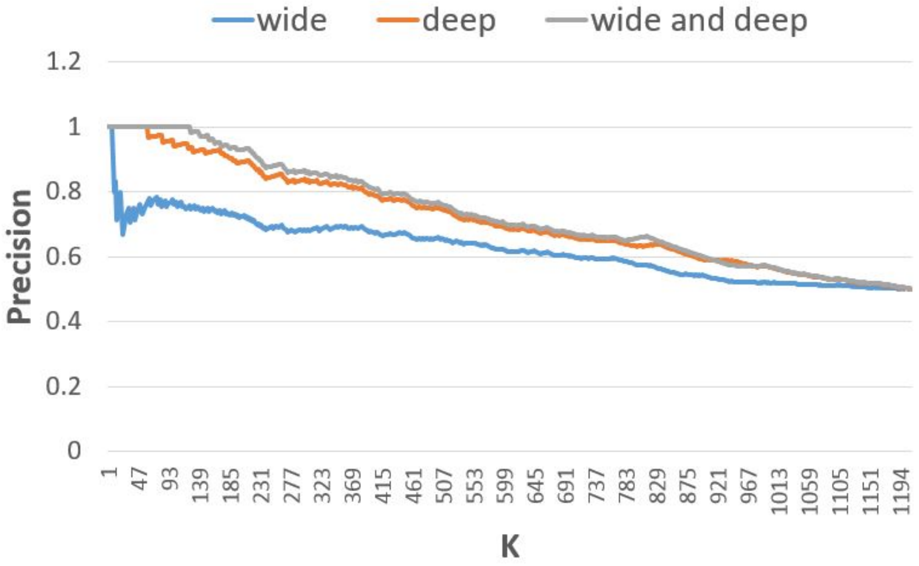
comparable to that of the children data sources, which resulted in an evenly balanced dataset. We used a 10-cross-fold-validation framework for computing prediction accuracy, which was measured and averaged for each fold. This procedure was applied using the 3 different submodels with the aim of demonstrating the validity of combining both the wide and deep models. As reported in Table C.3, the wide and deep model outperforms all its counterparts with a statistically significant improvement using a pairwise t-test with a confidence value $p < 0.05$.

Figure C.3: $Precision@K$, where K is defined as the percentage of queries and sentences analyzed. We define K as a percentage, as the raw counts of sentences and queries are not comparable otherwise



In order to demonstrate how this model can be translated to a more real query suggestion ranking environment we conducted an experiment where we ranked both sentences from $\boldsymbol{ReQ}_{corp}$ and queries from $\boldsymbol{ReQ}_{qs}$ using the model trained on sentences of $\boldsymbol{ReQ}_{corp}$. Figure C.3 illustrates the $Precision@K$, where $Precision$ is measured as the ratio of children queries/sentences among the K considered and K indicates the proportion of ReQ_{qs} and ReQ_{corp} examined. As shown in the figure, the model achieves similar results for both the sentences and the queries, proving that a model trained over sentences can be translated to the context of ranking queries.

Figure C.4: Model assessment based on $Precision@K$, where K is the number of queries examined



To further demonstrate the validity of each submodel (i.e., wide, deep, and wide and deep), we conducted an experiment using each of them. Figure C.4 illustrates the *Precision@K*, in this case measured as the ratio of children queries among the top-k analyzed, for $K \in 1..1194$. The wide and deep model outperforms both other models, followed closely by the deep model. The first adult result appears in position 123, 59, and

9, for the combined model, the deep model and the wide model respectively. Even if the suggestions of the wide model might look poor, Figure C.4 illustrates that it complements the deep model improving the overall ranking.

It is worth mentioning that upon manual examination, we noticed that most of the queries in *ReQ*$_{qs}$ labeled as “non-children" and ranked high by *ReQuIK* refer to content that could have been searched by children. For example, *“naruto shimpuden cheats"* ranked in 64^{th} position by the deep and wide model refers to a videogame that could be equally of interest for young or mature audiences, and thus it is treated by *ReQuIK* as a false positive.

C.4.4 Suitability of Retrieved Documents

We conducted another experiment to verify the suitability of the documents retrieved in response to queries suggested by *ReQuIK* and to highlight the need of readability based filtering. For conducting this experiment, we generated suggestions for the child queries in *ReQ*$_{qs}$ using popular search engines, children search engines, and *ReQuIK*. We treat as *suitable* documents that have readability levels matching the reading skills expected of children ages 6 to 13.

As children“systematically go through retrieved resources and rarely judge retrieved information sources" [106] we averaged the readability scores computed for the top-3 documents retrieved in response to each of the top-2 query suggestions generated by each evaluated strategy. For measuring readability, we use the well-known Flesch-Kincaid formula [88], which considers various textual features, such as term sentence length.

The results in Table C.4 show that documents retrieved by *ReQuIK*’s recommendations are in general easier to read and understand than the ones retrieved by other search engines,

ReQuIK	**Google**	**Bing**	**Yahoo!**	**CQS**
7.71	12.46	19.96	11.42	11.82
AskKids	**Ipl2**	**Sweet Search**	**KidRex**	**Kiddle**
13.3	10.9	12.3	12.7	12.83

Table C.4: Average readability of top-3 documents retrieved for test query recommendations

even those that explicitly target children, or CQS [219]. This is evidenced by the fact that the average readability score of documents retrieved by recommendations of popular search engines is above 11 in all cases, while for*ReQuIK* is 7.7. This correlates with the grade level of a 13 year old child, usually in 7th or 8th grade, for which the results have been filtered for, demonstrating the benefits of a readability filtering strategy as part of the query suggestion process.

We are aware that Kiddle does not offer query suggestion, however, being the premier search engine for children [136], it needed to be part of this comparison. For this reason, we analyze the readability level of the top results retrieved by Kiddle for the children queries in ReQ_{qs}. Kiddle achieves an average readability of 12.83 on retrieved document, which is comparable to the one obtained by the engine that powers it, Google. These results provide further evidence for the need for query recommendation tools for children that lead to the retrieval of resources children can read and understand.

C.4.5 Online Assessment of ReQuIK

To further validate the performance of *ReQuIK*, we conducted an online survey intended to quantify the effectiveness of *ReQuIK* from a user's perspective. For doing so, we included in the survey 10 queries randomly selected among the ones written by children in ReQ_{qs}. These queries were not included among the ones used to train our wide and deep model presented in Section C.3.3. To ensure diversity among sampled queries, we included unigrams and n-grams queries. We also considered question-type queries, which tend to be used by children. For each sampled query, we generated top-N suggestions using *ReQuIK*, a popular commercial search engine (Google), a popular children search engine (Kidzsearch), and CQS [134], the system discussed in Section C.2, which explicitly provides children query suggestions. Note that some of the sampled queries were misspelled, as they are written by children. We purposefully left these queries as they were originally written, in order to be able to consider the effect of misspelled query terms and measure its impact on the suggestions offered by each strategy.

For each sampled query, we selected the top-2 suggestions from among the list of suggestions provided by each strategy, including *ReQuIK*. These suggestions were randomly merged into a list, which was then presented to a group of independent appraisers: teachers. Following Institutional Review Board guidelines, we recruited a cohort of teachers from 5 schools in Boise, Idaho. This gave us a total of 11 teachers that participated in this study. We considered teachers as ideal appraisers for this experiment, given that they know what children are particularly interested in and are trained on the needs of children; making them capable of offering knowledgeable judgments. In the survey, we used a practical scenario, which prompted the teachers to select the best two suggestions for each query that would lead a child to locate child-friendly and interesting web-pages from their view-point. We treated their selections as the *gold standard*.

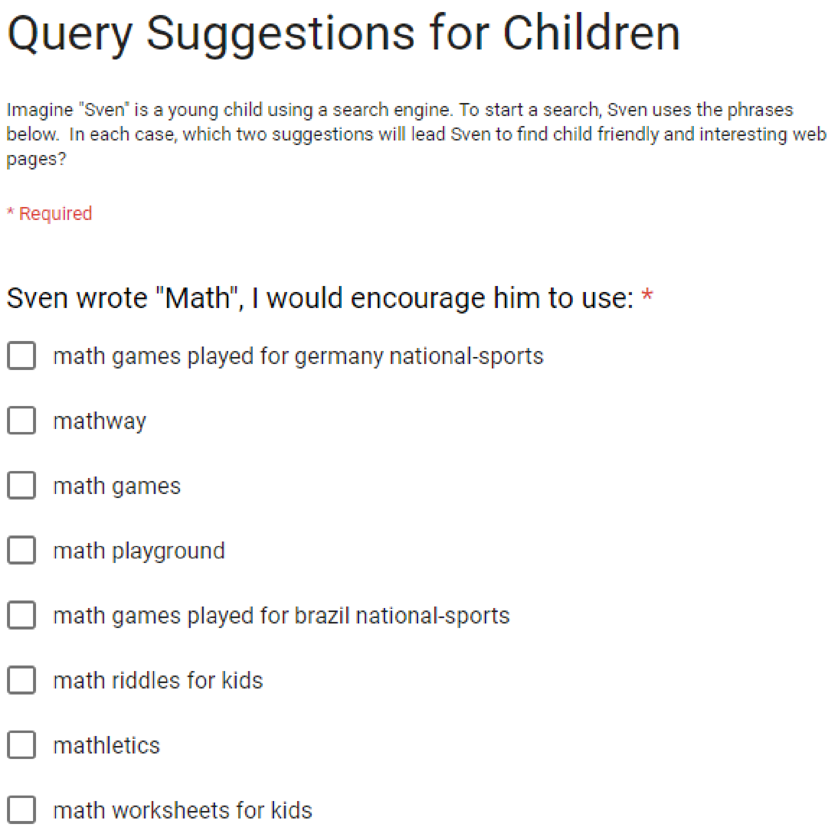
We collected 181 responses for the survey (i.e., 10 teachers selected at most two suggestions for each of the 10 sampled queries), a snapshot of which is shown in Figure C.5. For performance analysis we use accuracy of each query suggestion strategy S, computed as the fraction of suggestions selected by a teacher for a given query over the total number of suggestions generated by S. Based on the analysis of the collected responses, which we illustrate in Figure C.6, we observe that appraisers favored suggestions created by search engines and recommendation modules that target children, i.e., *ReQuIK* and Kidzsearch, rather than the ones generated by general purpose engines, i.e., Google. We should mention that except for the case of unigram queries, CQS is consistently outperformed by the remaining considered strategies. We believe this is caused by the “novelty” of the majority of the queries included in the survey, which refer to contemporary topics of interest to children (such as names and characters in recent Disney movies), when CQS suggestions are based on pre-trained probabilistic models that may not account for the probability of occurrence of such query terms.

We observed that for queries that could have been formulated by either children or more mature audiences, Google has an advantage. For example, for the query *“how do seahorses swim"* suggestions offered by Google were preferred almost exclusively over suggestions generated using any of the counterparts in this study. We hypothesize that this is due to the fact that Google’s suggestions are based on common query formulations, which in this case leads to better suggestions, as they do not necessarily have to focus on the retrieval of child-related resources. On the other hand, we observed that for other queries, bias towards children content, as in the case of Kidzsearch, CQS, and *ReQuIK*, positively affected the suggestion-selection process. For example, for the query *“Elsa"*, the top-2 suggestions generated by using Google were *“Elsa Pataky"* and *“Elsa Hosk"*, which are names of two popular celebrities. The suggestion most likely to be preferred by a child–as per teacher responses–is *"Elsa and Anna"*, which was not included among the top-2 suggestions of

Google, but was presented as the top suggestions using *ReQuIK*.

ReQuIK achieved the highest accuracy, as it consistently offered child-friendly suggestions for unigram, bigram and n-gram queries. Based on a paired t-test, the improvement in the *overall* performance of *ReQuIK* with respect to each of its counterparts is statistically significant with $p < 0.05$.

Figure C.5: Snapshot of online survey presented to teachers for overall assessment of *ReQuIK* and other strategies



Query Suggestions for Children

Imagine "Sven" is a young child using a search engine. To start a search, Sven uses the phrases below. In each case, which two suggestions will lead Sven to find child friendly and interesting web pages?

* Required

Sven wrote "Math", I would encourage him to use: *

□ math games played for germany national-sports

□ mathway

□ math games

□ math playground

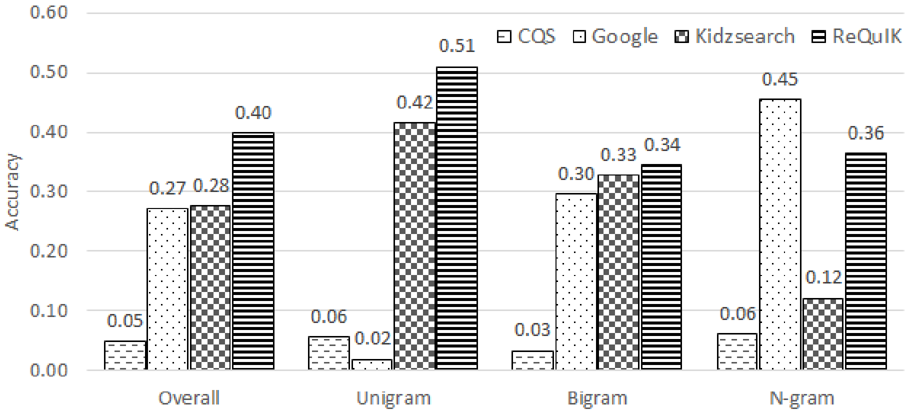
□ math games played for brazil national-sports

□ math riddles for kids

□ mathletics

□ math worksheets for kids

Figure C.6: Comparison of query recommendation systems



C.5 Scope and Limitations

We highlight below limitations we encountered in designing *ReQuIK*; which we could not address due to the project scope.

ReQuIK is a query recommendation system for children. The project is intended to explore the personalization aspect of the query recommendation task for children. Therefore, common tasks performed in traditional query recommendation systems, such as candidate generation or topic filtering do not suppose a novelty by themselves, and thus, are not extensively explored in this paper. We are also aware that both Flesch-Kincaid and Spache are simple readability formulas, which sometimes lack precision, an issue that is usually highlighted when the text is short. This constraint for short documents is common for all readability formulas and to solve it is out of scope for this project.

ReQuIK is a system that is meant to integrate and complement existing search engines, as opposed to function as a standalone tool. Consequently, tasks that are usually performed by the search engine itself, such as document retrieval and ranking, are omitted from the assessment of this project.

Due to project scope, the presented online experiment is meant to gather initial feedback on the quality of *ReQuIK*'s suggestions, as the experiment is based only on a small set of sampled queries. Given the promising results, we will conduct further online assessments.

Finally, user interaction and perceived usability of *ReQuIK*, as well as its effectiveness in terms of offering suggestions that lead to child friendly websites, are beyond the scope of this work, but will be addressed as future work (see Section C.6).

C.6 Conclusions and Future Work

In this paper, we presented *ReQuIK* a multi-perspective query recommendation system specifically tailored to facilitate information-seeking tasks for children. *ReQuIK* takes advantage of multiple strategies that inform the process of generating query suggestions and prioritize queries that are of interest to children. For assessing the performance of the system we conducted a study in collaboration with teachers of four different schools in the area of Idaho, where 97 students used our framework to complete factual and research/informational search tasks assigned to them by their teacher. We also conducted a user study to gather feedback from the teachers themselves. Using the set of queries written by children, we conducted a number of empirical analysis and demonstrated the validity of our proposed strategy and its value, in terms of offering up-to-date suggestions aimed at helping children with their information-seeking needs.

The technical contributions associated with our research work include (i) introducing a strategy which takes advantage of a search intent for capturing the purpose for which the query was written, (ii) creating a novel wide and deep neural network which considers both the raw text and traits frequently associated with child-related queries, (iii) employing a strategy to train a model based on children/non-children oriented resources that transfers well to the query suggestion context, (iv) explicitly considering the readability levels of both the query and results retrieved by the candidate suggestions, favoring the ones that leads to the retrieval of content that will be easier to read, (v) using a whole pipeline that considers multiple perspectives for candidate query generation, and (vi) creating a dataset comprised of children and non-children queries that can be used by the research community at large for children-related research.

In the future, we will extend the candidate query suggestion process by also considering phrases extracted from children corpora provided it evolves over time. In addition, we are aware that the range of 6 to 13 years can be too broad, as the interest of a 6 year old child are not the same as that of a 13 year old. Therefore, we plan to extend *ReQuIK* so that it is able to generate recommendation for more specific age ranges. Moreover, we plan to extend the manner in which resource readability is computed, to account for information beyond text, such as the different aesthetics of web pages, which can impact the level of complexity of resources. Finally, while in this paper we evaluated *ReQuIK* (and other query suggestions modules), in terms of offering queries that are suitable for children, in the future, we plan to conduct a user study to verify the degree to which *ReQuIK* leads to the retrieval of children-related resources and facilitates information discovery tasks for children.

REFERENCES

[1] Ubersuggest. `http://ubersuggest.io`, Accessed: 2017-05-18.

[2] A. Abbasi, H. Chen, and A. Salem. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *Association of Computer Machinery Transactions on Information Systems (TOIS)*, 26(3):12, 2008.

[3] A. Aizawa. An information-theoretic perspective of tf–idf measures. *Information Processing & Management*, 39(1):45–65, 2003.

[4] A. A. Al-Ajlan, H. S. Al-Khalifa, and A. Al-Salman. Towards the development of an automatic readability measurements for arabic language. In *Proceedings of the International Conference on Digital Information Management*, pages 506–511. IEEE, 2008.

[5] L. Albano, D. Beneventano, and S. Bergamaschi. Multilingual word sense induction to improve web search result clustering. In *Proceedings of the 24th International Conference on World Wide Web*, pages 835–839. ACM, 2015.

[6] J. Albright, C. de Guzman, P. Acebo, D. Paiva, M. Faulkner, and J. Swanson. Readability of patient education materials: implications for clinical practice. *Applied Nursing Research*, 9(3):139–143, 1996.

[7] S. Aluisio, L. Specia, C. Gasperin, and C. Scarton. Readability assessment for text simplification. In *North American Chapter of Association of Computational Linguistics: Human Language Technologies Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–9. Association of Computational Linguistics, 2010.

[8] B. R. Ambati, S. Reddy, and M. Steedman. Assessing relative sentence complexity using an incremental ccg parser. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1051–1057, 2016.

[9] W. Ammar, G. Mulcaire, Y. Tsvetkov, G. Lample, C. Dyer, and N. A. Smith. Massively multilingual word embeddings. *arXiv preprint arXiv:1602.01925*, 2016.

[10] A. Anagnostopoulos, L. Becchetti, C. Castillo, and A. Gionis. An optimization framework for query recommendation. In *Association of Computer Machinery International Conference on Web Search and Data Mining*, pages 161–170, 2010.

[11] D. Andor, C. Alberti, D. Weiss, A. Severyn, A. Presta, K. Ganchev, S. Petrov, and M. Collins. Globally normalized transition-based neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 2442–2452, 2016.

[12] A. Anula. Tipos de textos, complejidad lingüıstica y facilicitación lectora. In *Actas del Sexto Congreso de Hispanistas de Asia*, pages 45–61, 2007.

[13] B. Arfé, L. Mason, and I. Fajardo. Simplifying informational text structure for struggling readers. *Reading and Writing*, pages 1–20, 2017.

[14] M. Artetxe, G. Labaka, and E. Agirre. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2289–2294, 2016.

[15] M. Artetxe, G. Labaka, and E. Agirre. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462. ACL, 2017.

[16] M. Artetxe, G. Labaka, E. Agirre, and K. Cho. Unsupervised neural machine translation. *arXiv preprint arXiv:1710.11041*, 2017.

[17] M. Artetxe, G. Labaka, and E. Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, 2018.

[18] Askkids. Search engine. `http://www.askkids.com`, 2017.

[19] Backlinco. How to find long tail keywords. `http://backlinko.com/long-tail-keywords`, Accessed: 2017-05-18.

[20] R. Baeza-Yates and B. Ribeiro-Neto. *Modern information retrieval*, volume 463. ACM Press, 1999.

[21] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[22] L. Ballesteros and W. B. Croft. Phrasal translation and query expansion techniques for cross-language information retrieval. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 1997, pages 84–91. ACM, 1997. ISBN 0-89791-836-3.

[23] S. Ballesteros-Peña and I. Fernández-Aedo. Análisis de la legibilidad lingüística de los prospectos de los medicamentos mediante el índice de flesch-szigriszt y la escala inflesz. *Anales del Sistema Sanitario de Navarra*, 36(3):397–406, 2013.

[24] N. S. Baron. Language of the internet. *The Stanford Handbook for Language Engineers*, pages 59–127, 2003.

[25] M. Bea-Muñoz, M. Medina-Sánchez, and M. Flórez-García. Legibilidad de los documentos informativos en español dirigidos a lesionados medulares y accesibles por internet. *Anales del Sistema Sanitario de Navarra*, 38(2):255–262, 2015.

[26] R. G. Benjamin. Reconstructing readability: Recent developments and recommendations in the analysis of text difficulty. *Educational Psychology Review*, 24(1):63–88, 2012.

[27] G. K. Berland, M. N. Elliott, L. S. Morales, J. I. Algazy, R. L. Kravitz, M. S. Broder, D. E. Kanouse, J. A. Muñoz, J.-A. Puyol, M. Lara, et al. Health information on the internet: accessibility, quality, and readability in english and spanish. *JAMA*, 285 (20):2612–2621, 2001.

[28] E. V. Bernstam, D. M. Shelton, M. Walji, and F. Meric-Bernstam. Instruments to assess the quality of health information on the world wide web: what can our patients actually use? *International Journal Medical Information*, 74(1):13–19, 2005.

[29] D. Bilal and R. Ellis. Evaluating leading web search engines on children's queries. In *Human-Computer Interaction. Users and Applications*, pages 549–558. Springer, 2011.

[30] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[31] S. B. Bonsall, A. J. Leone, and B. P. Miller. A plain english measure of financial reporting readability. *Journal of Accounting and Economics*, 63(2-3):329–357, 2017.

[32] M. Bowden, M. Olteanu, P. Suriyentrakorn, J. Clark, and D. I. Moldovan. Lcc's power answer at qa@ clef 2006. In *Conference and Labs of the Evaluation Forum*, pages 310–317. Springer, 2006.

[33] S. Brin and L. Page. Reprint of: The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 56(18):3825–3833, 2012.

[34] P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin. A statistical approach to machine translation. *Computational linguistics*, 16(2):79–85, 1990.

[35] CCSSO. Council of chief state school officers & national governors association. supplemental information for appendix a of the common core state standards for

english language arts and literacy: New research on text complexity. Available at: http://www.corestandards.org/assets/E0813_Appendix_A_New_Research_on_Text_Complexity.pdf, 2010.

[36] J. S. Chall and E. Dale. *Readability revisited: The new Dale-Chall Readability Formula*. Brookline Books, 1995.

[37] P. Charoenpornsawat, V. Sornlertlamvanich, and T. Charoenporn. Improving translation quality of rule-based machine translation. In *Proceedings of the 2002 COLING Workshop on Machine Translation in Asia-Volume 16*, pages 1–6. ACL, 2002.

[38] G. Chen, C. Chen, Z. Xing, and B. Xu. Learning a dual-language vector space for domain-specific cross-lingual question retrieval. In *31st IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 744–755. IEEE, 2016.

[39] Y.-H. Chen, Y.-H. Tsai, and Y.-T. Chen. Chinese readability assessment using tf-idf and svm. In *Proceedings of the International Conference on Machine Learning and Computing*, volume 2, pages 705–710. IEEE, 2011.

[40] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, et al. Wide & deep learning for recommender systems. In *Deep Learning for Recommender Systems workshop*, pages 7–10, 2016.

[41] P.-Y. Chevalier and M. Brette. User readability improvement for dynamic updating of search results, June 12 2012. US Patent 8,201,107.

[42] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[43] P. Cimiano, A. Schultz, S. Sizov, P. Sorg, and S. Staab. Explicit versus latent concept models for cross-language information retrieval. In *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence*, pages 1513–1518, 2009.

[44] T. Cohn and M. Lapata. Machine translation by triangulation: Making effective use of multi-parallel corpora. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 728–735, 2007.

[45] K. Collins-Thompson and J. P. Callan. A language modeling approach to predicting reading difficulty. In *North American Chapter of Association of Computational Linguistics: Human Language Technologies*, pages 193–200, 2004.

[46] K. Collins-Thompson, P. N. Bennett, R. W. White, S. De La Chica, and D. Sontag. Personalizing web search results by reading level. In *Association of Computer Machinery International Conference on Information and Knowledge Management*, pages 403–412. ACM, 2011.

[47] B. Comrie. *Language universals and linguistic typology: Syntax and morphology*. University of Chicago press, 1989.

[48] A. Conneau, G. Lample, M. Ranzato, L. Denoyer, and H. Jégou. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*, 2017.

[49] J. Coulmance, J.-M. Marty, G. Wenzek, and A. Benhalloum. Trans-gram, fast cross-lingual word-embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1109–1113, 2015.

[50] G. Da San Martino, S. Romeo, A. Barroón-Cedeño, S. Joty, L. Maàrquez, A. Moschitti, and P. Nakov. Cross-language question re-ranking. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1145–1148. ACM, 2017.

[51] E. Dale and J. S. Chall. A formula for predicting readability: Instructions. *Educational Research Bulletin*, pages 37–54, 1948.

[52] A. Davison and R. N. Kantor. On the failure of readability formulas to define readable texts: A case study from adaptations. *Reading Research Quarterly*, pages 187–209, 1982.

[53] J. De Belder and M.-F. Moens. Text simplification for children. In *Proceedings of the SIGIR Workshop on Accessible Search Systems*, pages 19–26. ACM, 2010.

[54] O. De Clercq and V. Hoste. All mixed up? finding the optimal feature set for general readability prediction and its application to english and dutch. *Computational Linguistics*, 42(3):457–490, 2016.

[55] G. de Melo. Multilingual vector representations of words, sentences, and documents. In *Proceedings of the International Joint Conference on Natural Language Processing 2017, Tutorial Abstracts*, pages 3–5, 2017.

[56] F. Dell'Orletta, S. Montemagni, and G. Venturi. Read-it: Assessing readability of italian texts with a view to text simplification. In *Workshop on Speech and Language Processing for Assistive Technologies*, pages 73–83. ACL, 2011.

[57] J. Denning, M. S. Pera, and Y.-K. Ng. A readability level prediction tool for k-12 books. *Journal of the Association for Information Science and Technology*, 67(3): 550–565, 2016.

[58] G. Dinu, A. Lazaridou, and M. Baroni. Improving zero-shot learning by mitigating the hubness problem. *arXiv preprint arXiv:1412.6568*, 2014.

[59] Dogo. News, book, and movie reviews for kids by kids. `https://www.dogonews.com`, 2017.

[60] D. Dong, H. Wu, W. He, D. Yu, and H. Wang. Multi-task learning for multiple language 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)*, pages 1723–1732, 2015.

[61] W. Douma. De leesbaarheid van landblouwbladen een onderzoek naar en een toepassing van lees baarheidsformules. *afd. Sociologie en Sociographie van de Landbouwhogeschool te Wageningen*, 17, 1960.

[62] Y. Doval, J. Camacho-Collados, L. Espinosa Anke, and S. Schockaert. Improving cross-lingual word embeddings by meeting in the middle. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 294–304. ACL, 2018.

[63] N. Dragovic, I. Madrazo Azpiazu, and M. S. Pera. Is sven seven?: A search intent module for children. In *Association of Computer Machinery Special Interest Group on Information Retrieval*, pages 885–888. ACM, 2016.

[64] B. Drndarević, S. Štajner, S. Bott, S. Bautista, and H. Saggion. Automatic text simplification in spanish: a comparative evaluation of complementing modules. In *Conferences on Computational Linguistics and Natural Language Processing*, pages 488–500. Springer, 2013.

[65] A. Druin, E. Foss, L. Hatley, E. Golub, M. L. Guha, J. Fails, and H. Hutchinson. How children search the internet with keyword interfaces. In *Special Interest Group on Computer-Human Interaction*, pages 89–96. ACM, 2009.

[66] S. Duarte Torres, D. Hiemstra, and P. Serdyukov. An analysis of queries intended to search information for children. In *Proceedings of the third Symposium on Information Interaction in Context*, pages 235–244. ACM, 2010.

[67] S. Duarte Torres, D. Hiemstra, I. Weber, and P. Serdyukov. Query recommendation for children. In *Association of Computer Machinery International Conference on Web Search and Data Mining*, pages 2010–2014, 2012.

[68] S. Duarte Torres, D. Hiemstra, I. Weber, and P. Serdyukov. Query recommendation in the information domain of children. *Journal of the Association for Information Science and Technology*, 65(7):1368–1384, 2014.

[69] L. Duong, H. Kanayama, T. Ma, S. Bird, and T. Cohn. Learning crosslingual word embeddings without bilingual corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1285–1295, 2016.

[70] EdutainingKids. Data source for kids. `http://www.edutainingkids.com`, 2017.

[71] C. Eickhoff, P. Serdyukov, and A. P. de Vries. Web page classification on child suitability. In *Association of Computer Machinery International Conference on Web Search and Data Mining*, pages 1425–1428. ACM, 2010.

[72] C. Eickhoff, T. Polajnar, K. Gyllstrom, S. D. Torres, and R. Glassey. Web search query assistance functionality for young audiences. In *Advances in Information Retrieval*, pages 776–779. Springer, 2011.

[73] M. D. Ekstrand, M. Tian, I. M. Azpiazu, J. D. Ekstrand, O. Anuyah, D. McNeill, and M. S. Pera. Popularity and demographic biases in recommender evaluation and effectiveness. In *Conference on Fairness, Accountability and Transparency (FAT*)*, pages 172–186, 2018.

[74] M. El-Haj and P. Rayson. Osman: A novel arabic readability metric. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 250–255, 2016.

[75] A. Elfenbein. Research in text and the uses of coh-metrix. *Educational Researcher*, 40(5):246–248, 2011.

[76] H. Evans, M. G. Chao, C. M. Leone, M. Finney, and A. Fraser. Content analysis of web-based norovirus education materials targeting consumers who handle food: An assessment of alignment and readability. *Food Control*, 65:32–36, 2016.

[77] G. Eysenbach, J. Powell, O. Kuss, and E.-R. Sa. Empirical studies assessing the quality of health information for consumers on the world wide web: a systematic review. *JAMA*, 287(20):2691–2700, 2002.

[78] J. A. Fails, M. S. Pera, F. Garzotto, and M. Gelsomini. Kidrec: Children & recommender systems: Workshop co-located with association of computer machinery conference on recommender systems (conference series on recommendation systems 2017). In *Proceedings of the Eleventh Association of Computer Machinery Conference on Recommender Systems*, pages 376–377. ACM, 2017.

[79] B. Fang, Q. Ye, D. Kucukusta, and R. Law. Analysis of the perceived value of online tourism reviews: influence of readability and reviewer characteristics. *Tourism Management*, 52:498–506, 2016.

[80] L. Feng. Automatic readability assessment for people with intellectual disabilities. *Association of Computer Machinery SIG on Accessible Computing*, (93):84–91, 2009.

[81] L. Feng, M. Jansche, M. Huenerfauth, and N. Elhadad. A comparison of features for automatic readability assessment. In *Proceedings of the International Conference on Computational Linguistics: Posters*, pages 276–284. ACL, 2010.

[82] S. N. Ferdous and M. M. Ali. A semantic content based recommendation system for cross-lingual news. In *Proceedings of the Imaging, Vision & Pattern Recognition Conference (icIVPR)*, pages 1–6. IEEE, 2017.

[83] J. Fernández Huerta. Medidas sencillas de lecturabilidad. *Consigna*, 214:29–32, 1959.

[84] S. Ferrández, A. Toral, O. Ferrández, A. Ferrández, and R. Munoz. Applying wikipedia's multilingual knowledge to cross–lingual question answering. In *International Conference on Application of Natural Language to Information Systems*, pages 352–363. Springer, 2007.

[85] S. Ferrández, A. Toral, Ó. Ferrández, A. Ferrández, and R. Muñoz. Exploiting wikipedia and eurowordnet to solve cross-lingual question answering. *Information Sciences*, 179(20):3473–3488, 2009.

[86] O. Firat, K. Cho, and Y. Bengio. Multi-way, multilingual neural machine translation with a shared attention mechanism. *arXiv preprint arXiv:1601.01073*, 2016.

[87] P. Fitzsimmons, B. Michael, J. Hulley, and G. Scott. A readability assessment of online parkinson's disease information. *The Journal of the Royal College of Physicians of Edinburgh*, 40(4):292–296, 2010.

[88] R. Flesch. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221, 1948.

[89] M. L. Forcada, M. Ginestí-Rosell, J. Nordfalk, J. O'Regan, S. Ortiz-Rojas, J. A. Pérez-Ortiz, F. Sánchez-Martínez, G. Ramírez-Sánchez, and F. M. Tyers. Apertium:

a free/open-source platform for rule-based machine translation. *Machine Translation*, 25(2):127–144, 2011.

[90] P. Forner, A. Peñas, E. Agirre, I. Alegria, C. Forăscu, N. Moreau, P. Osenova, P. Prokopidis, P. Rocha, B. Sacaleanu, et al. Overview of the clef 2008 multilingual question answering track. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 262–295. Springer, 2008.

[91] J. N. Forsyth. *Automatic Readability Prediction for Modern Standard Arabic*. PhD thesis, Brigham Young University, 2014.

[92] E. Foss, A. Druin, R. Brewer, P. Lo, L. Sanchez, E. Golub, and H. Hutchinson. Children's search roles at home: Implications for designers, researchers, educators, and parents. *Journal of the Association for Information Science and Technology*, 63 (3):558–573, 2012.

[93] T. François and C. Fairon. An ai readability formula for french as a foreign language. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 466–477. ACL, 2012.

[94] M. Franz, J. S. McCarley, and S. Roukos. Ad hoc and multilingual information retrieval at ibm. In *Text Retrieval Conference*, pages 104–115, 1998.

[95] D. B. Friedman, L. Hoffman-Goetz, and J. F. Arocha. Health literacy and the world wide web: comparing the readability of leading incident cancers on the internet. *Medical Informatics and the Internet in Medicine*, 31(1):67–87, 2006.

[96] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1243–1252, 2017.

[97] G. Glavaš and S. Štajner. Simplifying lexical simplification: do we need simplified corpora? In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 63–68, 2015.

[98] F. Godin, V. Slavkovikj, W. De Neve, B. Schrauwen, and R. Van de Walle. Using topic models for twitter hashtag recommendation. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 593–596. ACM, 2013.

[99] T. Gollins and M. Sanderson. Improving cross language retrieval with triangulated translation. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 90–95. ACM, 2001.

[100] G. H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numerische Mathematik*, 14(5):403–420, 1970.

[101] I. Gonzalez-Dios, M. J. Aranzabe, A. D. de Ilarraza, and H. Salaberri. Simple or complex? assessing the readability of basque texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 334–344, 2014.

[102] T. Gossen, J. Höbel, and A. Nürnberger. A comparative study about children's and adults' perception of targeted web search engines. In *Special Interest Group on Computer-Human Interaction*, pages 1821–1824. ACM, 2014.

[103] S. Gouws and A. Søgaard. Simple task-specific bilingual word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1386–1390, 2015.

[104] S. Gouws, Y. Bengio, and G. Corrado. Bilbowa: Fast bilingual distributed representations without word alignments. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 748–756, 2015.

[105] A. C. Graesser, D. S. McNamara, and J. M. Kulikowich. Coh-metrix: Providing multilevel analyses of text characteristics. *Educational Researcher*, 40(5):223–234, 2011.

[106] L. Graham and P. T. Metaxas. Of course it's true; i saw it on the internet!: critical thinking in the internet era. *Communications of the Association of Computer Machinery*, 46(5):70–75, 2003.

[107] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.

[108] S. Grossberg. Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks*, 1(1):17–61, 1988.

[109] R. Gunning. *The Technique of Clear Writing*. McGraw-Hill, New York, 1952.

[110] J. Guo, W. Che, D. Yarowsky, H. Wang, and T. Liu. Cross-lingual dependency parsing based on distributed representations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1234–1244, 2015.

[111] J. Gwizdka and D. Bilal. Analysis of children's queries and click behavior on ranked results and their thought processes in google search. In *Proceedings of the 2017 Conference on Human Information Interaction and Retrieval (CHIIR)*, pages 377–380, 2017.

[112] T.-L. Ha, J. Niehues, and A. Waibel. Toward multilingual neural machine translation with universal encoder and decoder. *arXiv preprint arXiv:1611.04798*, 2016.

[113] A. Halavais. *Search engine society*. John Wiley & Sons, 2013.

[114] S. Hale and S. Campbell. The interaction between text difficulty and translation accuracy. *Babel*, 48(1):14–33, 2002.

[115] K. M. Hermann and P. Blunsom. Multilingual distributed representations without word alignment. In *Proceedings of International Conference on Learning Representations*, 2014.

[116] K. M. Hermann and P. Blunsom. Multilingual models for compositional distributional semantics. In *Proceedings of Association of Computational Linguistics*, 2014.

[117] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom. Teaching machines to read and comprehend. In *Proceedings of Advances in Neural Information Processing Systems*, pages 1693–1701, 2015.

[118] W. R. Hersh and D. Hickam. Information retrieval in medicine: the saphire experience. *Journal of the American Society for Information Science*, 46(10): 743–747, 1995.

[119] S. Hochreiter. Untersuchungen zu dynamischen neuronalen netzen. *Diploma, Technische Universität München*, 91:1, 1991.

[120] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

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

[122] K. Huang, M. Gardner, E. Papalexakis, C. Faloutsos, N. Sidiropoulos, T. Mitchell, P. P. Talukdar, and X. Fu. Translation invariant word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1084–1088, 2015.

[123] T. Huibers, N. Kucirkova, E. Murgia, J. A. Fails, M. Landoni, and M. S. Pera. 3rd kidrec workshop: What does good look like? In *Proceedings of the 18th ACM International Conference on Interaction Design and Children*, pages 681–688. ACM, 2019.

[124] A. M. Ibrahim, C. R. Vargas, P. G. Koolen, D. J. Chuang, S. J. Lin, and B. T. Lee. Readability of online patient resources for melanoma. *Melanoma Research*, 26(1): 58–65, 2016.

[125] Ikasbil. Resources. `http://www.ikasbil.eus`, 2018.

[126] Ilp2. Search engine. `http://www.Ilp2.com`, 2017.

[127] M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. Viégas, M. Wattenberg, G. Corrado, et al. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017.

[128] S. Joty, P. Nakov, L. Màrquez, and I. Jaradat. Cross-language learning with adversarial neural networks: Application to community question answering. *arXiv preprint arXiv:1706.06749*, 2017.

[129] A. Joulin, P. Bojanowski, T. Mikolov, H. Jégou, and E. Grave. Loss in translation: Learning bilingual word mapping with a retrieval criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2984, 2018.

[130] D. Kamholz, J. Pool, and S. Colowick. Panlex: Building a resource for panlingual lexical translation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 3145–3150, 2014.

[131] L. Kandel and A. Moles. Application de l'indice de flesch à la langue française. *Cahiers Etudes de Radio-Télévision*, 19:253–274, 1958.

[132] P. Kanerva. *Sparse Distributed Memory*. MIT press, 1988.

[133] T. Kanungo and D. Orr. Predicting the readability of short web summaries. In *Proceedings of the Second Association of Computer Machinery International Conference on Web Search and Data Mining*, pages 202–211. ACM, 2009.

[134] S. Karimi and M. S. Pera. Recommendations to enhance children web searches. *ACM Conference on Recommender Systems: Posters*, 2015.

[135] N. Karpov, J. Baranova, and F. Vitugin. Single-sentence readability prediction in russian. In *Proceedings of the International Conference on Analysis of Images, Social Networks and Texts*, pages 91–100. Springer, 2014.

[136] L. Keating. Kiddle search engine is the google for kids. `http://www.techtimes.com/articles/136639/20160225/kiddle-search-engine-kids.htm`, Accessed: 2017-05-18.

[137] Kidrex. Safe search for kids. `http://www.KidRex.org`, 2017.

[138] KidsInMind. Data source for kids. `https://www.kids-in-mind.com`, 2017.

[139] KidzSearch. Web search engine for kids. `http://www.kidzsearch.com`, 2017.

[140] Kidzvuz. Data source for kids. `https://www.kidzvuz.com`, 2017.

[141] J. P. Kincaid, R. P. Fishburne Jr, R. L. Rogers, and B. S. Chissom. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, DTIC Document, 1975.

[142] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[143] G. R. Klare. A second look at the validity of readability formulas. *Journal of Literacy Research*, 8(2):129–152, 1976.

[144] T. Kočiskỳ, K. M. Hermann, and P. Blunsom. Learning bilingual word representations by marginalizing alignments. *arXiv preprint arXiv:1405.0947*, 2014.

[145] P. Koehn. *Statistical Machine Translation*. Cambridge University Press, 2009.

[146] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, et al. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume. Proceedings of the Demo and Poster Sessions*, pages 177–180, 2007.

[147] K. Komiya, S. Shibata, and Y. Kotani. Cross-lingual product recommendation using collaborative filtering with translation pairs. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 141–152. Springer, 2014.

[148] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[149] S. Lai, L. Xu, K. Liu, and J. Zhao. Recurrent convolutional neural networks for text classification. In *Association for the Advancement of Artificial Intelligence*, pages 2267–2273, 2015.

[150] Y. Lai and J. Zeng. A cross-language personalized recommendation model in digital libraries. *The Electronic Library*, 31(3):264–277, 2013.

[151] T. K. Landauer, P. W. Foltz, and D. Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25(2-3):259–284, 1998.

[152] S. Lauly, A. Boulanger, and H. Larochelle. Learning multilingual word representations using a bag-of-words autoencoder. *arXiv preprint arXiv:1401.1803*, 2014.

[153] A. Lazaridou, G. Dinu, and M. Baroni. Hubness and pollution: Delving into cross-space mapping for zero-shot learning. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 270–280, 2015.

[154] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, pages 556–562, 2001.

[155] C. Lennon and H. Burdick. The lexile framework as an approach for reading measurement and success. *Available at http://cdn.lexile.com/cms_page_media/135/The%20Lexile%20Framework%20 for%20Reading.pdf*, 2004.

[156] D. J. Levinson. A conception of adult development. *American psychologist*, 41(1):3, 1986.

[157] M. P. Lewis and F. Gary. Simons, and charles d. fennig (eds.). 2013. *Ethnologue: Languages of the world*, pages 233–62, 2015.

[158] Lexile. How is the lexile measure of a text determined? Available at: https://lexile.desk.com/customer/en/portal/articles/508829-how-is-the-lexile-measure\-of-a-text-determined-, 2016.

[159] Lexile. Who are our publisher partners. lexile. https://www.lexile.com/about-lexile/how-to-get-lexile-measures/text-measure/publishers/who-else-is-doing-it/, 2017.

[160] Y. Li, D. McLean, Z. A. Bandar, J. D. O'shea, and K. Crockett. Sentence similarity based on semantic nets and corpus statistics. *IEEE transactions on knowledge and data engineering (TKDE)*, 18(8):1138–1150, 2006.

[161] Y. Lin, J.-B. Michel, E. L. Aiden, J. Orwant, W. Brockman, and S. Petrov. Syntactic annotations for the google books ngram corpus. In *Proceedings of the ACL 2012 System Demonstrations*, pages 169–174. ACL, 2012.

[162] M. L. Littman, S. T. Dumais, and T. K. Landauer. Automatic cross-language information retrieval using latent semantic indexing. In *Cross-language Information Retrieval*, pages 51–62. Springer, 1998.

[163] A. Lopez. Statistical machine translation. *ACM Computing Surveys (CSUR)*, 40(3): 8, 2008.

[164] P. Lops, C. Musto, F. Narducci, M. De Gemmis, P. Basile, and G. Semeraro. Cross-language personalization through a semantic content-based recommender system. In *International Conference on Artificial Intelligence: Methodology, Systems, Applications*, pages 52–60. Springer, 2010.

[165] P. Lucisano and M. E. Piemontese. Gulpease: una formula per la predizione della difficoltà dei testi in lingua italiana. *Scuola e città*, 3(31):110–124, 1988.

[166] M.-T. Luong, H. Pham, and C. D. Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[167] T. Luong, H. Pham, and C. D. Manning. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159, 2015.

[168] I. Madrazo Azpiazu. Towards multilingual readability assessment. Master's thesis, Boise State University, Boise, Idaho, USA, 2017.

[169] I. Madrazo Azpiazu and M. S. Pera. Is readability a valuable signal for hashtag recommendations? *ACM Conference on Recommender Systems: Posters*, 2016.

[170] I. Madrazo Azpiazu and M. S. Pera. Multiattentive recurrent neural network architecture for multilingual readability assessment. *Transactions of the Association for Computational Linguistics*, 7:421–436, 2019.

[171] I. Madrazo Azpiazu and M. S. Pera. Hierarhical compositional mapping for cross-lingual embedding generation. In *In-press*, 2019.

[172] I. Madrazo Azpiazu and M. S. Pera. Is cross-lingual readability assessment possible? *Journal of the Association for Information Science and Technology*, To appear.

[173] I. Madrazo Azpiazu, N. Dragovic, and M. S. Pera. Finding, understanding and learning: Making information discovery tasks useful for children and teachers. In *Association of Computer Machinery Special Interest Group on Information Retrieval: Workshop on Search as Learning*, 2016.

[174] I. Madrazo Azpiazu, N. Dragovic, O. Anuyah, and M. S. Pera. Looking for the movie seven or sven from the movie frozen?: A multi-perspective strategy for recommending queries for children. In *Proceedings of the 2018 Conference on Human Information Interaction & Retrieval (ACM CHIIR)*, pages 92–101. ACM, 2018.

[175] B. Magnini and C. Strapparava. Improving user modelling with content-based techniques. In *Proceedings of the International Conference on User Modeling*, pages 74–83. Springer, 2001.

[176] B. Magnini, M. Negri, R. Prevete, and H. Tanev. Multilingual question/answering: the diogene system. In *Text Retrieval Conference*, 2001.

[177] B. S. Manjunath and W.-Y. Ma. Texture features for browsing and retrieval of image data. *Institute of Electrical and Electronics Engineers Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.

[178] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association of Computational Linguistics System Demonstrations*, pages 55–60, 2014.

[179] M. Maritxalar Anglada and I. Madrazo Azpiazu. Testuen irakurgarritasuna neurtzeko sailkatzaile automatikoa [an automatic classifier of text legibility]. Master's thesis, University of the Basque Country (UPV/EHU), 2014.

[180] G. H. Mc Laughlin. Smog grading-a new readability formula. *Journal of Reading*, 12(8):639–646, 1969.

[181] J. S. McCarley. Should we translate the documents or the queries in cross-language information retrieval? In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 208–214. ACL, 1999.

[182] R. Mehrotra, A. Anderson, F. Diaz, A. Sharma, H. Wallach, and E. Yilmaz. Auditing search engines for differential satisfaction across demographics. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 626–633. International World Wide Web Conferences Steering Committee, 2017.

[183] M. Mesgar and M. Strube. Graph-based coherence modeling for assessing readability. In *Proceedings of the 4th Joint Conference on Lexical and Computational Semantics*, pages 309–318, 2015.

[184] M. Mesgar and M. Strube. Lexical coherence graph modeling using word embeddings. In *Proceedings of the 2016 Conference of the North American Chapter of the*

Association for Computational Linguistics: Human Language Technologies, pages 1414–1423, 2016.

[185] M. Mesgar and M. Strube. A neural local coherence model for text quality assessment. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4328–4339, 2018.

[186] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell. *Machine Learning: An Artificial Intelligence Approach*. Springer Science & Business Media, 2013.

[187] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[188] T. Mikolov, Q. V. Le, and I. Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*, 2013.

[189] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

[190] C. Musto, F. Narducci, P. Basile, P. Lops, M. De Gemmis, and G. Semeraro. Cross-language information filtering: Word sense disambiguation vs. distributional models. In *International Conference of the Italian Association for Artificial Intelligence*, pages 250–261. Springer, 2011.

[191] F. Narducci, M. Palmonari, and G. Semeraro. Cross-language semantic matching for discovering links to e-gov services in the lod cloud. *Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data*, 992:21–32, 2013.

[192] F. Narducci, M. Palmonari, and G. Semeraro. Cross-language semantic retrieval and linking of e-gov services. In *International Semantic Web Conference*, pages 130–145. Springer, 2013.

[193] F. Narducci, P. Basile, C. Musto, P. Lops, A. Caputo, M. de Gemmis, L. Iaquinta, and G. Semeraro. Concept-based item representations for a cross-lingual content-based recommendation process. *Information Sciences*, 374:15–31, 2016.

[194] A. Nenkova and K. McKeown. A survey of text summarization techniques. In *Mining Text Data*, pages 43–76. Springer, 2012.

[195] J.-Y. Nie, M. Simard, P. Isabelle, and R. Durand. Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the web. In *Proceedings of the 22nd Annual International Conference on Research and Development in Information Retrieval*, pages 74–81. ACM, 1999.

[196] D. W. Oard and P. Hackett. Document translation for cross-language text retrieval at the university of maryland. In *Text Retrieval Conference*, pages 687–696, 1997.

[197] G. H. Paetzold and L. Specia. Unsupervised lexical simplification for non-native speakers. In *Association for the Advancement of Artificial Intelligence*, pages 3761–3767, 2016.

[198] J. Palotti, G. Zuccon, and A. Hanbury. The influence of pre-processing on the estimation of readability of web documents. In *Association of Computer Machinery Conference on Information and Knowledge Management*, pages 1763–1766, 2015.

[199] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. ACL, 2002.

[200] C. R. Patel, S. Sanghvi, D. V. Cherla, S. Baredes, and J. A. Eloy. Readability assessment of internet-based patient education materials related to parathyroid surgery. *Annals of Otology, Rhinology & Laryngology*, pages 523–527, 2015.

[201] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[202] M. S. Pera and Y.-K. Ng. Automating readers' advisory to make book recommendations for k-12 readers. In *Proceedings of the ACM Conference on Recommender Systems*, pages 9–16. ACM, 2014.

[203] J. Petkovic, J. Epstein, R. Buchbinder, V. Welch, T. Rader, A. Lyddiatt, R. Clerehan, R. Christensen, A. Boonen, N. Goel, et al. Toward ensuring health equity: Readability and cultural equivalence of omeract patient-reported outcome measures. *The Journal of Rheumatology*, 42(12):2448–2459, 2015.

[204] X. H. Pham, J. J. Jung, N. T. Nguyen, and P. Kim. Ontology-based multilingual search in recommendation systems. *Acta Polytechnica Hungarica*, 13(2):195–207, 2016.

[205] M. Potthast, B. Stein, and M. Anderka. A wikipedia-based multilingual retrieval model. *Advances in Information Retrieval*, pages 522–530, 2008.

[206] F. Radlinski and N. Craswell. A theoretical framework for conversational search. In *Proceedings of the 2017 Conference on Human Information Interaction and Retrieval*, pages 117–126. ACM, 2017.

[207] R. Rahimi and A. Shakery. Online learning to rank for cross-language information retrieval. In *Proceedings of the 40th International Association of Computer Machinery Special Interest Group IR Conference on Research and Development in Information Retrieval*, pages 1033–1036. ACM, 2017.

[208] RaisingChildren. Data source for kids. `http://raisingchildren.net.au`, 2017.

[209] Renaissance. Text complexity, atos, and lexileÂő measures. Available at: `http://www.renaissance.com/products/practice/accelerated-reader-360/atos-and-text-complexity/`, 2017.

[210] F. Ricci, L. Rokach, and B. Shapira. Introduction to recommender systems handbook. In *Recommender Systems Handbook*, pages 1–35. Springer, 2011.

[211] S. Ruder. A survey of cross-lingual embedding models. *arXiv preprint arXiv:1706.04902*, 2017.

[212] S. Ruder, I. Vulić, and A. Søgaard. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:569–631, 2019.

[213] G. Salton. Automatic processing of foreign language documents. *Journal of the Association for Information Science and Technology*, 21(3):187–194, 1970.

[214] J. San Pedro and A. Karatzoglou. Question recommendation for collaborative question answering systems with rankslda. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 193–200. ACM, 2014.

[215] P. Sarath, S. Lauly, H. Larochelle, M. Khapra, B. Ravindran, V. C. Raykar, and A. Saha. An autoencoder approach to learning bilingual word representations. In *Advances in Neural Information Processing Systems*, pages 1853–1861, 2014.

[216] A. M. Saxe, J. L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

[217] S. Schmidt, P. Scholl, C. Rensing, and R. Steinmetz. Cross-lingual recommendations in a resource-based learning scenario. In *European Conference on Technology Enhanced Learning*, pages 356–369. Springer, 2011.

[218] S. E. Schwarm and M. Ostendorf. Reading level assessment using support vector machines and statistical language models. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 523–530. ACL, 2005.

[219] M. Shaikh, M. S. Pera, and Y.-K. Ng. Suggesting simple and comprehensive queries to elementary-grade children. In *Institute of Electrical and Electronics Engineers/Web Intelligence Conference/Association of Computer Machinery WI-IAT*, volume 1, pages 252–259, 2015.

[220] S. E. Shaywitz, M. D. Escobar, B. A. Shaywitz, J. M. Fletcher, and R. Makuch. Evidence that dyslexia may represent the lower tail of a normal distribution of reading ability. *New England Journal of Medicine*, 326(3):145–150, 1992.

[221] T. Shi, Z. Liu, Y. Liu, and M. Sun. Learning cross-lingual word embeddings via matrix co-factorization. In *Association of Computational Linguistics*, pages 567–572, 2015.

[222] Y. Shigeto, I. Suzuki, K. Hara, M. Shimbo, and Y. Matsumoto. Ridge regression, hubness, and zero-shot learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 135–151. Springer, 2015.

[223] S. L. Smith, D. H. Turban, S. Hamblin, and N. Y. Hammerla. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859*, 2017.

[224] A. Søgaard, Ž. Agić, H. M. Alonso, B. Plank, B. Bohnet, and A. Johannsen. Inverted indexing for cross-lingual nlp. In *The 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2015)*, 2015.

[225] P. Sorg and P. Cimiano. Exploiting wikipedia for cross-lingual and multilingual information retrieval. *Data & Knowledge Engineering*, 74:26–45, 2012.

[226] G. Spache. A new readability formula for primary-grade reading materials. *The Elementary School Journal*, 53(7):410–413, 1953.

[227] Spaghetti. Book reviews by kids for kids. http://www.spaghettibookclub.org, 2017.

[228] S. Spaulding. A spanish readability formula. *The Modern Language Journal*, 40(8): 433–441, 1956.

[229] S. Štajner and H. Saggion. Readability indices for automatic evaluation of text simplification systems: A feasibility study for spanish. In *International Joint Conference on Natural Language Processing 2013*, pages 374–382, 2013.

[230] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112, 2014.

[231] Sweetsearch. A search engine for students. http://www.sweetsearch.com, 2017.

[232] A. Takasu. Cross-lingual keyword recommendation using latent topics. In *Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, pages 52–56. ACM, 2010.

[233] K. Tanaka-Ishii, S. Tezuka, and H. Terada. Sorting texts by readability. *Computational Linguistics*, 36(2):203–227, 2010.

[234] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1555–1565, 2014.

[235] X. Tang, X. Wan, and X. Zhang. Cross-language context-aware citation recommendation in scientific articles. In *Proceedings of the 37th international ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 817–826. ACM, 2014.

[236] J. Tiedemann. Parallel data, tools and interfaces in opus. In N. C. C. Chair), K. Choukri, T. Declerck, M. U. Dogan, B. Maegaard, J. Mariani, J. Odijk, and S. Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. European Language Resources Association (ELRA), 2012.

[237] K. Topping, J. Samuels, and T. Paul. Independent reading: the relationship of challenge, non-fiction and gender to achievement. *British Educational Research Journal*, 34(4):505–524, 2008.

[238] Toyinsider. Data source for kids. `http://www.thetoyinsider.com`, 2017.

[239] J. Turian, L. Ratinov, and Y. Bengio. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394. ACL, 2010.

[240] K. Uchiyama, H. Nanba, A. Aizawa, and T. Sagara. Osusume: cross-lingual recommender system for research papers. In *Proceedings of the 2011 Workshop on Context-awareness in Retrieval and Recommendation*, pages 39–42. ACM, 2011.

[241] A. Uitdenbogerd. Readability of french as a foreign language and its uses. In *ADCS 2005: The Tenth Australasian Document Computing Symposium*, pages 19–25. University of Sydney, 2005.

[242] S. Upadhyay, M. Faruqui, C. Dyer, and D. Roth. Cross-lingual models of word embeddings: An empirical comparison. *arXiv preprint arXiv:1604.00425*, 2016.

[243] A. Usta, I. S. Altingovde, I. B. Vidinli, R. Ozcan, and Ö. Ulusoy. How k-12 students search for learning?: analysis of an educational search engine log. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1151–1154. ACM, 2014.

[244] A. Vallin, B. Magnini, D. Giampiccolo, L. Aunimo, C. Ayache, P. Osenova, A. Peñas, M. De Rijke, B. Sacaleanu, D. Santos, et al. Overview of the clef 2005 multilingual question answering track. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 307–331. Springer, 2005.

[245] N. Vanderschantz, A. Hinze, and S. J. Cunningham. “sometimes the internet reads the question wrong": Children’s search strategies & difficulties. *Journal of the Association for Information Science and Technology*, 51(1):1–10, 2014.

[246] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[247] I. B. Vidinli and R. Ozcan. New query suggestion framework and algorithms: A case study for an educational search engine. *Information Processing & Management*, 52(5):733–752, 2016.

[248] I. Vulić and M.-F. Moens. Bilingual distributed word representations from document-aligned comparable data. *Journal of Artificial Intelligence Research*, 55:953–994, 2016.

[249] I. Vulić, W. De Smet, and M.-F. Moens. Cross-language information retrieval models based on latent topic models trained with document-aligned comparable corpora. *Information Retrieval*, 16(3):331–368, 2013.

[250] H. X. Wang. Developing and testing readability measurements for second language learners. 2016.

[251] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang. Dimensional sentiment analysis using a regional cnn-lstm model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 225–230, 2016.

[252] W. Weaver. Translation. *Machine translation of languages*, 14:15–23, 1955.

[253] Webscope. L16 yahoo! answers dataset. yahoo! answers, 2016. URL `https://webscope.sandbox.yahoo.com/catalog.php?` [Online; accessed 17-June-2017].

[254] E. W. Whittaker, J. R. Novak, P. Chatain, P. R. Dixon, M. H. Heie, and S. Furui. Clef2006 question answering experiments at tokyo institute of technology. In

Workshop of the Cross-Language Evaluation Forum for European Languages, pages 351–361. Springer, 2006.

[255] Wikimedia. Simplification guidelines for simple wikipedia. `https://simple.wikipedia.org/wiki/Wikipedia:Simple_English_Wikipedia`, 2018.

[256] Wizenoze. Resources. `http://www.wizenoze.com`, 2018.

[257] A. Wood and Y.-K. Ng. Orthogonal query recommendations for children. In *Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services*, pages 298–302. ACM, 2016.

[258] M. Xiao and Y. Guo. Distributed word representation learning for cross-lingual dependency parsing. In *The Special Interest Group NLL Conference on Computational Natural Language Learning*, pages 119–129, 2014.

[259] C. Xing, D. Wang, C. Liu, and Y. Lin. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, 2015.

[260] B. Xu, N. Wang, T. Chen, and M. Li. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*, 2015.

[261] J. Xu, X. He, and H. Li. Deep learning for matching in search and recommendation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1365–1368. ACM, 2018.

[262] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the International Conference on Machine Learning*, pages 2048–2057, 2015.

[263] Yahoo! Academic Relations. L13 - yahoo! search query tiny sample. `http://webscope.sandbox.yahoo.com/catalog.php`.

[264] C.-Z. Yang, I.-X. Chen, and P.-J. Wu. Cross-lingual news group recommendation using cluster-based cross-training. *Computational linguistic and Chinese Language Processing*, 13(1):41–60, 2008.

[265] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, 2016.

[266] C.-H. Yu and R. C. Miller. Enhancing web page readability for non-native readers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2523–2532. ACM, 2010.

[267] H. Yu and V. Hatzivassiloglou. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 129–136. ACL, 2003.

[268] H. Zamani, M. Bendersky, X. Wang, and M. Zhang. Situational context for ranking in personal search. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1531–1540. International World Wide Web Conferences Steering Committee, 2017.

[269] E. Zangerle, W. Gassler, and G. Specht. Recommending#-tags in twitter. In *Proceedings of the Workshop on Semantic Adaptive Social Web (SASWeb 2011). CEUR Workshop Proceedings*, volume 730, pages 67–78, 2011.

[270] J. Zhang, X. Hu, Y. Zhang, and H. Liu. Your age is no secret: Inferring microbloggers' ages via content and interaction analysis. In *Tenth International AAAI Conference on Web and Social Media*, 2016.

[271] Y. Zhang, R. D. Brown, and R. E. Frederking. Adapting an example-based translation system to chinese. In *Proceedings of the first International Conference on Human Language Technology Research*, pages 1–4. ACL, 2001.

[272] Z. Zhang and O. Nasraoui. Mining search engine query logs for query recommendation. In *Proceedings of the 15th international conference on World Wide Web*, pages 1039–1040. ACM, 2006.

[273] W. Y. Zou, R. Socher, D. Cer, and C. D. Manning. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398, 2013.