Chapter 1

Introduction to Semantic Textual Similarity

Semantic Textual Similarity (STS) measures the equivalence of meanings between two textual segments. In Natural Language Processing (NLP), measuring semantic similarity between two textual segments plays an important role. It is one of the fundamental tasks for many NLP applications and their related areas. To measure STS, the input is always two textual segments, and the output is a continuous value that represents the degree of similarity of the two input textual segments (Agirre et al. 2012). These segments can be a short snippet of texts, complete sentences or even documents (Agirre et al. 2013).

STS is related to both textual entailment (TE) and paraphrasing but differs in several ways. TE is the task of identifying the relationship between two texts commonly addressed as text (t) and hypothesis (h). Entailment, contradiction, and neutral are the most popular relationship types in TE (Dagan et al. 2006; Marelli et al. 2014). On the other hand, paraphrasing identification is the task of recognising text fragments with approximately the same meaning within a specific context (Vrbanec and Meštrović 2020). Therefore, TE and paraphrasing give a categorical output while STS identifies the degree of equivalence of two texts as a continuous value.

Measuring STS is an important research problem, having many applications in NLP such as information retrieval (IR) (Varelas et al. 2005; Subhashini and Kumar 2010), text summarisation (Aliguliyev 2009; Schallehn et al. 2004), question answering (Mohler et al. 2011), relevance feedback (Wang et al. 2020a), text classification (Li and Han 2013; Albitar et al. 2014) and word sense disambiguation (Abdalgader and Skabar 2011). In the field of databases, text similarity can be used for schema matching. In the document databases like Elasticsearch[1], there is a core module called *"Similarity module"* that defines the document matching process. Furthermore, STS is also useful for relational join operations in databases where join attributes are textually similar to each other (Cohen 2000; Schallehn et al. 2004). Also, STS is widely used in semantic web applications like community extraction (Zhang et al. 2010), Twitter search (Feng et al. 2013) where it is required the ability to measure semantic relatedness between concepts or entities accurately.

These applications require to measure STS automatically, which means that computer programs should be developed to calculate STS between two textual inputs. The most natural way to approach this problem is to use a machine learning approach where the computer learns from examples. The development of rule-based methods would be too cumbersome, and it is unlikely to lead to robust solutions. Over the years, researchers have proposed numerous ML solutions for STS. These ML solutions can be categorised into two main

[1] Elasticsearch is a document database based on the Lucene library. It is available on `https://www.elastic.co/`. More information on the Similarity module is available on `https://www.elastic.co/guide/en/elasticsearch/reference/current/index-modules-similarity.html`

categories; (a) Linguistic feature-based (b) Vector/ Embedding-based . Most of the early approaches belong to the linguistic feature-based category. With this, features for the ML algorithm were hand-crafted. Such features include edge-distances between nodes in WordNet (Miller 1995), number of named entities in two input texts, corpus pattern analysis features etc. Then these features would be fed into an ML algorithm such as Support Vector Machine (SVM), Linear Regression etc. (Béchara et al. 2015). This ML algorithm will be trained on an annotated STS dataset and then can be used to measure STS automatically. Despite being extremely popular before the neural network era, linguistic feature-based algorithms have limitations. Determining the best linguistic features for calculating STS is not an easy task as it requires a good understanding of the linguistic phenomenon and relies on researchers' intuition. In addition, most of these features depend on lexical knowledge bases like WordNet, which makes it difficult to adopt them in languages other than English. Furthermore, these features primarily rely on parsers that are not available in other languages (Ranjan et al. 2016). However, these methods' most significant limitation would be that they no longer provide strong results compared to the vector-based methods (Cer et al. 2017).

With the introduction of word embeddings (Mikolov et al. 2013a), ML solutions in NLP shifted from feature-based methods to vector-based methods. Pre-trained word embedding models such as word2vec (Mikolov et al. 2013a), GloVe (Pennington et al. 2014b), fastText (Mikolov et al. 2018) etc. provide a learned representation for texts where the words with the same meaning have a similar representation. Since these word embeddings are already semantically powerful, ML solutions no longer require to depend on lexical knowledge bases. As a result, embedding based ML solutions are easy to adopt in different languages as long as the pre-trained embeddings are available in that language. Furthermore, these solutions are now state-of-the-art in NLP tasks, including STS, providing stronger results than feature-based ML solutions (Cer et al. 2017). Therefore, as STS solutions in this part of the thesis, we mainly explore embedding based ML approaches.

Similar to general ML algorithms, vector-based ML STS algorithms can too be classified into two main categories; Supervised and Unsupervised. In supervised learning, ML models will be trained using labelled data. Therefore, supervised ML algorithms require data that the humans already annotated with the closest answer. On the other hand, for unsupervised learning, you do not need an annotated dataset. Unsupervised ML approaches would discover the features by themselves. Given that annotated STS data is not commonly available in many languages and domains, exploring both supervised and unsupervised STS methods is essential. Therefore, the first two chapters in this part of the thesis explore unsupervised STS methods, while the last two chapters explore supervised STS methods.

The most common unsupervised STS approaches are vector aggregation methods like Word Vector Averaging, Word Mover's Distance (Kusner et al. 2015) and Smooth Inverse Frequency (Arora et al. 2017). In Chapter 2, we explore them in detail. We identify the best vector aggregation method empirically by analysing them in different STS datasets. Finally, we propose a new state-of-the-art vector aggregation method based on contextual word embeddings that outperforms other methods.

In Chapter 3, we explore another unsupervised STS method using sentence encoders. Sentence encoders are different from vector aggregation methods as they have end-to-end models to get sentence embeddings rather than a simple aggregation method. They provide strong results compared to other unsupervised STS methods. We use three different sentence encoders and analyse their performance in various aspects of English STS and also evaluate their portability to different languages and domains.

In Chapters 4 and 5, we explore most popular supervised STS approaches. Usually, in supervised vector-based STS approaches, word embeddings would be fed into a neural network like tree-structured neural networks (Tai et al. 2015) and Siamese neural networks (Mueller and Thyagarajan 2016). Among them, Siamese neural networks have been widely used in STS and have additional advantages compared to other structures. Therefore, we discuss them comprehensively in Chapter 4. We evaluate the existing Siamese Neural Network architectures in STS datasets and propose a novel Siamese Neural Network architecture for smaller STS datasets that outperforms current state-of-the-art Siamese neural models. We also assess its performance in different languages and domains.

In the final chapter of Part I of this thesis, we explore the newly released transformers in STS tasks. Transformers have taken the NLP field by storm, providing very successful results in various NLP tasks. In Chapter 5, we bring together various transformer architectures (Devlin et al. 2019; Yang et al. 2019b; Liu et al. 2019) and investigate their performance in various STS datasets. We explore the strengths and weaknesses of transformer models regarding accuracy and efficiency and discover the possible solutions for their limitations.

The main contributions of this part of the thesis are as follows.

1. Each chapter covers various supervised and unsupervised techniques to compute semantic textual similarity that benefits a wide range of NLP applications. We empirically evaluate all of them in three English datasets, two non-English datasets and an out of domain dataset to explore their adaptability.

2. We propose a novel unsupervised STS method based on contextual word embeddings that outperforms current state-of-the-art unsupervised vector aggregation STS methods in all the English datasets, non-English datasets, and datasets in other domains.

3. We propose a novel Siamese neural network architecture that is efficient and outperforms current state-of-the-art Siamese neural network architectures in smaller STS datasets.

4. We provide important resources to the community. The code of each chapter as an open-source GitHub repository and the pre-trained STS models will be freely available to the community. The link to the GitHub

repository and the models will be unveiled in the introduction section of each chapter.

The remainder of this chapter is structured as follows. Section 1.1 discusses the various datasets we used in *"Semantic Textual Similarity"* part of the thesis and briefly analyse the datasets for common properties. In Section 1.2, we discuss the main evaluation metrics we used in the *"Semantic Textual Similarity"* part of the thesis. The chapter finishes with the conclusions.

1.1 Datasets

The popularity of STS is partially owed to the large number of shared tasks organised in SemEval from 2012-2017 (Agirre et al. 2012; Agirre et al. 2013; Agirre et al. 2014; Agirre et al. 2015; Agirre et al. 2016; Cer et al. 2017). First, they have provided annotated datasets that can train STS ML models and evaluate them. Second, at the end of each shared task, the solutions submitted by the participants are published, and the best solutions can be considered as state-of-the-art STS methods.

To maintain the versatility of our methods, we experimented with several English STS datasets as well as several non-English datasets and a dataset from a different domain which we will discuss later in this section. These datasets carry different and interesting characteristics. Therefore, with the introduction, we also do an exploratory analysis of the dataset focussing on various properties. All of the datasets which are described here are publicly available and can be considered as STS benchmarks.

1.1.1 English Datasets

1. **SICK dataset**[2] - The SICK data contains 9927 sentence pairs with a 5,000/4,927 training/test split which were employed in the *SemEval 2014 Task1: Evaluation of Compositional Distributional Semantic Models on Full Sentences through Semantic Relatedness and Textual Entailment* (Marelli et al. 2014). The dataset has two types of annotations: Semantic Relatedness and Textual Entailment. We only use Semantic Relatedness annotations in our research. SICK was built based on two existing datasets: the 8K ImageFlickr dataset (Rashtchian et al. 2010)[3] and the SemEval-2012 STS MSR-Video Descriptions dataset (Agirre et al. 2012)[4]. The 8K ImageFlickr dataset is a dataset of images, where each image is associated with five descriptions. To derive SICK sentence pairs, the organisers randomly selected 750 images and sampled two descriptions from each of them. The SemEval2012 STS MSR-Video Descriptions data set is a collection of sentence pairs sampled from the short video snippets which compose the Microsoft Research Video Description Corpus[5]. A subset of 750 sentence pairs has been randomly chosen from this data set to be used in SICK.

   To generate SICK data from the 1,500 sentence pairs taken from the

[2] The SICK dataset is available to download at https://wiki.cimec.unitn.it/tiki-index.php?page=CLIC

[3] The 8K ImageFlickr data set is available at http://hockenmaier.cs.illinois.edu/8k-pictures.html

[4] The SemEval-2012 STS MSR-Video Descriptions dataset is available at https://www.cs.york.ac.uk/semeval-2012/task6/index.html

[5] The Microsoft Research Video Description Corpus is available to download at https://research.microsoft.com/en-us/downloads/38cf15fd-b8df-477e-a4e4-a4680caa75af/

source data sets, a 3-step process has been applied to each sentence pair, namely *(i) normalisation, (ii) expansion and (iii) pairing* (Marelli et al. 2014). The *normalisation* step has been carried out on the original sentences to exclude or simplify instances that contained lexical, syntactic or semantic phenomena such as named entities, dates, numbers, multiword expressions etc. In the *expansion* step, syntactic and lexical transformations with predictable effects have been applied to each normalised sentence, to obtain *(i)* a sentence with a similar meaning, *(ii)* a sentence with a logically contradictory or at least highly contrasting meaning, and *(iii)* a sentence that contains most of the same lexical items, but has a different meaning. Finally, in the *pairing* step, each normalised sentence in the pair has been combined with all the sentences resulting from the expansion phase and with the other normalised sentence in the pair. Furthermore, several pairs composed of completely unrelated sentences have been added to the data set by randomly taking two sentences from two different pairs (Marelli et al. 2014).

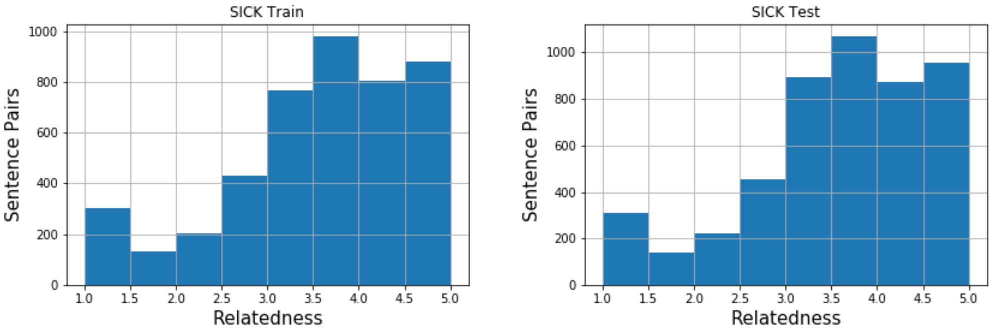
Each pair in the SICK dataset has been annotated to mark the degree to which the two sentence meanings are related (on a 5-point scale). The ratings have been collected through a large crowdsourcing study, where ten different annotators have evaluated each pair. Once all the annotations were collected, the relatedness gold score has been computed for each pair as the average of the ten ratings assigned by the annotators (Marelli et al. 2014). Table 1.1 shows examples of sentence pairs with different degrees of semantic relatedness; gold relatedness scores are expressed on a 5-point rating scale. Given a test sentence pair, the machine learning models require to predict a value between 0-5, which reflects the relatedness of the given sentence pair.

Sentence Pair	**Relatedness**
1. A little girl is looking at a woman in costume. 2. A young girl is looking at a woman in costume.	4.7
1. Nobody is pouring ingredients into a pot. 2. Someone is pouring ingredients into a pot.	3.5
1. Someone is pouring ingredients into a pot. 2. A man is removing vegetables from a pot.	2.8
1. A man is jumping into an empty pool. 2. There is no biker jumping in the air.	1.6

Table 1.1: Example sentence pairs from the SICK dataset with their gold relatedness scores (on a 5-point rating scale). **Sentence Pair** column shows the two sentence and **Relatedness** column denotes the annotated relatedness score.

Figure 1.1 shows the distribution of the relatedness value in the SICK training and SICK testing set. It is clear that there are more sentence pairs with high relatedness values compared to low relatedness values. SICK train and SICK test follow a similar distribution.

The SICK dataset consists of pairs of sentences. We will refer to the first sentence in the pair as *sentence 1* and to the second sentence as *sentence 2*. In Figure 1.2 we visualise the normalised distribution of word count for both *sentence 1* and *sentence 2* in the SICK train and SICK test. Both sentences have a similar distribution reaching a maximum of around nine words. SICK train and SICK test follow a similar pattern in word count



(a) Relatedness distribution of SICK training test

(b) Relatedness distribution of SICK testing set

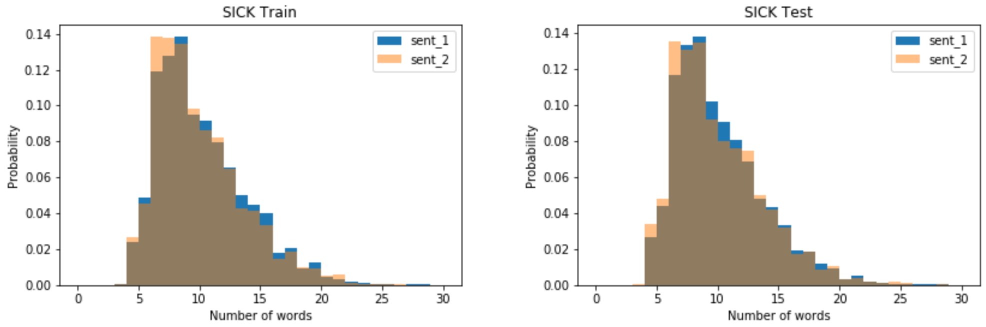
Figure 1.1: Relatedness distribution of SICK train and SICK test. *Sentence Pairs* shows the number of sentence pairs that a certain *Relatedness bin* has.

Measure	SICK Train		SICK Test	
	Sent_1	**Sent_2**	**Sent_1**	**Sent_2**
Word Count Mean	9.73	9.52	9.69	9.53
Word Count STD	3.66	3.70	3.69	3.65
Word Count MAX	28	32	28	30
Word Count MIN	3	3	3	3

Table 1.2: Word count stats in SICK training and SICK testing. *STD* indicates the standard deviation and the other acronyms indicate the common meaning

distribution too. Additionally we show some word count statistics in Table 1.2. In SICK train number of words for a sentence ranges from 3 to 32 and have the mean number of words around 9.5. These statistics are extremely close in the SICK test too.

The common judgement in STS is that when two sentences share a large number of words, the relatedness of that two sentences should be higher. In fact, in early feature-based approaches of calculating semantic textual similarity, the number of overlapping words between the two sentences was a common feature (Vilariño et al. 2014; Gupta et al. 2014a; Lynum et



(a) Normalised distribution of word count in SICK train

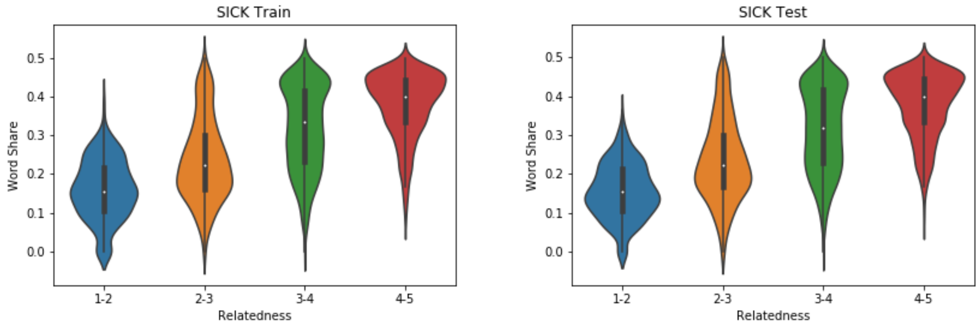
(b) Normalised distribution of word count in SICK test

Figure 1.2: Normalised distribution of word count in SICK train and SICK test. *Number of words* indicates the word count and *Probability* shows the total probability of a sentence with that word count appearing in the dataset.

al. 2014; Chávez et al. 2014). Systems like Vilariño et al. (2014) and Lynum et al. (2014) use the number of words common in two sentences as a feature directly, while systems like Gupta et al. (2014a) and Chávez et al. (2014) use Jaccard Similarity Coefficient as a feature, which is a measurement based on word overlap. To observe whether the number of words common in the two sentences has a relationship on the relatedness, we draw a violin plot[6] for each relatedness score bins with word share in Figure 1.3.

In figure 1.3, it is clear that sentence pairs with a higher relatedness tend to have a high word share. However, it should be noted that, in the "2-3" relatedness score bin, there are some sentence pairs with a high word share. The most common example for such a case would be *sentence 2* is the complete negation of the *sentence 1* (Marelli et al. 2014). In such cases,

[6] Violin plots are similar to box plots, except that they also show the probability density of the data at different values, usually smoothed by a kernel density estimator.



(a) Word share against relatedness bins in SICK train

(b) Word share against relatedness bins in SICK test

Figure 1.3: Word share against relatedness bins in SICK train and SICK test. *Word Share* indicates the ratio between number of common words in the two sentences to total number of words in the two sentences against each *Relatedness* bins

the two sentences share a large portion of the words, and one sentence has the *"not"* word that gives a completely opposite meaning compared to the other sentence. Similarly "4-5" relatedness score bin has some sentence pairs with a low word share. Those sentence pairs do not contain the same words but have synonyms or even paraphrases and possess the same overall meaning (Marelli et al. 2014). Therefore, the STS methods that focus on word share as a feature will not perform well in the SICK dataset (Ranasinghe et al. 2019a).

A clear strength in the SICK dataset is that the training set and the testing set reflect similar properties with regards to sentence length, relatedness distribution etc. Therefore, a properly trained machine learning model on the SICK train should give good results to the SICK test set as well (Marelli et al. 2014).

2. **STS 2017 English Dataset**[7] The second English STS dataset we used to experiment in this thesis is STS 2017 English Dataset, which was employed in *SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Cross-lingual Focused Evaluation* which is the most recent STS task in SemEval (Cer et al. 2017). As the training data for the competition, participants were encouraged to make use of all existing data sets from prior STS evaluations, including all previously released trial, training and evaluation data from SemEval 2012 - 2016 (Agirre et al. 2012; Agirre et al. 2013; Agirre et al. 2014; Agirre et al. 2015; Agirre et al. 2016). Once combined, we had 8277 sentence pairs for training. More information about the datasets used to build the training set is available in Table 1.3.

   On the other hand, a fresh test set of 250 sentence pairs was provided by SemEval-2017 STS Task organisers (Cer et al. 2017). The Stanford Natural Language Inference (SNLI) corpus (Bowman et al. 2015) was the primary data source for this test set. Similar to the SICK dataset, each pair in the STS 2017 English Test set has been annotated to mark the degree to which the two sentence meanings are related (on a 5-point scale). The ratings have been collected through crowdsourcing on Amazon Mechanical Turk[8]. Five annotations have been collected per pair, and the gold score has been computed for each pair as the average of the five ratings assigned by the

[7] The STS 2017 English Dataset is available to download at http://ixa2.si.ehu.es/stswiki/

[8] Amazon Mechanical Turk is a crowdsourcing website for businesses to hire remotely located *crowd workers* to perform discrete on-demand tasks. It is available at https://www.mturk.com/

Year	Dataset	Pairs	Source
2012 (Agirre et al. 2012)	MSRpar	1500	newswire
	MSRvid	1500	videos
	OnWN	750	glosses
	SMTnews	750	WMT eval.
	SMTeuroparl	750	WMT eval.
2013 (Agirre et al. 2013)	HDL	750	newswire
	FNWN	189	glosses
	OnWN	561	glosses
	SMT	750	MT eval.
2014 (Agirre et al. 2014)	HDL	750	newswire headlines
	OnWN	750	glosses
	Deft-forum	450	forum posts
	Deft-news	300	news summary
	Images	750	image descriptions
	Tweet-news	750	tweet-news pairs
2015 (Agirre et al. 2015)	HDL	750	newswire headlines
	Images	750	image descriptions
	Ans.-student	750	student answers
	Ans.-forum	375	Q&A forum answers
	Belief	375	committed belief
2016 (Agirre et al. 2016)	HDL	249	newswire headlines
	Plagiarism	230	short-answer plag.
	post-editing	244	MT postedits
	Ans.-Ans.	254	Q&A forum answers
	Quest.-Quest.	209	Q&A forum questions
2017 (Cer et al. 2017)	Trial	23	Mixed STS 2016

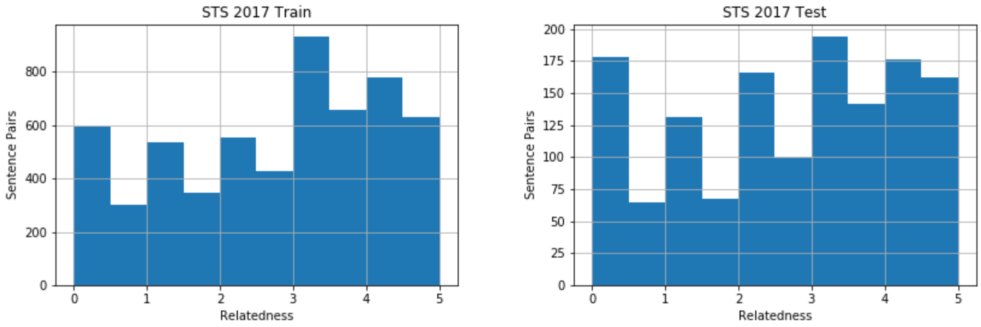
Table 1.3: Information about the datasets used to build the English STS 2017 training set. The **Year** column shows the year of the SemEval competition that the dataset got released. **Dataset** column expresses the acronym used describe a dataset in that year. **Pairs** is the number of sentence pairs in that particular dataset and **Source** shows the source of the sentence pairs.

annotators. However, unlike the SICK dataset, the organisers have a clear explanation for the score ranges. Table 1.4 shows some example sentence pairs from the dataset with the gold labels and their explanations. Similar to the SICK dataset, the machine learning models require predicting a value between 0-5 which reflects the similarity of the given sentence pair (Cer et al. 2017).

Sentence Pair	**Relatedness**
The two sentences are completely equivalent as they mean the same thing. 1. The bird is bathing in the sink. 2. Birdie is washing itself in the water basin.	5
The two sentences are completely equivalent as they mean the same thing. 1. The bird is bathing in the sink. 2. Birdie is washing itself in the water basin.	4
The two sentences are roughly equivalent, but some important information differs/missing. 1. John said he is considered a witness but not a suspect. 2. “He is not a suspect anymore.” John said.	3
The two sentences are not equivalent, but share some details. 1. They flew out of the nest in groups. 2. They flew into the nest together.	2
The two sentences are not equivalent, but are on the same topic. 1. The woman is playing the violin. 2. The young lady enjoys listening to the guitar.	1
The two sentences are completely dissimilar 1. The black dog is running through the snow. 2. A race car driver is driving his car through the mud.	0

Table 1.4: Example sentence pairs from the STS2017 English dataset with their gold relatedness scores (on a 5-point rating scale) and explanations. **Sentence Pair** column shows the two sentence and **Relatedness** column denotes the annotated relatedness score.

Similar to the SICK dataset, we conduct an exploratory data analysis on



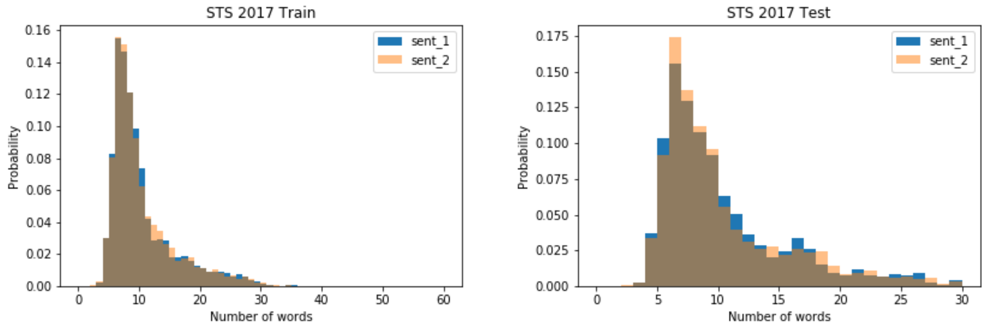
(a) Relatedness distribution of STS 2017 training test

(b) Relatedness distribution of STS 2017 testing set

Figure 1.4: Relatedness distribution of STS 2017 train and STS 2017 test. *Sentence Pairs* shows the number of sentence pairs that a certain *Relatedness bin* has.

STS2017 dataset too. Figure 1.4 shows the relatedness distribution, and Figure 1.5 shows the normalised distribution of word count for *sentence 1* and *sentence 2* in STS 2017 train and test sets. Most of these statistics are similar to the SICK dataset. One notable change is the maximum word count in STS 2017 training dataset, which is 57 in *sentence 1* and 48 in *sentence 2* according to Table 1.5 while both SICK datasets' and STS 2017 test set's maximum word count is limited to 30. We believe that the reason is STS training dataset is composed with many sources including news articles that can have lengthy sentences. However, the STS algorithm should be able to properly handle this imbalance nature between the STS2017 train and test set (Cer et al. 2017).

In Figure 1.6, we draw a violin plot for each relatedness score bin with word share. We can see that generally, higher word share leads to higher relatedness, but still, there can be sentence pairs that contradict this, which



(a) Normalised distribution of word count in STS 2017 train

(b) Normalised distribution of word count in STS 2017 test

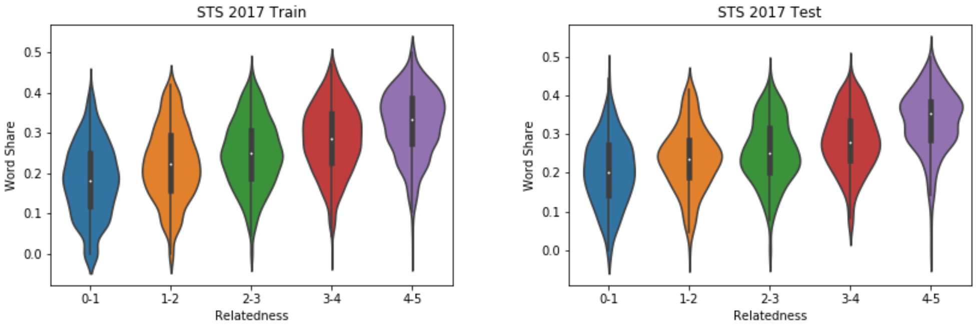
Figure 1.5: Normalised distribution of word count in STS 2017 train and STS 2017 test. *Number of words* indicates the word count and *Probability* shows the total probability of a sentence with that word count appearing in the dataset.

Measure	STS 2017 Train		STS 2017 Test	
	Sent_1	**Sent_2**	**Sent_1**	**Sent_2**
Word Count Mean	10.01	9.94	9.83	9.80
Word Count STD	5.52	5.36	5.14	5.14
Word Count MAX	57	48	30	30
Word Count MIN	3	2	3	2

Table 1.5: Word count stats in STS 2017 training and STS 2017 testing. *STD* indicates the standard deviation and the other acronyms indicate the common meaning

is similar to the observation we had with the SICK dataset.

Since the statistics of SICK and STS 2017 datasets are similar, one dataset can be used to augment the training data in the other dataset or perform transfer learning, which can lead to better results as neural networks perform stronger with more data (Wang et al. 2020c; Li et al. 2021). We hope to experiment this with supervised machine learning models in Chapters 4 and 5.



(a) Word share against relatedness bins in STS 2017 train

(b) Word share against relatedness bins in STS 2017 test

Figure 1.6: Word share against relatedness bins in STS 2017 train and STS 2017 test. *Word Share* indicates the ratio between number of common words in the two sentences to total number of words in the two sentences against each *Relatedness* bins

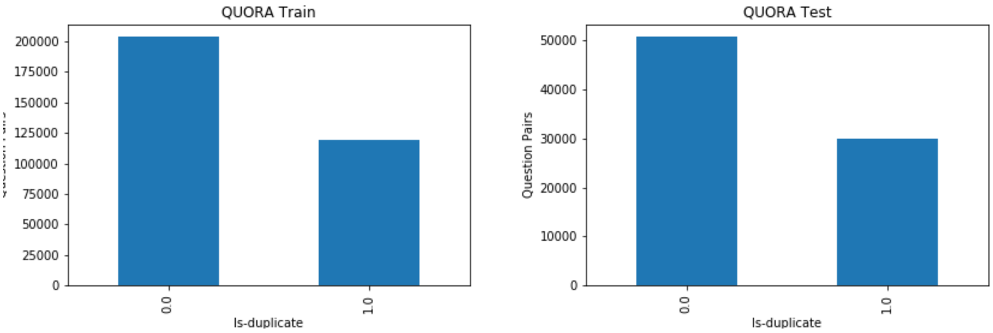
Question Pair	**is-duplicate**
1. What are natural numbers? 2. What is a least natural number?	0
1. Which Pizzas are most popularly ordered in Dominos menu? 2. How many calories does a Dominos Pizza have?	0
1. How do you start a bakery? 2. How can one start a bakery business?	1
1. Should I learn Python or Java first? 2. If I had to choose between learning Java and Python what should I choose to learn first?	1

Table 1.6: Example question pairs from the Quora Question Pairs dataset with their gold is-duplicate value. **Question Pair** column shows the two questions and **is-duplicated** column denotes whether it is a duplicated pair or not.

3. **Quora Question Pairs**[9] The Quora Question Pairs dataset is a big dataset that was first released for a Kaggle Competition[10]. Quora is a question-

[9] The Quora Question Pairs Dataset is available to download at http://qim.fs.quoracdn.net/quora_duplicate_questions.tsv

[10] Kaggle is an online community of data scientists and machine learning practitioners that hosts machine learning competitions. The Quora Question Pairs competition is available on https://www.kaggle.com/c/quora-question-pairs



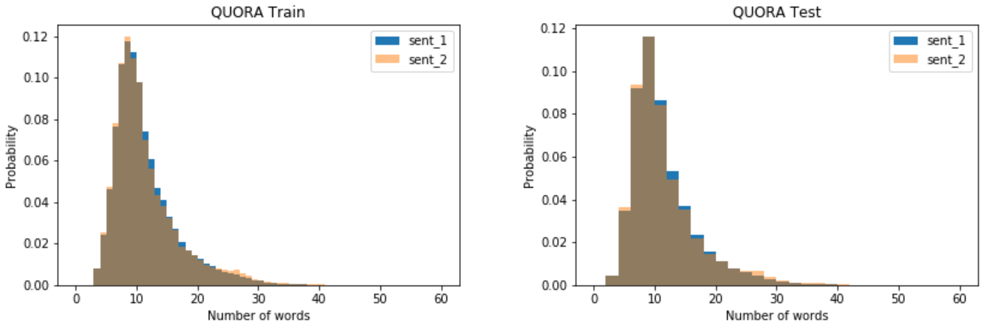
(a) Is-duplicate distribution of QUORA training test

(b) Is-duplicate distribution of QUORA testing set

Figure 1.7: Is-duplicate distribution of QUORA train and QUORA test. *Sentence Pairs* shows the number of sentence pairs that a certain *Is-duplicate* has.

and-answer website where internet users ask, answer, follow, and edit questions, either factually or in the form of opinions. If a particularly new question has been asked before, users merge the new question to the original question flagging it as a duplicate. The organisers used this functionality to create the dataset and did not use a separate annotation process. Their original sampling method has returned an imbalanced dataset with many more true examples of duplicate pairs than non-duplicates. Therefore, the organisers have supplemented the dataset with negative examples. One source of negative examples has been pairs of *related question* which belongs to similar topics but are not truly semantically equivalent.

The dataset has 400,000 question pairs, and we used 4:1 split on that to separate it into a training set and a testing set, resulting in 320,000 questions pairs in the training set and 80,000 sentence pairs in the testing



(a) Normalised distribution of word count in QUORA train

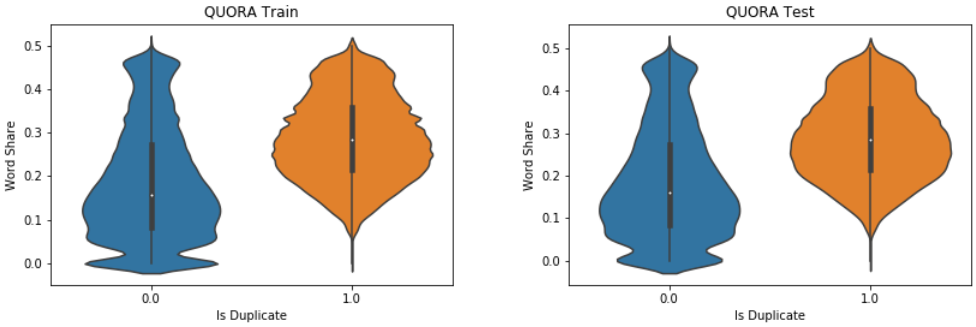
(b) Normalised distribution of word count in QUORA test

Figure 1.8: Normalised distribution of word count in QUORA train and QUORA test. *Number of words* indicates the word count and *Probability* shows the total probability of a sentence with that word count appearing in the dataset.

set. The machine learning models need to predict a continuous value between 0 and 1 that reflects whether it is a duplicate question pair or not. 1 indicates that a particular question pair is a duplicate, and 0 means it is not a duplicate.

This dataset is different from the previous datasets since it is not artificially created and uses day-to-day language. Since it has more than 300,000 training instances, deep learning systems will benefit more when used on this dataset.

In Figure 1.7 we show the distribution of the two classes in the QUORA dataset. The dataset seems to have more non-duplicate question pairs than duplicate sentence pairs, similar to the real-world scenario. According to the word count distribution in Figure 1.8 and word count statistics in Table 1.7, it is clear that QUORA datasets contain longer texts than SICK and STS 2017 datasets. Therefore, the QUORA dataset should be able to test



(a) Word share against is-relatedness value in QUORA train

(b) Word share against is-relatedness value in QUORA test

Figure 1.9: Word share against Is-duplicate values in QUORA train and QUORA test. *Word Share* indicates the ratio between number of common words in the two sentences to total number of words in the two sentences against each *Is-duplicate*

machine learning models' ability to handle lengthy texts properly.

In Figure 1.9 we show a violin plot for each *"is-duplicate"* value with word share. We can see that duplicate questions have a high word share. However, it should be noted that there are non-duplicate question pairs that still have a high word share. This shows that determining STS is not a trivial task.

According to statistics provided by the Director of Product Management at Quora on 17 September 2018, over 100 million people visit Quora every month, which raises the problem of different users asking similar questions with the same intent but in different words (Imtiaz et al. 2020). Multiple questions with the same intent can cause seekers to spend more time finding the best answer to their question and make writers feel they need to answer multiple versions of the same question. Therefore, identifying duplicate questions will make finding high-quality answers to questions

Measure	QUORA Train		QUORA Test	
	Ques_1	**Ques_2**	**Ques_1**	**Ques_2**
Word Count Mean	10.95	11.20	10.92	11.14
Word Count STD	5.44	6.31	5.40	6.31
Word Count MAX	125	237	73	237
Word Count MIN	1	1	1	1

Table 1.7: Word count stats in QUORA training and QUORA testing.*STD* indicates the standard deviation and the other acronyms indicate the common meaning

easier, resulting in an improved experience for Quora writers, seekers, and readers.

1.1.2 Datasets on Other Languages

One of the main requirements in our research was to build an STS method without depending on the language. Therefore throughout our study, we worked on several datasets from different languages. Those non-English datasets are described below.

1. **Arabic STS Dataset** [11] The Arabic STS dataset we selected was also used for the Arabic STS subtask in *SemEval 2017 Task 1: Semantic Textual Similarity Multilingual and Cross-lingual Focused Evaluation* (Cer et al. 2017). Unlike English, no data from previous SemEval competitions were available since this was the first time an Arabic STS task was organised in SemEval. More information about the extracted sentences will be shown in Table 1.9.

   A subset of the English STS 2017 dataset has been selected and human

[11] The Arabic STS dataset can be downloaded at http://alt.qcri.org/semeval2017/task1/index.php?id=data-and-tools

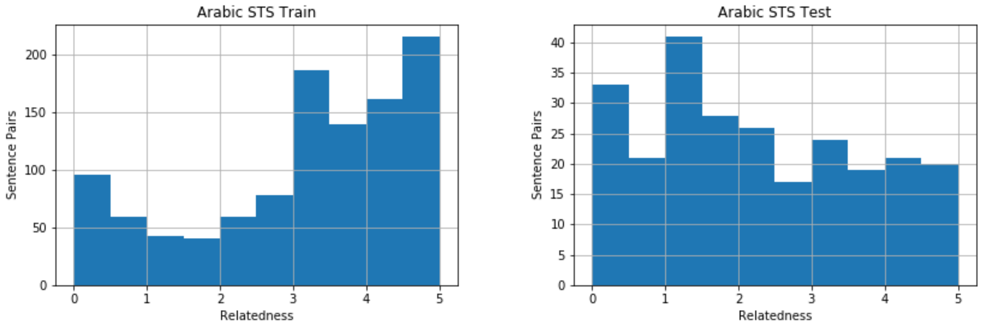
Sentence Pair	Similarity
1. أحدهم يقلي لحما. *Someone is frying meat.* 2. أحدهم يعزف البيانو. *Someone plays the piano.*	0.250
1. أمرأة تظيف المكونات في الإناء. *A woman cleaning ingredients in the bowl.* 2. إمرأة تكسر ثلاثة بيضات في الإناء. *A woman breaks three eggs in a bowl.*	1.750
1. طفلة تعزف القيثارة. *A Child is playing harp.* 2. رجل يعزف القيثارة . *A man plays the harp.*	2.250
1. المرأة تقطع البصل الأخضر. *The woman chops green onions.* 2. إمرأة تقشر بصلة. *A woman peeling an onion.*	3.250
1. الأيل قفز فوق السياج. *The deer jumped over the fence.* 2. أيل يقفز فوق سياج الإعصار. *Deer Jumps Over Hurricane Fence*	4.800

Table 1.8: Example question pairs from the Arabic STS dataset. **Sentence Pair** column shows the two sentences. We also included their translations in the table. The translations were done by a native Arabic speaker. **Similarity** column indicates the annotated similarity of the two sentences.

translated into Arabic to prepare the annotated instances. Sentences have been translated independently from their pairs. Arabic translations have been provided by native Arabic speakers with strong English skills at Carnegie Mellon University in Qatar. Translators have been given an English sentence and its Arabic machine translations where they have

Dataset	Pairs	Source
Trial	23	Mixed STS 2016
MSRpar	510	newswire
MSRvid	368	videos
SMTeuroparl	203	WMT eval.

Table 1.9: Information about the datasets used to build the Arabic STS training set. **Dataset** column expresses the acronym used describe the dataset. **Pairs** is the number of sentence pairs in that particular dataset and **Source** shows the source of the sentence pairs.



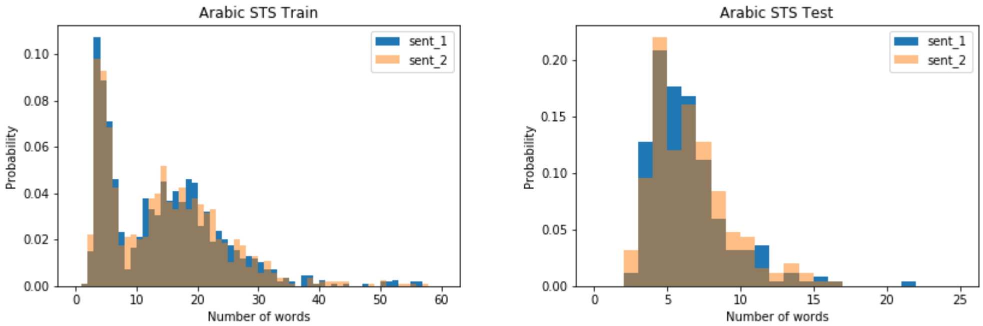
(a) Relatedness distribution of Arabic STS training test

(b) Relatedness distribution of Arabic STS testing set

Figure 1.10: Relatedness distribution of Arabic STS train and Arabic STS test. *Sentence Pairs* shows the number of sentence pairs that a certain *Relatedness bin* has.

performed post-editing to correct errors. STS labels have been then transferred to the translated pairs. Therefore, annotation guidelines and the template are similar to the English STS 2017 dataset. 1103 sentence pairs were available for training, and 250 sentence pairs were available in the test set. Table 1.8 shows few pairs of sentences with their similarity scores. The machine learning models require predicting a value between 0-5, which reflects the similarity of a given Arabic sentence pair.

Similar to the English STS datasets, we also analysed the Arabic STS dataset



(a) Normalised distribution of word count in Arabic STS train

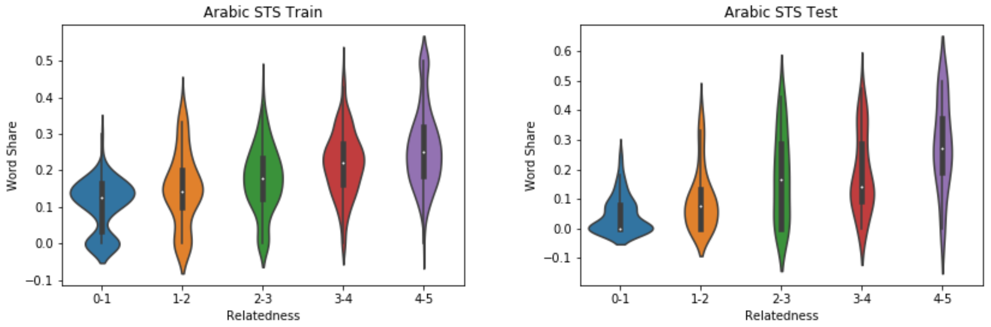
(b) Normalised distribution of word count in Arabic STS test

Figure 1.11: Normalised distribution of word count in Arabic STS train and Arabic STS test. *Number of words* indicates the word count and *Probability* shows the total probability of a sentence with that word count appearing in the dataset.

considering the same set of properties. As can be seen in Figure 1.10, the relatedness distribution is different in the training and test sets. In the training set, there are many sentences with high relatedness scores compared to low relatedness scores. On the other hand, there are many sentences with low relatedness scores compared to the high relatedness scores in the test set.

Word count distribution in the training and test sets of the Arabic dataset is different too. As shown in Figure 1.11 the sentences in the training set are longer than the sentences in the test set. This is further confirmed by the stats in Table 1.10. The average word count in the training set is 31, while this is 9 in the test set. With these observations, we can conclude that the Arabic training set and test are different with regard to several properties. This nature of the dataset can be a challenge for ML systems.

In Figure 1.12, we draw a violin plot for each relatedness bin with



(a) Word share against relatedness bins in Arabic STS train

(b) Word share against relatedness bins in Arabic STS test

Figure 1.12: Word share against relatedness bins in Arabic STS train and Spanish STS test. *Word Share* indicates the ratio between number of common words in the two sentences to total number of words in the two sentences against each *Relatedness* bins

Measure	Arabic STS Train		Arabic STS Test	
	Sent_1	**Sent_2**	**Sent_1**	**Sent_2**
Word Count Mean	31.23	31.02	9.03	9.34
Word Count STD	12.15	12.37	3.66	3.74
Word Count MAX	90	90	22	24
Word Count MIN	5	1	3	3

Table 1.10: Word count stats in Arabic STS training and Arabic STS testing.*STD* indicates the standard deviation and the other acronyms indicate the common meaning

word share. The higher word share generally leads to higher similarity. However, there are sentence pairs that contradict this theory. This observation is similar to the English datasets.

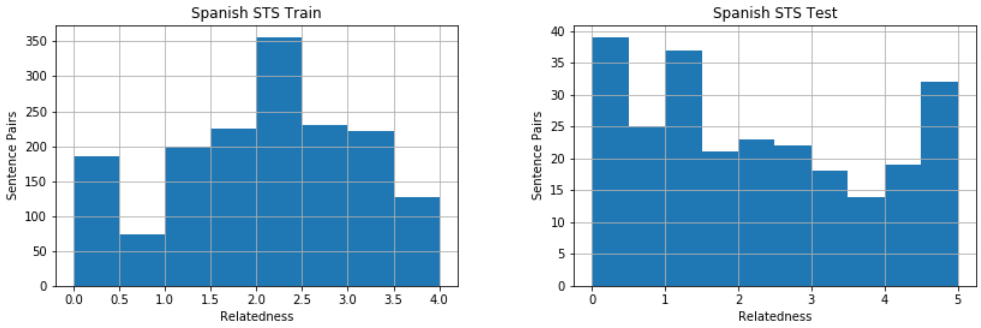
2. **Spanish STS Dataset**[12] - Spanish STS dataset that we used was employed for Spanish STS subtask in *SemEval 2017 Task 1: Semantic Textual Similarity Multilingual and Cross-lingual Focused Evaluation* (Cer et al. 2017). The training set has 1250 sentence pairs annotated with a relatedness score

[12] The Spanish STS dataset can be downloaded at http://alt.qcri.org/semeval2017/task1/index.php?id=data-and-tools

Sentence Pair	**Similarity**
1. Amás, los misioneros apunten que los númberos d'infectaos puen ser shasta dos o hasta cuatro veces más grandess que los oficiales. *(Furthermore, missionaries point out that the numbers of infected can be up to two or up to four times larger than the official ones.)* 2. Los cadáveres de personas fallecidas pueden ser hasta diez veces más contagiosos que los infectados vivos. *(The corpses of deceased people can be up to ten times more contagious than those infected alive.)*	0.6
1. La policía abatió a un caníbal cuando devoraba a una mujer Matthew Williams, de 34 años, fue sorprendido en la madrugada mordiendo el rostro de una joven a la que había invitado a su hotel. *(Police killed a cannibal while devouring a woman Matthew Williams, 34, was caught early in the morning biting the face of a young woman he had invited to his hotel.)* 2. La policía de Gales del Sur mató a un caníbal cuando se estaba comiendo la cara de una mujer de 22 años en la habitación de un hotel. *(South Wales police killed a cannibal when he was eating the face of a 22-year-old woman in a hotel room.)*	2
1. Ollanta Humala se reúne mañana con el Papa Francisco. *(Ollanta Humala meets tomorrow with Pope Francis.)* 2. El Papa Francisco mantuvo hoy una audiencia privada con el presidente Ollanta Humala, en el Vaticano. *(Pope Francis held a private audience today with President Ollanta Humala, at the Vatican.)*	3

Table 1.11: Example sentence pairs from the Spanish STS dataset. **Sentence Pair** column shows the two sentences. We also included their translations in the table. The translations were done by a native Spanish speaker. **Similarity** column indicates the annotated similarity of the two sentences.

between 0 and 4. The training set combined several datasets from previous SemEval STS shared tasks (Cer et al. 2017). Table 1.12 shows more information about the training set. There were two sources for the test set



(a) Relatedness distribution of Spanish STS training test

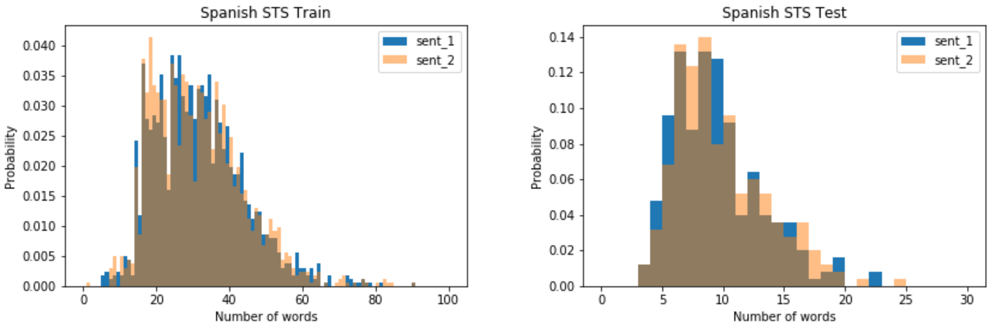
(b) Relatedness distribution of Spanish STS testing set

Figure 1.13: Relatedness distribution of Spanish STS train and Spanish STS test. *Sentence Pairs* shows the number of sentence pairs that a certain *Relatedness bin* has.

- Spanish news and Spanish Wikipedia dump having 500 and 250 sentence pairs respectively (Cer et al. 2017). Both datasets were annotated with a relatedness score between 0 and 5. Table 1.11 shows few pairs of sentences with their similarity score. The machine learning models require to predict a value between 0-5, which reflects the similarity of the given Spanish sentence pair.

Year	Dataset	Pairs	Source
2014 (Agirre et al. 2014)	Trial	56	NR
	Wiki	324	Spanish Wikipedia
	News	480	Newswire
2015 (Agirre et al. 2015)	Wiki	251	Spanish Wikipedia
	News	500	Newswire

Table 1.12: Information about the datasets used to build the Spanish STS training set. The **Year** column shows the year of the SemEval competition that the dataset got released. **Dataset** column expresses the acronym used describe a dataset in that year. **Pairs** is the number of sentence pairs in that particular dataset and **Source** shows the source of the sentence pairs.



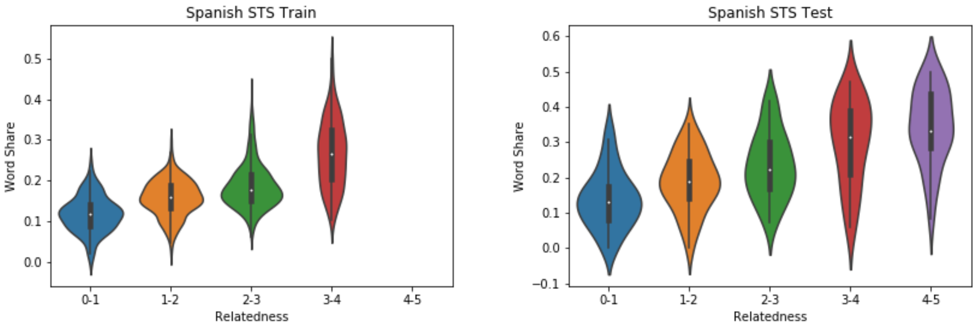
(a) Normalised distribution of word count in Spanish STS train

(b) Normalised distribution of word count in Spanish STS test

Figure 1.14: Normalised distribution of word count in Spanish STS train and Spanish STS test. *Number of words* indicates the word count and *Probability* shows the total probability of a sentence with that word count appearing in the dataset.

Similar to the Arabic STS dataset, we also analysed the Spanish STS dataset considering the same set of properties. A key challenge in the Spanish STS dataset too is that the test set is very different from the training set. As can be seen in Figure 1.13 training set has been annotated with relatedness scores 0-4 while the test set has been annotated with relatedness scores 0-5. Therefore, STS methods need to be developed in such a way that they can handle this situation. This can be observed as a weakness in this dataset, but at the same time, this property of the dataset can be exploited to measure the robustness of an STS system as well.

Furthermore, as shown in Figure 1.14 and Table 1.13 sentence pairs in the test set are shorter in word length than the sentence pairs in the train set. Therefore, STS methods working on this dataset should be able to properly handle that too. This is similar to what we observed with Arabic STS dataset.



(a) Word share against relatedness bins in Spanish STS train

(b) Word share against relatedness bins in Spanish STS test

Figure 1.15: Word share against relatedness bins in Spanish STS train and Spanish STS test. *Word Share* indicates the ratio between number of common words in the two sentences to total number of words in the two sentences against each *Relatedness* bins

Measure	Spanish STS Train		Spanish STS Test	
	Sent_1	**Sent_2**	**Sent_1**	**Sent_2**
Word Count Mean	31.23	31.02	9.03	9.34
Word Count STD	12.15	12.37	3.66	3.74
Word Count MAX	90	90	22	24
Word Count MIN	5	1	3	3

Table 1.13: Word count stats in Spanish STS training and Spanish STS testing.*STD* indicates the standard deviation and the other acronyms indicate the common meaning

The violin plot between the word share against the relatedness bin in Spanish STS is similar to the previous datasets we analysed. As can be seen in Figure 1.15, higher word share leads to a higher similarity, but some sentence pairs contradict this.

1.1.3 Datasets on Different Domains

To experiment with how our STS methods can be adopted into different domains, we also used a dataset from a different discipline which we introduce in this section.

1. **Bio-medical STS Dataset: BIOSSES**[13] - BIOSSES is the first and only benchmark dataset for biomedical sentence similarity estimation (Soğancıoğlu et al. 2017). The dataset comprises of 100 sentence pairs, in which each sentence has been selected from the TAC (Text Analysis Conference) Biomedical Summarisation Track - training dataset containing articles from the biomedical domain [14]. The sentence pairs have been evaluated by five different human experts that judged the similarity and gave scores ranging from 0 (no relation) to 4 (equivalent). The score range was described based on the guidelines of SemEval 2012 Task 6 on STS (Agirre et al. 2012). Besides the annotation instructions, example sentences from the bio-medical literature have also been provided to the annotators for each similarity degree. To represent the similarity between two sentences, we took the average of the scores provided by the five human experts. Table 1.14 shows few examples in the dataset. The machine learning models require to predict a value between 0-4, which reflects the similarity of the given biomedical sentence pair.

   The relatedness distribution is shown in Figure 1.16. It is similar to the relatedness distribution we saw in SICK and STS2017, where there were more sentences with high relatedness scores than low relatedness scores.

   As shown in Figure 1.16, sentences in the BIOSSES dataset are longer than

[13] Bio-medical STS Dataset: BIOSSES can be downloaded from https://tabilab.cmpe.boun.edu.tr/BIOSSES/DataSet.html

[14] Biomedical Summarisation Track is a shared task organised in TAC 2014 - https://tac.nist.gov/2014/BiomedSumm/

Sentence Pair	Similarity
1. It has recently been shown that Craf is essential for Kras G12D-induced NSCLC. 2. It has recently become evident that Craf is essential for the onset of Kras-driven non-small cell lung cancer.	4
1. Up-regulation of miR-24 has been observed in a number of cancers, including OSCC. 2. In addition, miR-24 is one of the most abundant miRNAs in cervical cancer cells, and is reportedly up-regulated in solid stomach cancers.	3
1. These cells (herein termed TLM-HMECs) are immortal but do not proliferate in the absence of extracellular matrix (ECM) 2. HMECs expressing hTERT and SV40 LT (TLM-HMECs) were cultured in mammary epithelial growth medium (MEGM, Lonza)	1.4
1.The up-regulation of miR-146a was also detected in cervical cancer tissues. 2. Similarly to PLK1, Aurora-A activity is required for the enrichment or localisation of multiple centrosomal factors which have roles in maturation, including LATS2 and CDK5RAP2/Cnn.	0.2

Table 1.14: Example question pairs from the BIOSSES dataset. **Sentence Pair** column shows the two sentences. **Similarity** column indicates the averaged annotated similarity of the two sentences.

the sentences in the English datasets we mentioned before. The average length of a sentence in English datasets was below 15, while in the BIOSSES dataset, the average length is around 20.

As we mentioned before, the BIOSSES dataset only has 100 sentence pairs. A dataset as small as this one can not be used to train a supervised ML method, requiring alternative approaches such as unsupervised methods and transfer learning techniques which we will be exploring in the

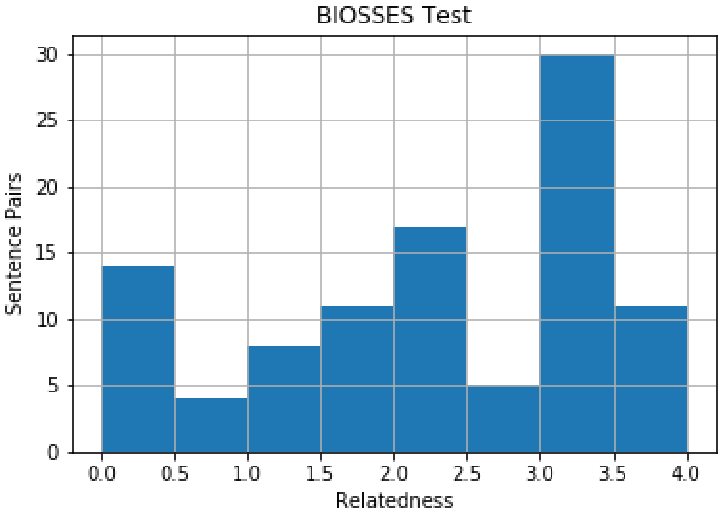


Figure 1.16: Relatedness distribution of BIOSSES. *Sentence Pairs* shows the number of sentence pairs that a certain *Relatedness bin* has.

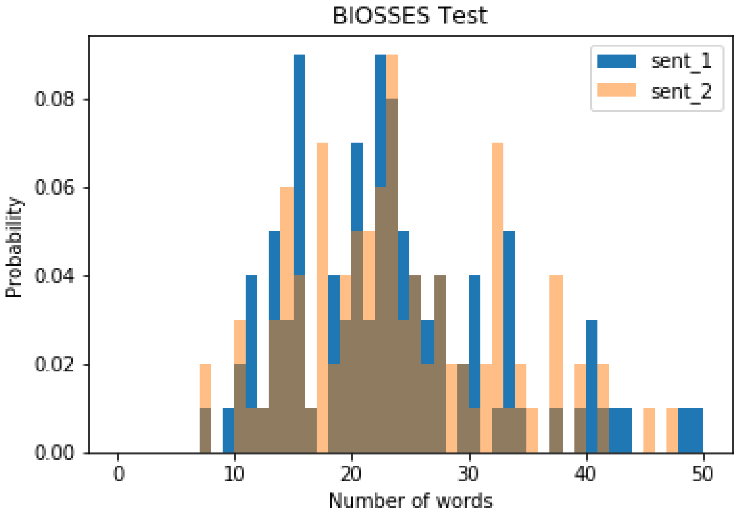


Figure 1.17: Normalised distribution of word count in BIOSSES. *Number of words* indicates the word count and *Probability* shows the total probability of a sentence with that word count appearing in the dataset.

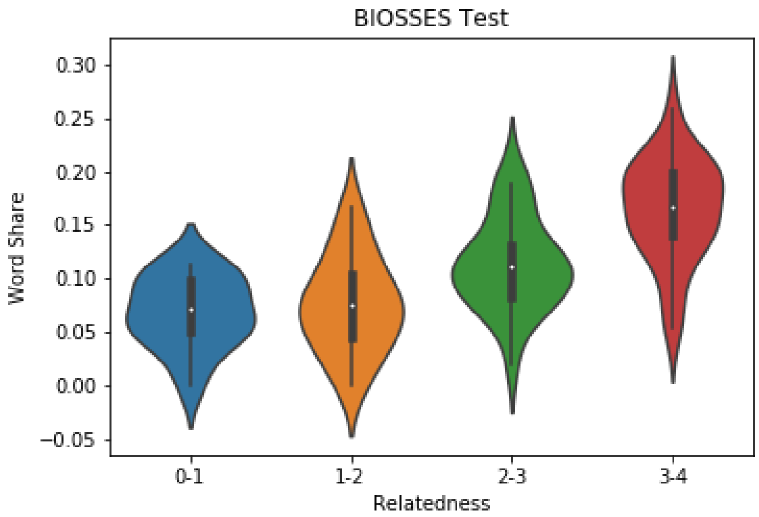


Figure 1.18: Word share against relatedness bins in BIOSSES. *Word Share* indicates the ratio between number of common words in the two sentences to total number of words in the two sentences against each *Relatedness* bins

following few chapters.

1.2 Evaluation Metrics

While training a machine learning model is a crucial step, how the model generalises on unseen data is an equally important aspect that should be considered in every machine learning model. We need to know whether it actually works and, consequently, if we can trust its predictions. This is typically called as *evaluation*. All of the datasets that we introduced in the previous section has what we call a *test* set. The machine learning models need to provide their predictions for the test set, and the predictions will be evaluated against the gold values of the test set.

There are three common evaluation metrics that are employed in Semantic

Textual Similarity tasks, which we explain in this section. We will be using them to evaluate our models throughout the first part of our research.

In the equations presented for each evaluation metrics, we represent the gold labels with X and predictions with Y. Therefore, a gold label in the i^{th} position will be represented by X_i and the prediction in i^{th} position will be represented by Y_i.

1. **Pearson's Correlation Coefficient** - Correlation is a technique for investigating the relationship between two quantitative, continuous variables. Pearson's correlation coefficient (ρ) measures the strength of the linear association between the two variables. A value of +1 is the total positive linear correlation between the variables, 0 is no linear correlation, and -1 is the total negative linear correlation.

   Pearson's Correlation Coefficient is one of the most common evaluation metrics in STS shared tasks (Marelli et al. 2014; Agirre et al. 2012; Agirre et al. 2013; Agirre et al. 2014; Agirre et al. 2015; Agirre et al. 2016). A machine learning model with a Pearson's Correlation Coefficient close to 1 indicates that the predictions of that model and gold labels have a strong positive linear correlation. Therefore, it is a good model to predict STS. Pearson's Correlation Coefficient equation is shown in Equation 1.1 where cov is the covariance, σ_X is the standard deviation of X, and σ_Y is the standard deviation of Y.

$$\rho = \frac{\mathrm{cov}(X, Y)}{\sigma_X \sigma_Y} \tag{1.1}$$

2. **Spearman's Correlation Coefficient** - Spearman's Correlation Coefficient (τ) is another common evaluation metric in STS shared tasks (Marelli et al. 2014; Agirre et al. 2012; Agirre et al. 2013; Agirre et al. 2014; Agirre et al. 2015; Agirre et al. 2016). It assesses how well the relationship between two variables can be described using a monotonic function. A monotonic relationship is a relationship that does one of the following:

   (a) as the value of one variable increases, so does the value of the other variable, *OR*,

   (b) as the value of one variable increases, the other variable value decreases.

   However, a monotonic relationship does not require a constant rate, whereas in a linear relationship, the rate of increase/decrease is constant. The fundamental difference between Pearson's correlation coefficient and Spearman's correlation coefficient is that the Pearson's correlation coefficient only works with a linear relationship between the two variables, whereas the Spearman's correlation coefficient works with the monotonic relationships as well. Spearman's correlation coefficient is shown in Equation 1.2 where D_i is the pairwise distances of the ranks of the variables

X_i and Y_i and n is the number of elements in X or Y.

$$\tau = 1 - \frac{6 \sum D_i^2}{n(n^2 - 1)} \tag{1.2}$$

In Spearman's correlation coefficient, a value of +1 is the total positive correlation between the variables, 0 is no correlation, and -1 is the total negative correlation. Therefore, similar to the Pearson's correlation coefficient, a machine learning model with a Spearman's Correlation Coefficient close to 1 indicates that the predictions of that model and gold labels have a strong positive correlation, and it is an excellent model to predict STS.

3. **Root Mean Squared Error** - Both Pearson's Correlation Coefficient and Spearman's Correlation Coefficient works only when both gold labels(X) and predictions (Y) are continuous. Therefore, for the datasets like Quora Question Pairs, where the gold labels are discrete values, Root Mean Squared Error (RMSE) is preferred for evaluation than Correlation Coefficient values. RMSE measures the distance between the gold labels and the predictions. RMSE equation is shown in Equation 1.3 where n is the number of elements in X or Y.

$$RMSE = \sqrt{(\frac{1}{n}) \sum_{i=1}^{n} (Y_i - X_i)^2} \tag{1.3}$$

In RMSE, a value close to 0 means that the error between the predictions and the gold labels are minimal. Therefore, a machine learning model with an RMSE value close to 1 indicates fewer errors and is an excellent model to predict STS.

1.3 Conclusion

Calculating the STS is an important research area in NLP, which plays a vital role in many applications such as question answering, document summarisation, information retrieval and information extraction. Most of the early approaches were based on traditional machine learning and involved heavy feature engineering. However, these approaches are difficult to be adopted in different languages and do not provide competitive results anymore. With the advances of word embeddings, and due to the success neural networks have achieved in other fields, most of the methods proposed in recent years rely on word vectors. These methods can be further categorised into supervised and unsupervised methods. Analysing STS methods belong to both of these categories would be beneficial to the community. Furthermore, exploring the ability of these methods to perform in a multilingual setting and a multi-domain setting would be a timely contribution to the NLP field.

The introduction of competitive STS shared tasks led to the development of standard datasets. We selected three recently released English STS datasets; SICK, STS2017 and Quora Question Pairs. They carry different characteristics. We exploratory analysed these datasets focussing on common properties like the size of the dataset, sentence length, common number of words etc. Furthermore, we identified specific properties of these datasets that would limit the performance of traditional STS methods like edit distance. For the multilingual experiments, we selected a Spanish and an Arabic dataset. Similar to the English STS datasets, we exploratory analysed them for certain characteristics. For the multi-domain experiments, we chose a Biomedical STS Dataset. This dataset brings a key challenge to the STS methods as it does not have a separate training set. Therefore, this dataset would provide the opportunity to evaluate various STS methods in an out-of-domain and in an unsupervised setting.

The STS shared tasks has further contributed to the development of evaluation measures in STS. In all the datasets except Quora Question Pairs, Pearson Correlation and Spearman Correlation has been used to evaluate STS methods. In the Quora dataset, Root Mean Squared Error has been used to assess the methods. We followed the same evaluation measures in order to compare our methods with other systems submitted to the competition.

In the next few chapters, we will be exploring different unsupervised and supervised STS methods. We will be evaluating them in English STS datasets, non-English STS datasets as well as out of domain STS datasets to investigate their adaptability in different environments.