UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Claudia Kittask

Metaphor Identification for Estonian

Master's Thesis (30 ECTS)

Supervisor: Eduard Barbu, PhD

Tartu 2021

Metaphor Identification for Estonian

Abstract:

Metaphors are a common facet of written and spoken language. For humans, it is pretty easy to identify and interpret metaphors, but machines struggle to match this capability. Much research about metaphors has been done in the last decades, but mainly for English using different approaches - ranging from rule-based to deep learning-based systems. As of the date of this thesis, there has been no research done for computational metaphor processing for the Estonian language. In this thesis, the research in the field of computational metaphors is explicitly applied to the Estonian language. All the methods implemented are unsupervised or semi-supervised because the resources for Estonian regarding metaphors do not exist. This thesis also attempts to incorporate contextualized embeddings from the BERT language model into metaphor identification systems to enhance performance. For testing the performance of the methods, a new evaluation dataset for the Estonian language was created[1]. This dataset contains 500 sentences, from which 232 sentences contain VERB-NOUN phrase where VERB is used metaphorically and 268 which the VERB was used literally. The best results were obtained using BERT embeddings alongside with information from Estonian WordNet.

Keywords:
Metaphors, clustering, natural language processing, unsupervised learning, semi-supervised learning, metaphor identification, BERT

CERCS: P176 - Artificial Intelligence

[1] https://github.com/ckittask/metaphor_identification_for_estonian/blob/main/evaluation_dataset/metaphor_evaluation_dataset_est.txt

Metafooride Tuvastamine Eesti Keele Jaoks

Lühikokkuvõte:

Metafoorid on kirja- ja kõnekeeles väga tavalised. Inimeste jaoks on metafooride tekstist tuvastamine ja neile tähenduse andmine lihtne. Masinate jaoks see nii kerge ei ole. Sellega seoses on metafooride valdkonnas viimastel aastakümnendal palju teadustööd tehtud kasutades nii reeglipõhiseid kui ka sügavate närvivõrkude põhiseid süsteeme. Enamus nendest teadustöödest keskendub siiski ainult inglise keelele. Selle magistritöö tegemise ajal ei olnud Eesti keele jaoks veel arvutuslikke käsitlusi metafooride kohta tehtud. Käesolevas magistritöös keskendutakse olemasolevate metafooride tuvastamise meetodite rakendamisega just Eesti keele jaoks. Kõik rakendatud meetodid on juhendamata või nõrgalt juhendatud, sest Eesti keele jaoks ei ole metafooride andmestike veel loodud. Käesolev magistritöö katsetab ka BERT mudeli kontektuaalsete vektoresituste lisamisega metafooride tuvastamise süsteemi, et tulemusi parandada. Meetodite testimiseks loodi Eesti keele jaoks uus andmestik[2]. Koostatud andmestik koosneb 500 lausest, millest 232 lauset omab TEGUSÕNA-NIMISÕNA fraasi, kus TEGUSÕNA on kasutatud metafooriliselt ja 268 lauset, kus fraasi on kasutatud literaalselt. Parim tulemus saavutati kasutades BERTi kontektuaalseid vektoresitusi koos WordNet'iga.

Võtmesõnad:
Metafoorid, klasterdamine, naturaalse keele töötlus, juhendamata õpe, nõrgalt juhendatud õpe, metafooride tuvastamine, BERT

CERCS: P176 - Tehisintellekt

[2] https://github.com/ckittask/metaphor_identification_for_estonian/blob/main/evaluation_dataset/metaphor_evaluation_dataset_est.txt

Contents

1 Introduction

The concept of a metaphor is defined as a type of non-literal language used to portray one concept or domain using the properties of other unrelated concept. For example, it is common to use properties of *journey* to explain the concept of *love*. Phrases like *"We are going our separate ways"*. and *"It has been a long and bumpy road."* describe relationships as a journey. Even though *love* and *journey* are two different things and involve different actions *love* is still understood through the concept of *journey*.

Language without metaphors would be quite dull. Using metaphors allows people to suggest much more than is actually said and provide new ways of saying it as metaphors create new similarities between seemingly unrelated concepts. Metaphors are frequently used in the spoken and written language. According to a corpus study conducted by Shutova [47] metaphors are persistent in language. Out of 761 sentences, 241 contained a metaphor. As new metaphors are created all the time, the use of metaphors is only increasing.

1.1 Motivation

In Estonian, the computational study of metaphors is almost non-existent. The author was able to find only work done by Aedmaa et al. [1], who created a dataset containing literal and non-literal particle verbs. There is no other work concerning metaphors specifically. Metaphor identification is the first step in other tasks related to metaphors, for example, metaphor interpretation, which identifies the literal meaning behind the metaphorical expression.

1.2 Contributions

The goal of this thesis is to start with the computational study of metaphors for the Estonian language. As there is no annotated dataset for metaphors in Estonian, only unsupervised and semi-supervised approaches are used to identify metaphors from the unrestricted text. The thesis is restrained from identifying VERB-NOUN phrases only where the verb is used metaphorically. Already existing approaches for metaphor identification are implemented first, and then the extensions and modifications of those approaches are proposed and evaluated. These solutions include a clustering-based method proposed by Shutova et al. [45] and two semi-supervised methods using WordNet [33] and features learned from image data [44]. As previous works on semi-supervised approaches have been using non-contextualised distributional models, this thesis also explores if contextualised embeddings from Bidirectional Encoder Representation from Transformers (BERT)

have any advantages in the task of metaphor identification. Word2vec word embeddings focus mainly on the local context window and not on global statistical information. Contextualised embeddings from BERT do not have that issue. Because of that, BERT embeddings could be useful for the metaphor identification task as the context plays a significant role in whether a word is used literally or metaphorically. The implemented methods are evaluated on the test set created in this thesis. All the implemented methods are tested on the English data to see if there are any differences in how the models behave in other languages.

1.3 Thesis Roadmap

This thesis is structured into six sections. In the Introduction section, brief overview of the topic of the metaphor in language is provided. In the Related Work section, literature review of past and current metaphor extraction systems is given. The Related Work is restricted to unsupervised and semi-supervised methods only. The Technical Background section provides an overview of relevant algorithms, resources and models. Implementation and evaluation details about the methods are given in the Method section. In Results, all of the models' evaluation metrics are reported and insights about the models are presented. Finally, in the Conclusion section, the results of the thesis are summarised and future work suggestions are provided.

2 Background

This chapter aims to give the theoretical background of metaphors. The chapter begins with the definition and origin of metaphor. Next, the view of metaphors that are used in this thesis is explained. At the end of the chapter, other non-literal figures of speech and tasks involving metaphors are introduced.

2.1 Metaphor

Using metaphors is common in language and speech, but it is challenging to explain in formal terms. Because of that - there are many different definitions for what a metaphor is. The word *metaphor* is itself a metaphor, *-phor* means *to carry, to transfer* and *meta-* means *across,behind*, indicating that something is carried across to another. The first use of the word *metaphor* comes from the Greek philosopher Aristotle [30], who describes metaphor as a form of semiotic displacement where a signifier from an object, idea or experience is used to express some other object idea or experience. The second, where the signifier is used, is called the topic, tenor or target of the metaphor and the one where the signifier is taken is called vehicle, or the source of the metaphor [58].

An example of a common metaphorical concept is *TIME IS MONEY* - here the concept TIME is the target and MONEY is the source. This metaphorical concept is exemplified by sentences:

- She is *wasting* my time.
- This unexpected bug *cost* me an hour.
- He is living on *borrowed* time.

All of these sentences express the importance of time through the concept of money. It is possible to waste time, like it is possible to waste money, it is also possible to cost someone some time like it is actual money.

According to Cambridge English Dictionary[3] metaphor is defined as:

1. *metaphor (noun) - an expression, often found in literature, that describes a person or object by referring to something that is considered to have similar characteristics to that person or object:*

- *"The mind is an ocean" and the "the city is a jungle" are metaphors.*
- *Metaphor and simile are the most commonly used figures of speech in everyday language.*

[3] `https://dictionary.cambridge.org/dictionary/english/metaphor`

2.2 Views on Metaphor

Through time, there have been three main different views on metaphors discussed in linguistics and philosophy.

Gentner [11] proposed the comparison view, which addresses metaphors as comparisons that show some preexisting similarity between the target and the source concepts.

Black [4] is the author of the interaction theory of metaphor. He uses a metaphor *man is a wolf* and through calling a *man* a *wolf*, evoking the features of a wolf to be placed to human. It means that the man is given features like a fierce, hungry hunter. In other words, the interpretation between the source and target is not about comparing the similarities that both of these concepts hold but instead of creating new similarities between them.

Lakoff and Johnson [26] in 1980 proposes the Conceptual Metaphor Theory (CMT) that went against the common thought that metaphor is merely a linguistic phenomenon. They found that the conceptual system in which people think and act is already metaphorical. The actions, ideas, and feelings are automatically done, so there is no easy way to comprehend the conceptual system directly. One more straightforward way to understand it is to analyse language. Through linguistic evidence, Lakoff and Johnson found that the ordinary conceptual system is metaphorical. They give an example where the concept is ARGUMENT, and the conceptual metaphor is ARGUMENT IS WAR. Different expressions support this metaphor like *He attacked every weak point in my argument.* or *I have never won an argument before.*. Many things that are done in arguing is partially structured in terms of war. Thinking in these terms shows how people see the act of arguing.

2.3 Systematicity of Metaphorical Concepts

Lakoff and Johnson [26] also defined the systematicity of metaphorical concepts. They differentiate three main types of metaphors: ontological, orientational and structural metaphors.

Ontological metaphors arise from our experience with physical objects and substances. When it is possible to map the experiences with entities or substances, it is also possible to refer to them and categorise them. Ontological metaphors are a way of viewing events, activities, emotions and others as entities or substances. For example:

THE MIND IS A MACHINE

- She is not *operating* that well today.

- I am not able to finish the task as I am *running out of steam.*

Orientational metaphors organize the system of concepts by how these concepts relate to each other. Most orientational metaphors are related to spatial orientations like up and down, in and out. For example:

GOOD IS UP

- He is *in top* of his game.
- They *hit a peak* this year.

BAD IS DOWN

- Her mood *has been down* for a long time.
- The discussion *fell* to the emotional level.

Structural metaphors are the most common conceptual metaphors. In this case, abstract and complex experiences are conceptualized based on a more simpler experience. For example:

ARGUMENT IS WAR

- He *attacked* my arguments.
- She *won* the argument.

In this example, the actions of debating are viewed through the concept of war. It is possible to win and lose the debate, attack the opponents and defend or protect the arguments.

Some metaphors are so often used in our everyday language that they are not recognised as metaphoric anymore. First, when a new metaphor is born, it is thought of as a novel metaphor. Then, through repeated use, it will become conventional. Goatly (1997) [12] categorised metaphors as dead, inactive and active. Some words have lost their literal meaning, and only the metaphorical meaning has stayed. For example, the *the wing of plane* does not refer to the wing of the bird anymore, which means that the word *wing* has been expanded to include the non-living things as well.

2.4 Identification of Metaphors

In 2007, Steen [50] proposed five-step procedure for identifying metaphors from text. Steen suggested that finding metaphors in discourse is not as easy as just identifying the metaphorical words. It also involves the identification of the related conceptual structures.

For that, Steen proposed five steps and illustrates them with an example sentence *"Now sleeps the crimson petal"*:

1. Find the metaphorical words. In the example sentence, the metaphorical word is *sleeps*.
2. Find the metaphorical proposition. In this step, the linguistic expression is transformed into conceptual structures, which have a form of propositions. There are three propositions: P1 (sleep, petal), P2 (mod p1 now), P3 (mod petal crimson). Here, *sleep* is from the source domain, *now* and *petal crimson* are from the target domain.
3. Find the metaphorical comparison. Here, the proposition involving concepts from two domains is transformed into an open comparison of sim(crimson petal, sleep).
4. Find the metaphorical analogy. This step transforms the open comparison into a closed comparison. In this case, the crimson petal is from BE-INACTIVE, and the sleep is an activity of a HUMAN.
5. Find the metaphorical mapping. In this case, it can be found that SLEEP means to BE-INACTIVE. Also, HUMAN properties are given to the CRIMSON PETAL. It could be inferred that the goal of sleep is the goal of be-inactive, which is rest. Time of sleep is the same as the time of be-inactive, which is night.

2.5 Other Figures of Speech

In identifying metaphors, it is essential to differentiate between metaphors and other figures of speech. Many speech figures are closely related to metaphor, including metonymy, idioms, simile and analogy.

2.5.1 Metonymy

Metonymy can be explained with an example:

*"The **ham sandwich** is waiting for his check."*[26]

Here, the expression *ham sandwich* is used to refer to an actual person who ordered a ham sandwich. This case is not a personification metaphor as it is impossible to impute human qualities to it; instead, this phrase refers to something else related to it.

2.5.2 Simile

A simile is a figurative device used to compare one item with another unrelated item to create a more vivid description. For example, simile *"He is as cold as ice"* compares someone with the coldness of ice. It can easily be transferred to metaphor by removing the comparison *"He is ice cold"*. Simile intends to create a picture of an item being like another item, whereas metaphor intends to portray one item as another item.

2.5.3 Analogy

An analogy is another figure of speech used to compare two different words through their similarity. For example, in sentence *"Life is like a box of chocolates—you never know what you are gonna get."* life is compared with chocolate as in both cases it will not be known what will be gotten.

2.5.4 Idioms

A phrase, saying or group of words with a different meaning from its parts is called an idiom. For example to *cut corners* or to *hit the sack*.

2.6 Tasks in Metaphor Processing

There are several common sub-tasks in the field of metaphor processing. The most common tasks are metaphor identification and interpretation.

Metaphor identification requires annotating metaphorical language in the text. This task is studied the most as it has a lot of available datasets. Metaphors in a text can be identified at different levels: sentence, grammatical relation or word levels. In sentence level identification, a whole sentence that contains one or many metaphorical words or phrases is classified as metaphorical—no explicit mapping to where the actual metaphor is done. Relation level or phrase level identification will annotate specific grammatical relations. Most commonly, VERB-NOUN and ADJECTIVE-NOUN pairs are used for the identification task. Given the context, word-level or token-level identification classifies each word as metaphorical or not. Only the source domain words are labelled as metaphorical or literal.

Metaphor interpretation discovers the literal meaning behind the metaphor and tries to provide a paraphrase for it. There are many issues with paraphrasing as it is possible to interpret the same metaphor in different ways. It depends on the person who is interpreting it, what is their knowledge about the source and the target domain of the metaphor.

3 Related Work

There are many algorithms for identifying metaphors. However, most of the work in this area is done in a supervised manner or using hand-coded features and knowledge sources. It is also nearly impossible to compare these methods with each other based on computed scores as most of them identify different kinds of metaphors and use various evaluation techniques and datasets. In this section, only unsupervised or semi-supervised methods for detecting metaphors in a text are described.

3.1 Selectional Preference Based Approaches

One of the first proposed systems for metaphor identification and interpretation came from Fass (1991) [9]. Their system called met* used hand-coded knowledge to look for violated semantic constraints to identify metaphors, metonymy, anomaly and literalness from short English sentences. In case the met* system detects a violation, the phrase is first tested for being metonymic. If that fails, the met* system searches the knowledge base for a relevant analogy to discriminate metaphorical relations from anomalous ones. Figure 1 illustrates how met* system works.

This system had a problem differentiating the non-literal from literal as the hand-coded knowledge used for that purpose was quite limited. Also, as some of the metaphorical expressions are very common in everyday language, it was impossible to detect any violation using selectional preferences.

Mason (2004) [34] proposed a corpus-based system called CorMet to discover metaphorical mappings between concepts. Unlike other previous metaphor identification methods, CorMet is specifically designed to work on a large class of metaphors using knowledge extracted from large corpora without using any hand-coded knowledge sources except WordNet. Instead, it works by analysing domain-specific corpora and learning the selectional preferences of the characteristic verbs of each domain. Finally, he compares the output of CorMet source-domain mappings with the Master Metaphor List [25] which contains hand-crafted metaphorical mappings between concepts.

3.2 Using Different Knowledge Sources

Peters et al. (2000) [40] detected figurative language in lexical resources using WordNet. Their idea was to parse WordNet to detect systematic polysemy. Systematic polysemy is defined by a set of word senses related systematically and predictably. Peters et al. first extracted the WordNet nodes high in the hierarchy

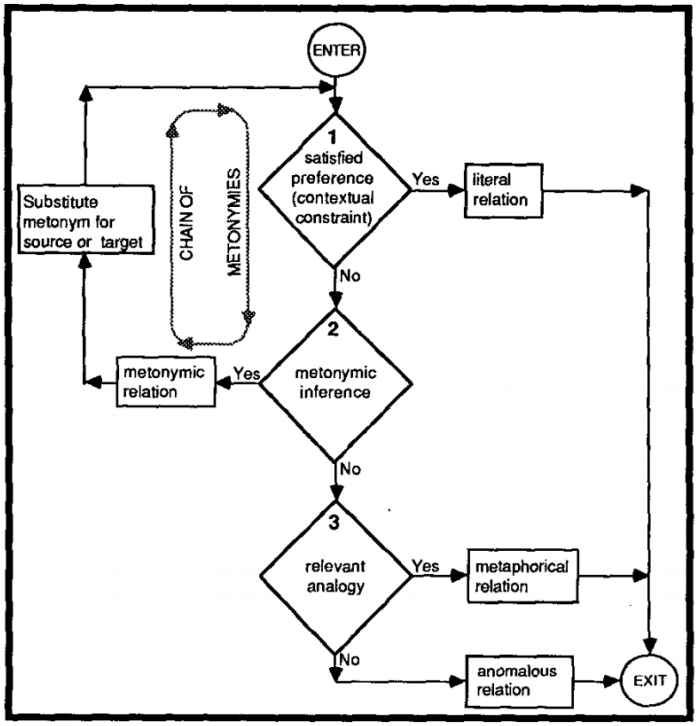

Figure 1. The met* method [9]

and shared common word forms in their descendants. With this work, authors found that such nodes are often metonymic or metaphorical.

Krishnakumaran (2007) [22] proposed a method to classify sentences into metaphorical or normal. They specifically restrict the metaphoric usage involving only nouns. Their approach does not need any training and only uses WordNet and bigram counts for the classification. They classify metaphors into three types and use different methods to identify those types.

3.3 Clustering Based Approaches

Hierarchical Graph Factorisation Clustering (HGFC) algorithm was first proposed by Yu et al. [63] and was adapted for metaphor identification task by Shutova et al. (2013) [45]. HGFC is a clustering method that assigns data to clusters in a probabilistic way. The intuition behind this method is to investigate how metaphor partitions the linguistic feature space. This system first identifies source-target domain mappings from the text and then extracts the text's metaphorical expressions. This method is fully unsupervised, which only needs a corpus to extract the features to use in the clustering.

The effectiveness of the HGFC algorithm has been shown in other research papers as well. In 2015, Pernes [39] implemented the HGFC system for mining metaphors from German historical novels. They also evaluated their method using Master Metaphor List [25].

Pramanick et al. (2018) [41] proposed an unsupervised framework for detecting metaphors that is robust to scale and adaptive to language change. They concentrated on the Adjective-Noun pairs. They used K-means clustering as the clustering algorithm and used cosine similarity, edit distance and abstractness ratings of the Adjective and the Noun as the features to the clustering method. For evaluation, they extracted the Adjective-Noun pairs from the dataset created by Tsvetkov et al. [56].

3.4 Word Embeddings Based Approaches

Wilks et al. (2013) [60] proposed an automatic metaphor detection algorithm for identifying conventionalised metaphors using WordNet. Conventionalised metaphors are metaphors that have become part of the traditional knowledge of the language and are almost impossible to detect using an algorithm based on selectional preferences. Wilks et al. base their work on a hypothesis that if a word has a sense in WordNet that is less frequent and satisfies the preference for the sentence slot it fills, then the word is metaphoric. They detect nouns and verbs

that are conventionalised metaphors.

Gutierrez et al. (2016) [14] proposed a method for learning metaphors as linear transformations in a vector space. They use these representations in the phrase-level metaphor identification task. A new dataset containing 8592 adjective-noun pairs was created for the evaluation.

Shutova et al. (2016) [44] was first to use visual features for the identification of metaphorical expressions, more specifically adjective-noun and verb-noun phrases. They experimented with combining word and phrase-level embeddings with image embeddings and classified the metaphoricity of the phrase using a threshold. The threshold was calculated from a small development set containing literal and non-literal phrases. When the similarity of the embeddings was above the threshold, the phrase was considered literal and vice versa. They found that linguistic and visual embeddings outperform the methods that only use linguistic or visual models in isolation.

In 2017, Su et al. [52] presented their automatic approach to nominal metaphor detection and interpretation. As the nominal metaphors have source and target domains, then domains present in metaphors will be less related than domains present in non-metaphors. They use high-dimensional vectors to represent the concepts as much crucial semantic information is implied in the word representation. After that, they measure the relatedness using cosine similarity between the concepts and query if one is a hyponym or hypernym of the other. If yes, then the sentence is determined to be literal.

3.5 Metaphor identification for Estonian

There has not been much work done for the Estonian language regarding metaphors or non-literal language in general. Aedmaa et al. [1] created a dataset of literal and non-literal language usage for Estonian particle verbs (PVs). Particle verbs are multi-word expressions, which consists of an adverbial particle with a base verb. The created dataset contains 1490 sentences, with 1102 non-literal and 388 literal usages across 184 PVs. Aedmaa et al. used a random forest classifier to distinguish between PVs' literal and non-literal language usage. Unigrams, abstractness rating, animacy of subject and object, case of the subject and object, case government were used as a feature for the classifier. Their method only classifies between literal and non-literal PVs but does not indicate what non-literal language is used.

4 Technical Background

This chapter gives the background information needed to understand the methods implemented in this thesis. The chapter begins by describing a clustering method called hierarchical graph factorization clustering (HGFC) employed for metaphor identification. After that, various NLP resources and tools used in the thesis like BERT, Word2Vec and WordNet, are introduced.

4.1 Hierarchical Graph Factorization Clustering

Clustering is defined as the task of grouping together a set of objects based on similarity. Each cluster is containing objects that are more similar to each other than objects outside the cluster. There are many types of clustering methods like hierarchical, subspace, overlapping clustering. In this thesis, the hierarchical clustering approach is used.

Hierarchical clustering is one of the methods of cluster analysis that builds a hierarchy of clusters. There are usually two types of hierarchical clustering strategies: agglomerative and divisive. Agglomerative clustering uses a bottom-up approach. Initially, every object is in its separate cluster. With each step of the algorithm, pairs of clusters are merged when moved up the hierarchy. Divisive is a top-down approach. Initially, all observations are in one cluster and splits are recursively made as one moves down the hierarchy.

Ye et al. [63] were the ones to propose the method of hierarchical graph factorization clustering (HGC). In broad terms, HGFC organizes data objects into a hierarchy of clusters. Each level of this hierarchy has a different granularity. One of the good properties of HGFC is that it delays the cluster assignments until the whole graph structure has been computed. This allows optimizing the assignment of data objects into clusters. Below is a more detailed description of the clustering method.

To fully understand hierarchical graph factorization clustering, defining the graph factorization clustering and bipartite graphs are needed. If not stated otherwise, the following paragraphs are based on the paper by Yu et al. [63].

The base idea of the algorithm is that graphs can be used to encode data similarity relations, where graph vertices denote some data objects, and adjacency weights define similarity between those objects. Graph factorization clustering uses probabilistic partitioning to cluster those data objects encoded as vertices into clusters.

In formal form, let $G(V, E)$ be a weighted and undirected graph with vertices

$V = v_{i i=1}^{n}$ and edges $E \in (v_1, v_j)$. $W = w_{ij}$ is the adjacency matrix of graph G, where $w_{ij} = wji, w_{ij} > 0$ if $(v_i, v_j) \in E$ and $w_{ij} = 0$ otherwise. Figure 2-(a) shows a sample undirected graph.

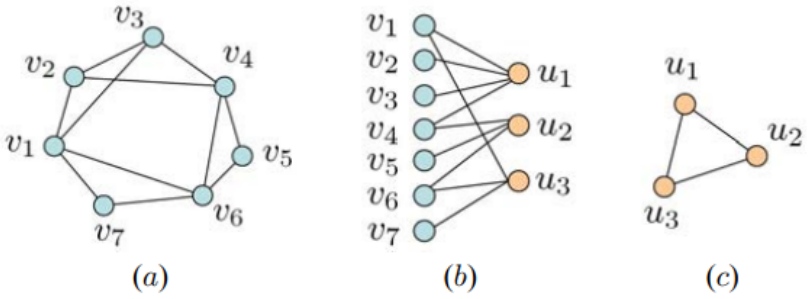

Figure 2. (a) The original undirected graph representing data affinities; (b) The bipartite graph representing data-to-cluster relations, (c) The induced affinities [63]

It is also important to understand bipartite graphs. Graph G is bipartite if its vertex set can be partitioned into disjoint subsets U and V such that every edge of G has the form u, v, where $u \in U$ and $v \in V$. Partition (U, V) is called a bipartition of the graph G and U and V are its parts [18]. More formally, let $K(V, U, F)$ be a bipartite graph. This graph has two disjoint vertex sets $V = v_{i i=1}^{n}$ and $U = u_{p_{p=1}}^{m}$ and F is a set of edges connecting V and U. Also, let $B = b_{ip}$ be the $n \times m$ adjacency matrix where $b_{ij} \leq 0$ is the weight between edge $[v_i, u_p]$. Similarity between vertices v_i and v_j can be derived from the graph:

$$w_{ij} = \sum_{p=1}^{m} \frac{b_{ip} b_{jp}}{\lambda_p} = (B \Lambda^{-1} B^T)_{ij}, \Lambda = diag(\lambda_1, ..., \lambda_m) \tag{1}$$

where $\lambda_p = \sum_{i=1}^{n} b_{ip}$ is the degree of vertex $u_p \in U$. This equation can be interpreted from the perspective of Markov random walks on graphs. $w_i ij$ is a quantity proportional to the stationary probability of direct transitions between v_i and v_j denoted by $p(v_i, v_j)$. W is normalized to ensure $\sum_{ij} w_{ij} = 1$ and $w_{ij} = p(v_i, v_j)$. As K is bipartite graph, there are no connections between vertices in V so all paths from v_i to v_j have to go through vertices in U. This means that it is possible to calculate the conditional transition probability $p(v_i, v_j)$ from v_i to v_j:

$$p(v_i, v_j) = p(v_i) p(v_j | v_i) = d_i \sum_{p} p(u_p | v_i) p(v_j | u_p) = \sum \frac{p(v_i, u_p) p(u_p, v_j)}{\lambda_p} \tag{2}$$

where $d_i = p(v_i)$ the degree of v_i. This means that the values of the adjacency matrix are the conditional transition probabilities $b_{ip} = p(v_i, u_p)$.

4.1.1 Graph factorization

For a bipartite graph K, $p(u_p|v_i) = \frac{b_{ip}}{d_i}$ is the conditional probability of transitions from v_i to u_p. If the size of U is smaller than of V, namely $m < n$, then the probability $p(u_p|v_i)$ indicates how likely data point i belongs to vertex p. This tells that one can construct a bipartite graph $K(V, U, F)$ to approximate a given $G(V, E)$, and then obtain a soft clustering structure, where U corresponds to clusters. The constructed bipartite graph is illustrated in Figure 2-(b). Equation (1) suggests that this approximation can be done by minimizing $l(W, B\Lambda^{-1}B^T)$, given a distance l between the two adjacency matrices. To make the problem easy to solve, the coupling between B and Λ is removed by $H = B\Lambda^{-1}$. Then the problem that needs to be solved is:

$$\min_{H,\Lambda} l(W, H\Lambda H^T), s.t. \sum_{i=1}^{n} h_{ip} = 1, H \in R_+^{(nxm)}, \Lambda \in D_+^{(mxm)}, \tag{3}$$

where $D_+^{(mxm)}$ is a set of mxm diagonal matrices.

It has been shown by Yu et al [63] that the cost function is non-increasing under the update rule:

$$\tilde{h}_{ip} \propto h_i p \sum_j \frac{w_{ij}}{(H\Lambda H^T)_{ij}} \lambda_p h_{jp}, s.t \sum_i \tilde{h}_{ip} = 1 \tag{4}$$

$$\tilde{h}_p \propto h_p \sum_j \frac{w_{ij}}{(H\Lambda H^T)_{ij}} h_{ip} h_{jp}, s.t \sum_p \tilde{\lambda}_p = \sum_{ij} w_{ij}. \tag{5}$$

It means that the cost function can be optimized by optimizing the h and λ.

To get the similarity between clusters u_p and u_q:

$$p(u_p, u_q) = \sum_{i=1}^{n} \frac{b_{ip} b_{iq}}{d_i} = (B^T D^{-1} B)_{pq}, D = diag(d_1, ...d_n) \tag{6}$$

A new graph is built using the calculated similarities between clusters. Previous clusters U are now vertices, and the similarities are the connection weights. This process can be seen from Figure 2(c), where the new clusters are being created. The clustering algorithm can be applied again attractively, thus creating a hierarchical graph.

The HGFC algorithm can be summarized as follows:

Algorithm 1 HGFC

Require: N nouns V, initial number of clusters m_1
Compute the similarity matrix W_0 from V
Build the graph G_0 from W_0, $l \leftarrow 1$
while $m_l > 1$ **do**
    Factorize G_{l-1} to obtain bipartite graph K_l with the adjacency matrix B_l (eq. 1, 2 and 3)
    Build a graph G_l with similarity matrix $W_l = B_l^T D_l^{-1} B_l$ according to equation 4
    $l \leftarrow l + 1$; $m_l \leftarrow$ No. non-empty clusters (eq. 5)
end
return $B_l, B_{l-1}...B_1$

To put it all together, HGFC takes in a non-negative, symmetric adjacency matrix $W = w_{ij}$ where w_{ij} represents the similarity between nodes. HGFC works by factorizing W into a bipartite graph, where nodes on one side represent nouns, and the other side represents a cluster of nodes. HGFC outputs a set of clustering if increasingly coarse granularity. The local and global clustering structures are learned via the random walk properties of the graph.

4.2 Jensen-Shannon Divergence

Jensen-Shannon divergence (JSD)[31] is a divergence measure, it quantifies how distinguishable two distributions are from each other. JSD is a symmetric version of Kullback-Leibler (KL) divergence [23], which is another statistical measure for estimating the difference between two probability distributions. KL divergence between distributions $Q = p_1, p_2, ..., p_n$ and $P = q_1, q_2, ..., q_n$ can be calculated as

$$KL(P||Q) = -\sum_{i}^{n} P(x_i) * \log \frac{Q(x_i)}{P(x_i)} \tag{7}$$

where $||$ indicates divergence.

JSD can now be calculated using the KL divergence as

$$JSD(P||Q) = \frac{1}{2} * KL(P||M) + \frac{1}{2} * KL(Q||M), \tag{8}$$

where $M = \frac{1}{2} * (P + Q)$.

4.3 WordNet

WordNet [36], created by the Cognitive Science Laboratory of Princeton University, is a lexical inheritance database for the English language. It includes verbs, nouns, adjectives and adverbs. All the words in the WordNet are grouped into a set of synonyms called synsets. Each synset has conceptual-semantic and lexical relations like super-subordinate, part-whole relations with other synsets.

For example, WordNet[4] provides four Synsets for word **coffee**:

- S: (n) **coffee**, java (a beverage consisting of an infusion of ground coffee beans) "he ordered a cup of coffee"
- : S: (n) **coffee**, coffee tree (any of several small trees and shrubs native to the tropical Old World yielding coffee beans)
- : S: (n) coffee bean, coffee berry, **coffee** (a seed of the coffee tree; ground to make coffee)
- S: (n) chocolate, **coffee**, deep brown, umber, burnt umber (a medium brown to dark-brown color)

For the first Synset, WordNet provides a direct hypernym:

- S: (n) beverage, drink, drinkable, potable (any liquid suitable for drinking) "may I take your beverage order?"

Estonian WordNet (EstWN) [37] is build upon Princeton WordNet's and EuroWordNet's principles. It contains more than 86000 concepts and 239000 relationships, and it has nouns, verbs, adjectives, adverbs and multi-word phrases.

4.4 BERT

Bidirectional Encoder Representations from Transformers (BERT) is a transformer-based language representation model developed by Google in 2018 [8].

BERT model itself is a multi-layer bidirectional Transformer encoder. It is based on the original Transformer architecture from [57]. Transformer model is composed of encoder and decoder stacks. The encoder stacks have identical structured: input to the encoder first goes through self-attention layer, outputs of this layer are inputs to the feed-forward neural network. Self-attention layer will help the model to see words in other positions and determine which of these are important for the encoding of a specific word. Decoder has almost the same structure but has encoder-decoder attention between the self-attention layer and feed-forward neural

[4] `http://wordnetweb.princeton.edu/perl/webwn`

network. This extra attention layer helps to focus on the most relevant parts of the input sentence. BERT model is composed only of the encoder stacks of the Transformer.

Pre-training of the BERT model involves two tasks trained on unlabeled data: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). The task of MLM is to predict the masked token using the context tokens. The NSP task is to predict if a sentence B follows sentence A, given sentence A. Learning to solve the NSP task will help the model to understand the relationship between the sentences.

BERT does not use full words as an input; instead, it uses WordPiece [62] embeddings. Words can be split into multiple WordPieces. For example, word *metaphorical* is split into two: *metaphor* and *ical*. BERT also uses a special classification token ([CLS]) at the start of each input sentence and a separation token ([SEP]), which denotes the end of a sentence. BERT information of the position of each WordPiece in a sentence through position embeddings. Furthermore, BERT uses segment embeddings, which will segment the input sentences. Those three embeddings (token, segment, position embeddings) are summed together and the summed representation is used as a final input embedding to the BERT model. Figure 3 shows the illustrated BERT input representation.

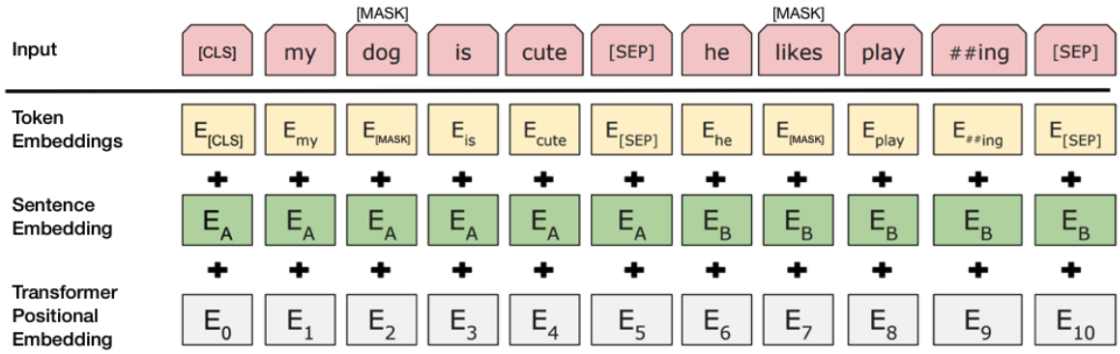

Figure 3. BERT input representation [8]

4.4.1 EstBERT

A language-specific BERT for Estonian (EstBERT) was trained at the University of Tartu in 2020 by Tanvir et al. [54]. They used Estonian National Corpus 2017 [19] for pre-training, which was the most extensive corpus available during that time. The architecture is the same as in the BERT base model, meaning 12 layers of transformer blocks, 768 hidden layers and 12 attention heads. It has shown

remarkable results in many downstream natural language processing tasks such as entity recognition, part-of-speech tagging, and text classification.

4.5 Distributional Models

To use the textual data in natural language processing there needs to be a method of converting text into the format that is understandable for different machine learning algorithms. Usually it means that tokens have to be converted into vectors containing real values, in other terms into word embeddings.

Distributional word representations are based on the distributional hypothesis proposed by Harris [15] and Firth [10]. This hypothesis states that words in similar contexts tend to have similar meanings. Mikolov et al. [35] used this hypothesis to develop two novel model architectures to get continuous vector representations of words using a large corpus. These two models are the Continuous Bag-of-Words (CBOW) and Continuous Skip-gram (SG) model. Figure 4 shows the architecture of both models.

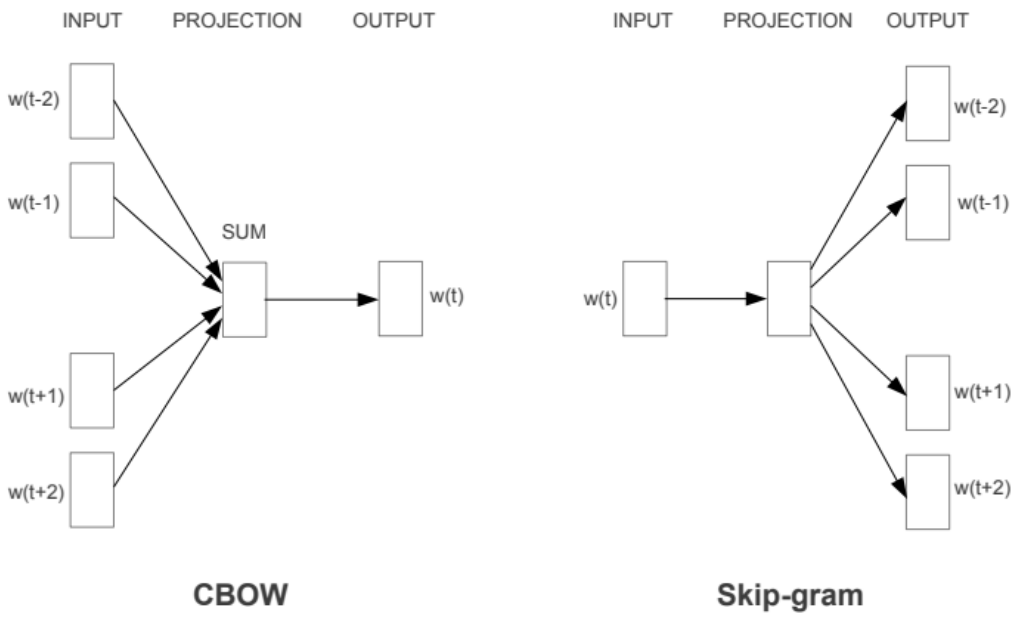

Figure 4. CBOW and Skip-gram architectures [35]

In the Continuous Bag-of-Words model, the input and output layers are contexts and the centre word one-hot encoding. CBOW is trained so that the probability of predicting a centre word, given the context, is maximized.

As it can be seen from the Figure 4, Skip-gram model learns to predict the context words from the input word.

4.5.1 Contextualized embeddings

In Word2vec methods, each word has one vector. Polysemous words, like *nail* (noun):

- nail (n) - a small piece of metal with a pointed end and a flat end that you hit into something with a hammer
- nail (n) - a thin, hard area that covers the upper side of the end of each finger[5]

will be represented with one vector as well. Word's typically do not appear in isolation; the word used depends on its context.

[5] https://dictionary.cambridge.org/dictionary/english/nail

Contextualized word embeddings fix that issue as they provide meaningful representation for words in their contexts.

4.6 ResNet

Deep convolutional neural networks are commonly used for image classification. It has also been shown that the deeper the neural network the better performance [49] it achieves, but on the negative side, deeper neural networks are slower to train and the network architecture weights are quite large. The problem of degradation and vanishing gradient occurs also with very deep neural networks. To fix these issues, He et al. proposed Deep Residual learning framework (ResNet) [17].

The idea of the ResNet is that instead of letting the network fit the actual desired mapping $H(x)$, the residual mapping $F(x) = H(x) - x$ is fit instead. This was shown [17] to be easier to optimize than the original mapping. For that ResNet utilizes identity connections that are between the layers of the network. Figure 5 shows the residual block used in the network. The curved arrow is the identity connection.

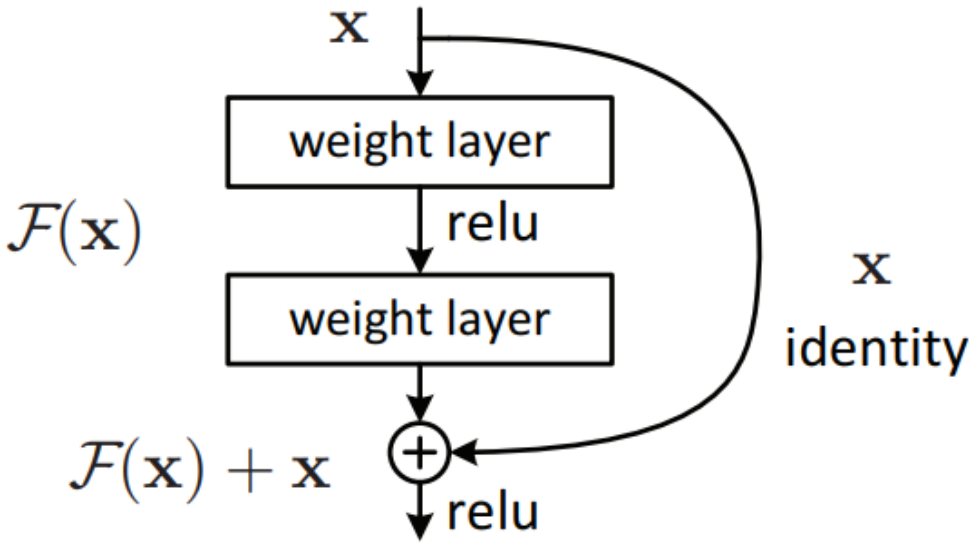

Figure 5. A building block of residual learning [17]

5 Metaphor Identification Algorithms

This chapter describes the implementation and evaluation of the metaphor identification methods used in this thesis. In total, three metaphor identification methods were implemented. These methods are built to detect VERB-NOUN phrases, where the verb is used metaphorically. The first method uses hierarchical graph factorisation clustering and selectional preferences to create metaphorical expressions. These expressions are used in the metaphor identification task. The second model uses distributed word representations and features extracted from relevant images to classify phrases as metaphorical or literal. The third model also uses distributed word representations but instead of image data it extracts knowledge from WordNet. All of the methods are tested on an Estonian and English dataset containing 500 sentences. Recall, precision and accuracy are presented for each method.

The technical background needed to understand the methods is given in the chapter section 4. All methods are implemented in Python 3.6. Text processing like tokenisation, lemmatisation, part-of-speech tagging and dependency parsing was done using Stanza [42] pipeline. Stanza is a Python library for linguistic analysis for many human languages. BERT-based models were accessed through the Hugging Face transformers library[6].

5.1 Clustering-Based Metaphor Identification Approach

In this subsection, the implemented clustering-based method is explained. First, it is reported how the hierarchical graph factorisation clustering algorithm and selectional preferences are used to identify metaphorical expressions from the text. Second, the modifications added to the HGFC using BERT contextualised embeddings are described.

5.1.1 HGFC for metaphor identification

Hierarchical Graph Factorization Clustering (HGFC) algorithm is applied to the task of metaphor identification the same way as was done by Shutova et al. [45]. Using HGFC for metaphor identification is based on a hypothesis of clustering by associations first introduced by Shutova et al. [46]. This hypothesis states that abstract concepts tend to cluster together in distributional noun clustering if they are associated with the same source domain, but concrete concepts tend to cluster by the meaning similarity instead. Shutova et al. [45] connected this idea with the findings of cognitive science, which suggested that the human brain organises abstract and concrete concepts differently [6][3]. It was found that concrete concepts

[6] https://github.com/huggingface/transformers

would organise into a tree-like structure and abstract concepts into a more complex pattern of associations. This knowledge is usable in the source-target domain mappings, where the source is considered more concrete and targets more abstract concepts.

The HGFC algorithm involves distributional learning from an extensive collection of text; because of that, it is essential to choose a representative text corpus. Estonian National Corpus 2017 (ENC2017) [19] was chosen for Estonian as it contains a wide range of texts: web texts, articles, news, books and more. EstBERT is also trained on this corpus, making the comparison between HGFC and other implemented models fairer. This corpus contains four sub-corpora: Estonian Wikipedia Corpus 2017, Estonian Web Corpus 2013, Estonian Web Corpus 2017 and Estonian Reference Corpus 1999-2008. Initial statistics of this corpus are shown in the top row of Table 1.

	Documents	**Sentences**	**Words**
Initial	3.9M	87.6M	1340M
After cleanup	3.3M	75.7M	1154M

Table 1. Estonian National Corpus 2017 statistics

An already cleaned and filtered corpus which was created for pre-training EstBERT [54] was used in this thesis. Authors of EstBERT filtered out all duplicates and paragraphs not written in Estonian using a language-detection library[7]. Furthermore - they filtered the corpus using hand-written heuristics to filter out sentences and paragraphs that were too short, contained too many stop-words or punctuation marks. Statistics after the corpus was preprocessed is shown in the last row of Table 1.

The English implementation of HGFC was trained using Open Super-large Crawled Aggregated coRpus (OSCAR) [38]. This is a substantial multilingual corpus that has been obtained via filtering Common Crawl data and using language classification. More specifically, a deduplicated version of the English sub-corpus was used. As the entire corpus for English was too big compared with the ENC2017 size - only parts of the corpus were downloaded and used.

The 2000 most frequent noun lemmas were extracted from the corpus. The amount of nouns was chosen based on the availability of examples - a reasonable amount of examples are needed to be able to extract representational grammatical features. The nouns in the corpus were lemmatised using the Stanza pipeline prior to the

[7] https://github.com/shuyo/language-detection

Estonian	English
aasta	time
inimene	year
aeg	people
laps	day
asi	way
päev	thing

Table 2. Sample from the 2000 most frequent noun lemmas.

N: mäng	**N: poliitika**	**N: game**	**N: politics**
16020 mängima	1419 tegema	55883 play	493 play
7703 tegema	1357 ajama	15310 win	382 enter
6366 võitma	832 viima	6824 make	381 say
...	...	...	...
75 venima	11 kaaluma	65 attack	20 destroy
24 lihvima	10 päästma	23 explode	14 kill

Table 3. Sample context vectors for nouns *mäng-game* and *poliitika-politics*.

frequency being calculated. Table 2 shows some extracted most frequent nouns for Estonian and English.

After the extraction of nouns, the corpus was parsed using Stanza's dependency parser. All VERB-SUBJECT, VERB-DIRECT_OBJECT and VERB-INDIRECT_OBJECT relations with the 2000 most common nouns were extracted from the parser's output.

Grammatical features were used for clustering as it is expected that target concepts associated with the same source concept should appear in similar lexico-syntactic environments. Because of this - clustering concepts using grammatical relations (GRs) allow the capture of their relatedness by association and therefore can detect metaphorical expressions. Table 3 shows an example of extracted grammatical features for nouns *mäng-game* and *poliitika-politics*. As it can be seen, the context vector of the noun *mäng* contains literal terms (e.g., mängu *mängima*, mängu *võitma*) and also metaphorical terms (e.g., mäng *venib*, mängu lihvima). The same could be seen from the context vector of *poliitika*.

The resulting NOUN-VERB feature matrix containing relative feature frequencies was then used to calculate the NOUN-NOUN similarity matrix on the co-occurrence vectors. This similarity matrix was constructed using the Jensen-Shannon diver-

Estonian	**English**
TEEKOND	JOURNEY
LAPS	CHILD
TULI	FIRE
MASIN	MACHINE
SÕDA	WAR
TAKISTUS	OBSTACLE
RAHA	MONEY
ELU	LIFE
TEE	PATH
LIIKUMINE	MOVEMENT

Table 4. Subset of extracted source domains.

gence described in Section 4.2.

This similarity matrix was the input to the HGFC algorithm. The initial number of clusters m_0 was set to 800. This number was based on Shutova et al. [45] doing the same in a similar situation. For the following levels, m_l was the number of non-empty clusters on the parent level. The algorithm terminated when all nouns were assigned to exactly one cluster. For every two adjacent levels, 1000 iterations of updates were run of h and λ (see Equation 4 and 5).

After the HGFC algorithm was run, the source-target domain mappings were extracted using a predefined set of source domains (given in Appendix). This set was extracted from the Master Metaphor List [25]. The Master Metaphor List contains already mapped source and target domains. All the source domains were randomly chosen from the list and filtered by checking if the source domain was inside the 2000 most frequent nouns. A predefined source domain set ensures that the source already has been mapped to multiple targets.

An example of extracted source domain mappings can be seen in Table 4.

The source-target domain mappings were derived using the probability of the word v_i to be assigned to cluster $x_p^{(l)} \in X_l$ at level l. The probability was calculated with the following equation:

$$p(x_p^{(l)}|v_i) = \sum_{X_{l-1}} ... \sum_{x^{(1)} \in X_1} p(x_p^{(l)}|x^{(l-1)})...p(x^{(1)}|v_i) = (D_1^{(-1)} B_1 D_2^{-1} B_2 ... D_l^{-1} B_l)_{ip} \tag{9}$$

Here, B is the adjacency matrix, which represents the connections between the clusters at an upper and lower level of clustering. D is $diag(d_1, ..., d_n)$ where

$d_i = \sum_{p=0}^{m} b_{ip}$.

Mappings were extracted from level four based on comparing the source-target domain mappings from each level. Six top-ranked clusters were then selected from this chosen level. The cluster containing the input concept was removed as it is expected to represent the source's literal meaning. For example, for noun *fire*, the literal cluster contained concepts like *heat*, *match*, *flame* and *blaze*. All the remaining five clusters represent the target concepts associated with the source.

Table 5 and 6 shows the extracted targets for sources *tuli* and *mäng*. Each noun-to-cluster mapping represent a new conceptual metaphor, for example **EMOTSIOON ON TULI** and **PROBLEM IS DISEASE**.

SOURCE: TULI
TARGET 1: probleem, arvamus
TARGET 2: hing, süda, tunne, huul, naeratus, unenägu, emotsioon
TARGET 3: kriis, oht, tähtaeg, pilk
TARGET 4: pimedus, olend, vari
SOURCE: HAIGUS
TARGET 1: mure, takistus, koormus, häda, pisar, õnnetus
TARGET 2: vaidlus, pinge, hirm, raskus,
TARGET 3: konkurens, vanelane, häire
TARGET 4: segadus, kahju, viha, müra, hinnatõus

Table 5. Example source-target domain mappings for Estonian.

SOURCE: FIRE
TARGET 1: feeling, sin, trust, fantasy
TARGET 2: impression, news,
TARGET 3: mistake, trouble, war,
TARGET 4: protest, crowd,
SOURCE: DISEASE
TARGET 1: loss, damage, harm, injury, fraud, fault , problem
TARGET 2: stress, tension, anxiety, poverty, abuse
TARGET 3: addiction, failure, destruction
TARGET 4: reaction, attack, evil

Table 6. Example source-target domain mappings for English.

mure-haigus
kasvama, jääma, vaevama, rõhuma, põdema, kaduma, algama, võtma, ravima, põhjustama, panema, olema, nägema, jätma, hakkama, minema, saama, tulema, ennetama, tooma, kandma, vältima, leidma, tekkima, tekitama, avaldama, mõistma, murdma, leevendama, taanduma
problem-disease
cause, prevent, include, treat, relate, develop, affect, diagnose, know, occur, get, lead, have, become, catch, grow, suffer, combat, diagnose, begin, reduce, manage, need, keep, give, associate, cure, overcome

Table 7. Extracted salient features for *haigus-disease* and*mure-problem* cluster.

After the extraction of the source-target mappings, the metaphorical expressions were obtained. The salient features that lead to the input noun being strongly associated with the extracted clusters were harvested. These features were selected by ranking the features according to the joint probability of the feature (f) occurring both with the input noun (n) and the cluster (c):

$$p(n, c|f) = p(n|f) * p(c|f), \tag{10}$$

where $p(n|f)$ and $p(c|f)$ represent the ratio between the frequency of the feature f and the total frequency of the input noun and the cluster, respectively. Highly ranked features represent the source domain vocabulary and - when used alongside the target - are metaphorical. The top 50 features were taken from the extracted list. Table 7 shows an example of extracted salient features for the *tuli-fire* and *probleem-problem* cluster.

These metaphorical features were filtered through selectional preference (SP). As was done in [45], SPs were used to see how well the extracted features describe the source domain.

For all verbs in the feature list, nominal argument distributions were derived. SP classes are created by using the algorithm created by Sun et al. [53]. They use

unsupervised ways to create SPs. They first take the GR relations associated with verb and then extract all the argument heads in these relations if they occur with a frequency larger than 20 with more than three verbs. Lastly, they cluster the resulting N most frequent argument heads into M classes using the spectral clustering method.

The measure of Resnik [43] is used to quantify how well a particular argument class fits the verb. Resnik measures the SP strength $S_R(v)$ of a predicate as a Kullback-Leibler distance between two distributions: the prior probability of the noun class $P(c)$ and the posterior probability $P(c|v)$ of the noun class given the verb:

$$S_R(v) = D(P(c|v)||P(c)) = \sum_C P(c|v) \log \frac{P(c|v)}{P(c)} \tag{11}$$

To quantify how well argument class fits the verb, Resnik also defines the selectional association $A_r(v, c)$:

$$A_R(v, c) = \frac{1}{S_r(v)} P(c|v) \log \frac{P(c|v)}{P(c)} \tag{12}$$

All the nominal arguments of the verbs in the feature lists were ranked using their selectional association with the verb. Features whose top five arguments also contained the source concept were kept. For example, the common verb *cause* for *illness* and *problem* cluster (e.g. "cause an illness", "cause a problem") is filtered out by the selectional preference filtering. Whereas verbs *treat* and *heal* were not filtered.

After the filtering, the Stanza pipeline was used to extract grammatical relations, where the noun is from one of the target domains and the verb from the source domain vocabulary.

The whole framework is illustrated in Figure 6.

5.1.2 Extending HGFC using BERT

In metaphor identification, context plays a significant role in determining if the phrase is metaphorical or not. For example, the VERB-NOUN pair *pierced-skin* would be extracted from the sentence *The cold air pierced my skin.*, it would not be possible to say if this phrase is metaphorical or not as it can also be used in a literal sense *When skin is pierced, there is blood.*. Previously described HGFC uses grammatical features extracted from a corpus and does not use any other

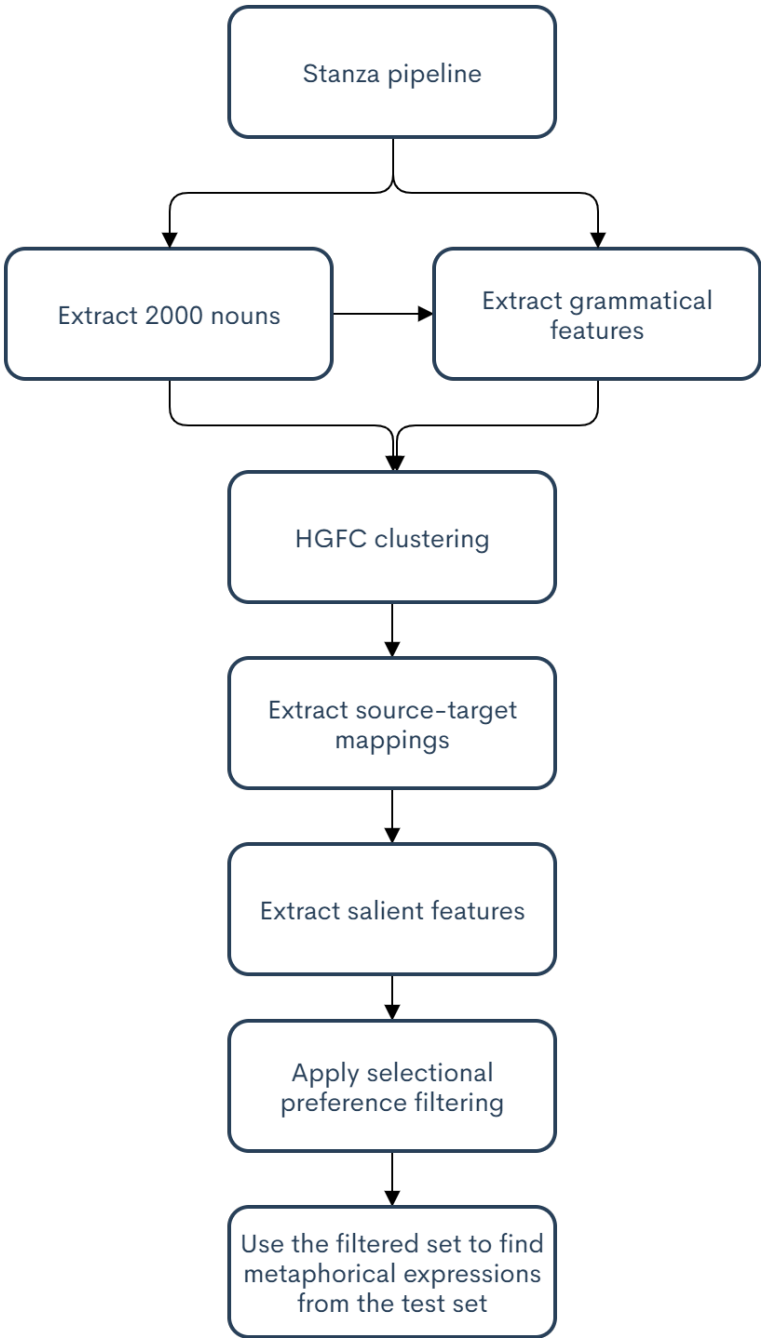

Figure 6. HGFC for identification of metaphorical expressions.

information about the context of the phrase. Here is the place where the BERT context-dependent embeddings come into play.

Previous studies [24][16] have shown that BERT contextualised embeddings can be successfully used in an unsupervised manner. There has been no research about how BERT embeddings perform in an unsupervised or semi-supervised manner in the task of metaphor identification. Because of that, the usefulness of BERT embeddings was unclear prior to this thesis.

The HGFC algorithm is extended by replacing the grammatical features with the contextualised sentence embeddings from BERT. For English, the BERT-base-cased [8] model is used, and for Estonian, EstBERT [54] is used. Both models have the same amount of layers and parameters. The unique token [CLS] representation, which is a pooled output from the last four hidden layers summed together, is used as the final embeddings.

As the BERT embeddings are contextualized, meaning that the same word has different embeddings depending on the context, it is not reasonable to input one word to the BERT model and use the embedding representation of it as a feature. Phrase-level embeddings make more sense in terms of BERT as it provides some more context to the noun. All possible NOUN-VERB phrases were created, and the token [CLS] representation from BERT was extracted for each phrase. In the end, each noun was represented with a set of embeddings, creating a matrix. A distance measure based on matrix norms introduced in [59] was used to find the similarity between two matrices $A = (a_{ij})$ and $B = (b_{ij})$:

$$sm(A, B) = \frac{||A^T B||}{\sqrt{||A^T A|| * ||B^T B||}}, \tag{13}$$

where $||Z|| = \sqrt{\rho(Z^T Z)}$ and $\rho(X)$ denotes the largest absolute eigenvalue of a squared matrix X. This similarity measure fulfills all the properties of similarity measures. It is upper bounded, reflexive and symmetric.

The computed similarity matrix was used as an input to the HGFC algorithm. The source-target domain mappings and features were induced the same way as HGFC with grammatical features.

An example of the extracted source-target domain mappings can be seen from the Table 8 and 9. BERT based HGFC model produced bigger clusters and overall fewer levels. The clusters were extracted from the second level and not from the fourth, as was done with the regular HGFC using grammatical features.

5.2 Semi-Supervised Approaches

Two semi-supervised approaches of detecting NOUN-VERB metaphors from the unrestricted text were implemented in this thesis. The first model uses word embeddings alongside visual features extracted from pre-trained ResNet-18 model [17] trained on ImageNet classification task. The second model also uses word embeddings, but it uses information extracted from WordNet instead of visual

SOURCE: TULI
TARGET 1: revolutsioon, vaev, stress, kaebus,
TARGET 2: sõjavägi, koormus, piirang, poliitika, seis, piiramine, lahing
TARGET 3: tunne, tõde, rahu, torm, nauding, võime, suhe, kaaslane
TARGET 4: hoiatus, tasumine
SOURCE: HAIGUS
TARGET 1: töötu, vang, kahju, vastutus, õnnetus, andmine, erand
TARGET 2: rahvas, levik, viirus, tagasiside, mäng, omand, uurimus, valitseja
TARGET 3: tegevus, asjaolu, kaubandus, saaja
TARGET 4: erand, rünne, tüli, trahv

Table 8. Example source-target domain mappings from BERT based HGFC

SOURCE: FIRE
TARGET 1:mistake,
TARGET 2: feeling, trust, emotion, fear, difficulty, pain, storm, excitement, horror, doubt, concern
TARGET 3: war, negotiation, regime, investigation, society
TARGET 4: passion, talent, understanding, partnership, equity, reputation, justice
SOURCE: DISEASE
TARGET 1: error, situation, attack, loss, condition
TARGET 2: crime, damage, death, injury, circumstance, act, violation, failure, delay, fraud, religion, crash
TARGET 3: disability, accident, shooting, encounter, shock, stress, difficulty, poverty, tension, complexity
TARGET 4: content, document, guide, detail, stage, course, journal

Table 9. Example source-target domain mappings from BERT based HGFC

features. Estonian WordNet version 2.3.2 was accessed through EstNTLK library [28], an open-source tool for Estonian natural language processing. English WordNet version 3.0 was accessed through the NLTK library [32], which is a platform for building Python scripts that work with natural languages. Both implemented methods only need a small seed-set for defining the threshold for classification.

5.2.1 Word embedding and visual features based approach

Incorporating visual features with linguistic features to the metaphor identification model approach came from Shutova et al. (2016) [44]. The main idea behind it is that human meaning representations are not only grounded in linguistic features, but in the perceptual system as well [2][3]. For example - it is possible to show the target concept through different sources using images. Figure 7 shows the images about *love* through different sources. Each of the images represents various ways of representing *love*.

Figure 7. Different representations of love.

Information from visual features have previously been incorporated into models of semantic similarity [7], semantic relatedness [29] and compositionality [61] and have been shown to increase the overall performance.

The same approach of Shutova et al. [44] is implemented. Their approach incorporates linguistic features and visual features and applies different arithmetic operations to classify phrases as literal and metaphorical based on the pre-defined threshold.

To get the linguistic features a Skip-gram model was trained. For the Estonian language, the Estonian National Corpus 2017 [19] was used, and for English, part of the OSCAR [38] corpus was used. Both corpora were lemmatised, tagged and parsed with Stanza Pipeline [42]. All the lemmatised words with a frequency less than 100 were ignored. Learned embeddings were 100-dimensional. Word- and phrase-level embeddings were trained in two separate stages. In the first stage,

word-level embeddings were obtained using the skip-gram with negative sampling. In the second stage, phrase-level embeddings were learned over the same corpus. For this, the context embeddings were taken from the first stage and were kept fixed. Verb-noun phrases were extracted from the parsed corpus. For learning phrase-embeddings, no frequency cutoff was set. Both embeddings were trained for three epochs using window size five and ten negative samples per word-context pair.

To get visual features for words and phrases - 10 images were automatically downloaded from Google Images[8] using Selenium[9]. More than one image is used to get a different representations of the word. It improves the final representation of the concept by alleviating the problem that not all images might not contain the correct concept. The Estonian phrases and words were first automatically translated into English to get better images from Google Images. Images using Estonian keywords often do not represent the keyword the best as there is less image content for Estonian than there is for English. Tilde Translator [10] was used for the translations from Estonian to English.

For example, the phrase *mägesid liigutama* was first translated into English - *move the mountains*. Then the phrase was split into verb *liigutama* and noun *mägesid*, both were first lemmatised and then translated separately, *mägi* into *mountain* and *liigutama* into *move*. Through visual expertise, the word *to* was added to the verb as this keyword yielded to better and more representative images. Figures 10, 8, 9 show the example images downloaded for the phrase, verb and noun.

Figure 8. Example images downloaded for word *mägi*

512-dimensional embeddings for the images were extracted from the ResNet-18 model's average pool layer. The architecture of the model can be seen from Figure 11.

This model was chosen because it has been trained on the ImageNet classification task, which means that the model has learned to represent various types of concepts

[8] https://images.google.com/
[9] https://www.selenium.dev/
[10] https://translate.tilde.com/#/

Figure 9. Example images downloaded for word *liigutama*.

Figure 10. Example images downloaded for phrase *mägesid liigutama*.

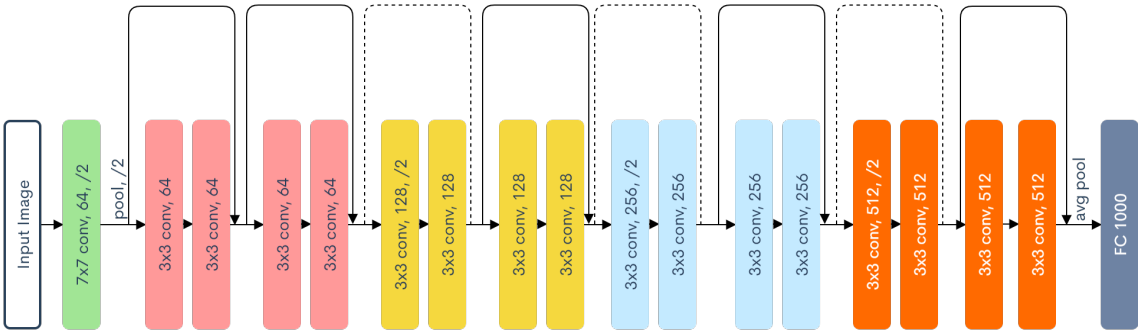

Figure 11. Obtaining visual features

as the ImageNet itself contains more than 80 000 concepts from WordNet. For the final visual embeddings, the average from all the extracted image embeddings was taken. Figure 12 shows the detailed workflow of how the visual features were obtained.

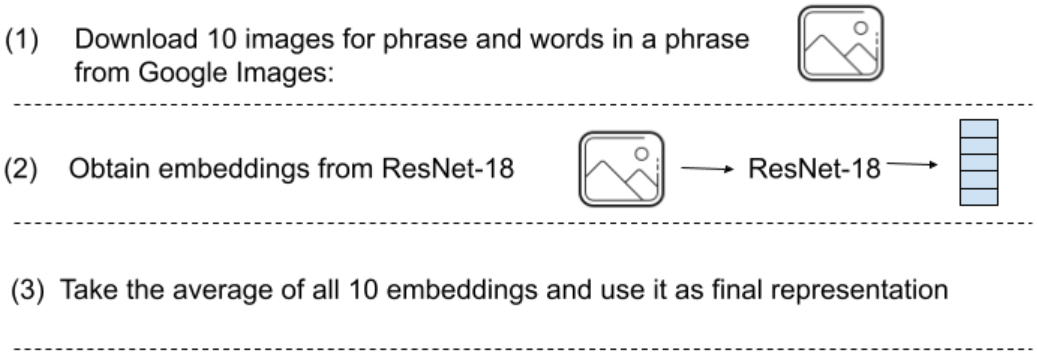

Figure 12. ResNet-18 architecture

As was done in [44], different ways of measuring metaphoricity were used. First, only word-level embeddings (WordCos) were used to see if the words in the phrase are from the same domain. It can be determined by measuring the similarity between the representations of the two words in a phrase:

$$WordCos(word_1, word_2), \tag{14}$$

where $word_1$ is a noun and $word_2$ is a verb. The similarity is measured using cosine similarity. In metaphorical expressions, one word should come from the source domain and the other from the target domain. It means that the similarity should be lower for such phrases than literal phrases, where both words come from the target domain. For example,

Phrase-level embeddings were also tried experimentally:

$$PhrasCos1 : cos(phrase - word_1, word_2) \tag{15}$$

$$PhrasCos2 : cos(phrase - word_2, word_1) \tag{16}$$

$$PhrasCos3 : cos(phrase, word_1 + word_2) \tag{17}$$

Here, the phrase is the phrase embedding vector and $word_1, word_2$ is defined above.

Phrase	Label
ajudele hakkama	M
kirgi kütma	M
valgus särama	L
häält kuulma	L

Table 10. Example seed phrases for Estonian.

The image embeddings were added to the linguistic ones in two ways: middle fusion and late fusion. In middle fusion, the vectors of visual and linguistic representations were first L-2 normalised and concatenated, and then metaphoricity score was computed using this new representation. In late fusion, the metaphoricity scores were calculated separately, and the average score of both metaphoricity scores was used as the final score.

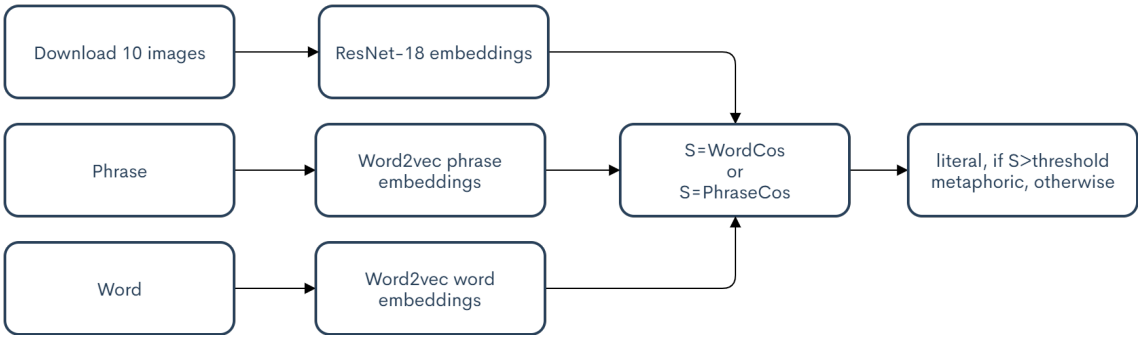

Figure 13. Metaphor identification framework using Word2vec embeddings and visual features.

A threshold for metaphoricity was needed to classify the phrases. This threshold was found by using a small seed set.

The Estonian seed set was manually collected from different sources. Metaphorical phrases were taken from Õim's book *Komistusi Metafooridega*[64]. Literal phrases were collected from local newspapers. The created seed-set contained 30 metaphorical phrases and 30 literal phrases. Table 10 shows some of the seeds in the set.

An English seed-set was created by filtering VU Amsterdam Metaphor Corpus [51]. The final seed-set for English also consisted of 30 metaphorical and 30 literal phrases. Table 11 shows some of the seeds in the set.

The classification accuracy was maximised in this seed set to optimise the scoring

Phrase	Label
breath life	M
break voice	M
pour coffee	L
lend money	L

Table 11. Example seed phrases for English.

methods. All the values above the threshold were considered literal, and all the values below were considered metaphorical. The thresholds were different for each method of calculating the final threshold. The thresholds ranged from 0.4 to 0.6. The comprehensive framework of this approach is illustrated in Figure 13.

5.2.2 Extension using BERT embeddings

The visual features based model was extended by replacing Word2vec word, and phrase embeddings with BERT contextualised embeddings. Word2vec embeddings do not use the full context to classify the phrases; using BERT embeddings would fix that.

First, the word embeddings were replaced with BERT embeddings. For that, the whole sentence was input into the chosen BERT model; the token embeddings representing the word were extracted from the model and added together if the word is split into multiple tokens. The embeddings were taken from the last four hidden layers from the BERT model. The final word embedding v_w was calculated as:

$$v_w = \sum_{j=1}^{k} v_j * \frac{1}{k} \tag{18}$$

where k is the number of word pieces the word is split into; v_j is the BERT's representation of the j^{th} word piece.

Second, to get the phrase embedding, the phrase itself was input to the BERT model, and the [CLS] representation was used as the final phrase embedding. All other steps are the same as in the original version. Figure 14 shows the visual representation of the whole framework.

The previously created seed set was extended to be used with this extension. As the BERT extension uses the whole sentence, the seed set had to include the whole sentence. For each phrase, a sentence using this phrase was included. Sentences containing metaphorical phrases for Estonian were taken from the Õis's book *Komistusi Metafooridega* and sentences containing literal phrases were manually

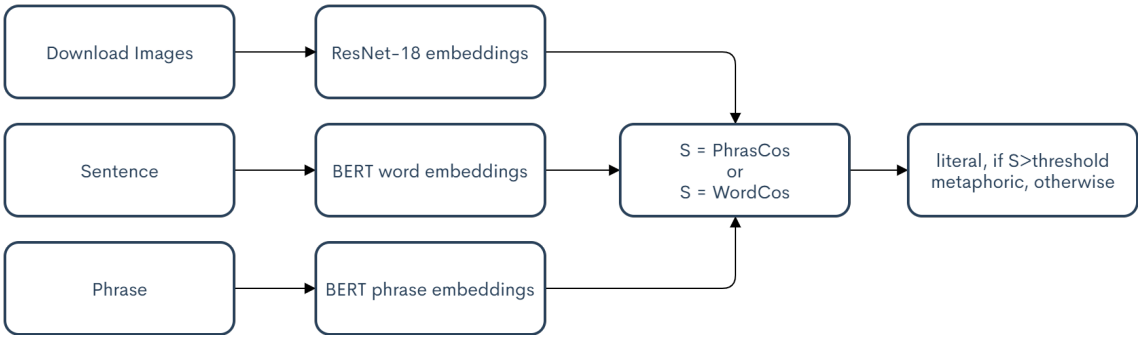

Figure 14. Metaphor identification framework using BERT embeddings and visual features.

extracted from the local newspapers. For the English seed set, the sentence from where the initial phrase was extracted was used. The calculated threshold was different for each way of calculating the final similarity score. Overall the threshold ranged from 0.3 to 0.8.

5.2.3 Word embedding and WordNet based approach

The idea behind using WordNet and embeddings for phrase-level metaphor identification originates from Mao et al. [33]. It is based on two hypotheses. The first hypothesis (H1) is that it is possible to identify metaphorical words based on the sense it takes within its context and the literal sense it has. If the literal sense and the sense from the context come from different domains, the word is metaphorical. This hypothesis is based on the theory of Selectional Preference Violation[43]. The second hypothesis (H2) states that it is more common to see the literal sense than its metaphorical sense.

As can be seen from the Figure15, the first step was the training of the word embeddings. CBOW and Skip-gram embeddings were trained using the same settings as were used in the previous visual feature-based approach [44]. It means that CBOW and Skip-gram models were trained for three epochs with a context size of five to obtain 100-dimensional word input and output vectors. All words with a total frequency of less than 100 were excluded. In total, ten negative samples were randomly selected for each centre word. As was said in Mao et. al [33] - the output vectors from the CBOW model showed worse results compared with input vectors and input vectors combined with output vectors. Because of this, the pure output vectors of CBOW were not used in this thesis.

The second step used the input sentence, the target word and its context words. The target word was the word for which the metaphoricity was to be determined.

In this case, it is a verb. After that, all the possible senses of the target word were added into a candidate set W by extracting the synonyms and direct hypernyms of the target word from WordNet. This set was then augmented with the inflexions of the extracted synonyms, hypernyms and the target word. All auxiliary verbs were excluded as these words do not have much linguistic meaning. Context words were all the other words in the sentence that were not the target word.

In the third step, the best-fit word was identified. This word represents the literal sense that the target word most likely took, given the context. For example, for the sentence *Ta neelas raamatuid ajaloost.*, the target word is *neelas* and the best fit word is *luges*. If the target word itself is literal, the best-fit word might be the same as the target word or very similar to it. Given an input sentence s and target word w_t, $w^* \in W$ is the best fit word for target w_t and w_c, which is the surrounding context for w_t. Context embedding v_c^i was computed by averaging out the input vectors of each context word of w_c:

$$context\_vector = \frac{1}{m} * \sum_{n=1}^{m} v_{c,n}^{iT} \tag{19}$$

where m is the number of context words and $v_{c,n}$ is the nth context word embedding. Each candidate word $k \in W$ was ranked by measuring its similarity to the context input vector v_c^i. The best fit word was the candidate word with the highest similarity to the context:

$$w^* = argmax_k SIM(v_k, v_c) \tag{20}$$

where v_k is the vector of a candidate word $k \in W$.

In the last step, the cosine similarity between the lemmatized best fit and target word was calculated:

$$SIM(w^*, W_t) = cos(v_{w^*}^i, v_{w_t}^i) \tag{21}$$

If the lemmatized best-fit word was not included in the embeddings model, the next best-fit word was used. Also, if none of the candidate words was in the embeddings model, the phrase was automatically classified as literal as statistically there are more literal phrases than metaphorical ones.

Pre-defined threshold τ was used to classify the word as metaphorical or literal. The If the similarity was above the threshold, the target word was classified as literal, otherwise metaphorical. The threshold τ was empirically determined on a separate seed-set of metaphorical and literal VERB-NOUN phrases by maximizing the classification accuracy on this seed-set. The used seed-set for Estonian and English was the same as that was used in the previous method. The acquired threshold was $\tau = 0.8$

Figure 15 shows the metaphor identification framework.

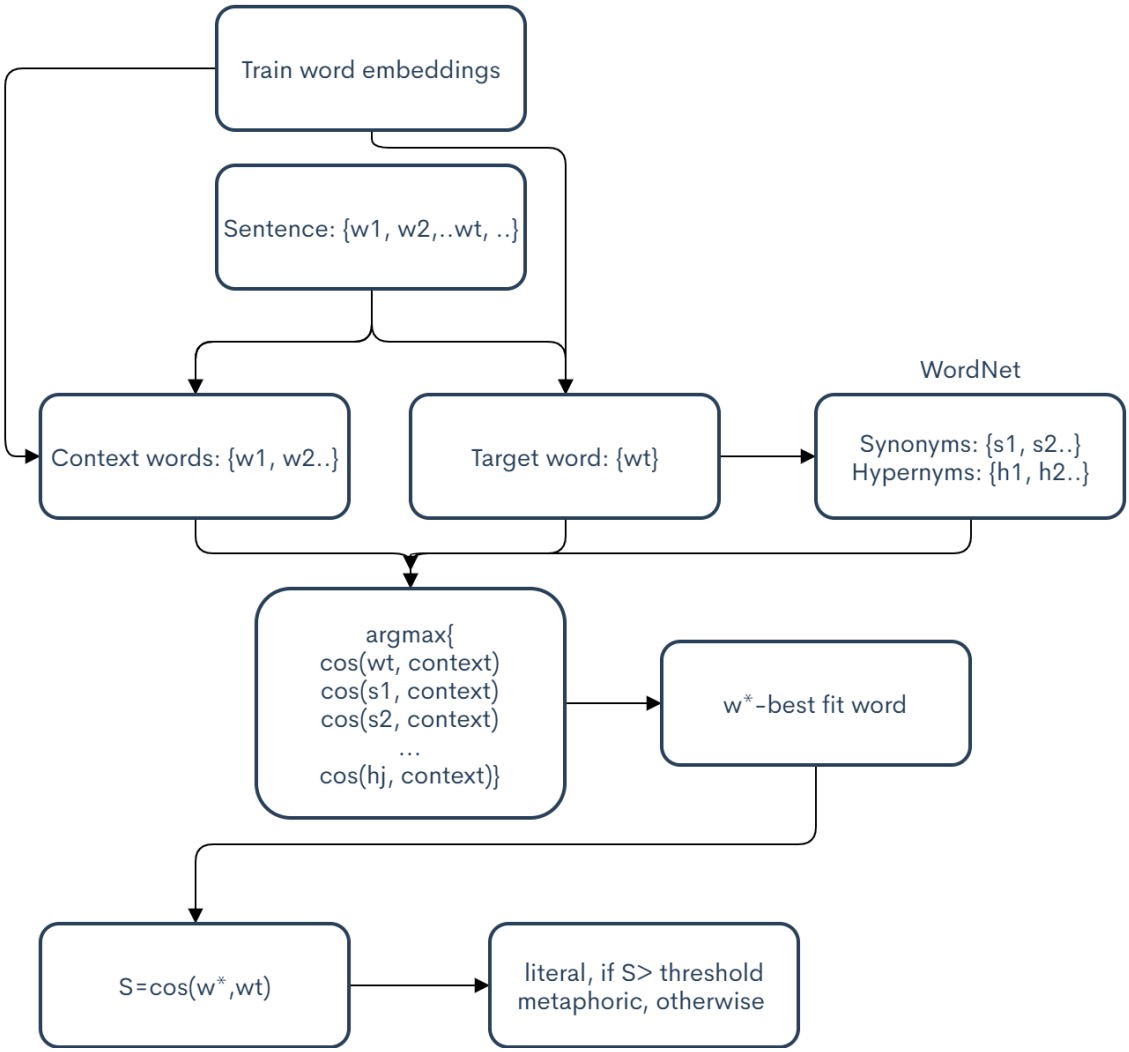

Figure 15. Metaphor identification framework using WordNet. w_t is target word and $w*$ is best fit word.

5.2.4 Extension using BERT embeddings

WordNet-based semi-supervised method of [33] is extended by substituting the non-contextualised word embeddings with contextualised embeddings from BERT.

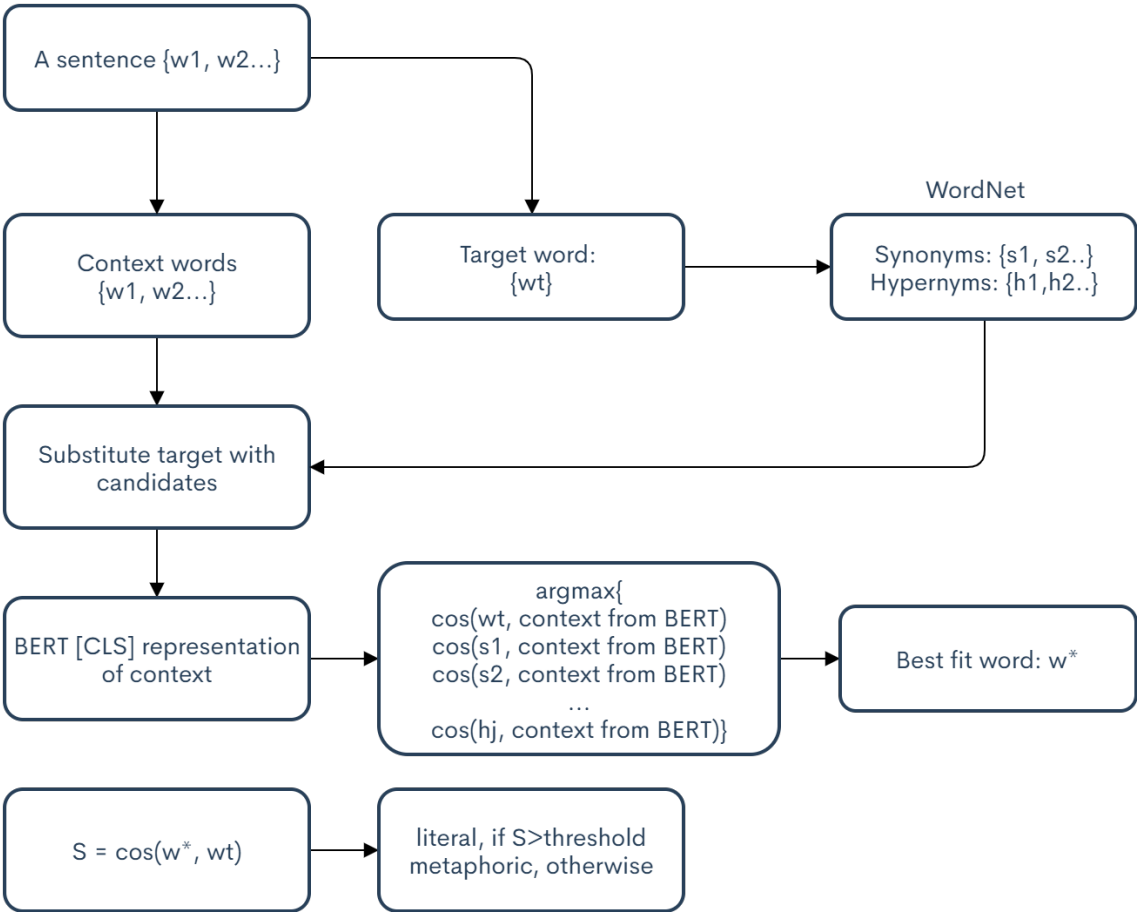

Figure 16. Metaphor identification framework using BERT and WordNet. $w*$ is a best fit word and w_t is a target word.

Figure 16 shows the implementation for WordNet-based method. First, the context vector was replaced by the embedding obtained from the BERT model. For this, the initial sentence was input to the BERT model. All the token embeddings except for the initial word embeddings were extracted from the model. Final context embedding was the summed token embeddings divided by the amount of token embeddings:

$$context\_embedding = \frac{1}{m} * \sum_{n=1}^{m} v_n \tag{22}$$

where m is the number subwords the whole sentence was split into. When all the

context embeddings were obtained, the similarity between the possible candidate word embedding and the actual word context embedding was measured.

Next, the similarity between the initial target word and the suitable word is measured. Word embeddings are extracted by inputting the whole sentence where the target word is replaced by the best-fit word or the target word itself. The token embeddings from this sentence are extracted to be used as the final word embeddings. If the word is tokenised into many subwords, all the embeddings representing the word are added together and then divided by the number of subwords:

$$word\_vector = \frac{1}{m} * \sum_{n=1}^{m} v_n \tag{23}$$

where m is the number of subwords the word is split into. Cosine similarity is used to measure the similarity between the best fit and target word. If the similarity is above the threshold, then the phrase is literal, otherwise metaphorical. The same seed set is used as in previous methods for determining the threshold. Again the threshold was found by maximizing the classification accuracy on this seed-set. The threshold for this approach was 0.6. If the calculated value for a specific phrase is below the threshold, the phrase is classified as metaphorical otherwise literal.

For example, when the sentence is *"Kuidas me üldse selleni jõudnud oleme , et meil on ligi sada elukutselist päästekomandot , mis neelavad palju raha ja pakuvad vähe päästet ?"* and the phrase is *"neelavad Raha"*, verb *"neelavad"* is first lemmatised into its base form *"neelama"*. Then the lookup to the WordNet is made, all the direct hypernyms and synonyms are extracted. To get the context vector, the initial sentence is input to the BERT model, and the token embeddings are extracted from the BERT's output. Then the candidate words are substituted to the sentence and the token embeddings for this word are extracted from the BERT model. The similarities based on the context embedding and the candidate word embedding is measured. The highest similarity yielding candidate word is the best-fit word. In this case, the highest similarity was achieved with word *absorbeeruvad*. This whole example is illustrated in Figure 17.

5.3 Evaluation Datasets

This subsection presents the model evaluation data sets for Estonian and English. Additionally this subsection will present the evaluation metrics used to judge the efficacy of the models.

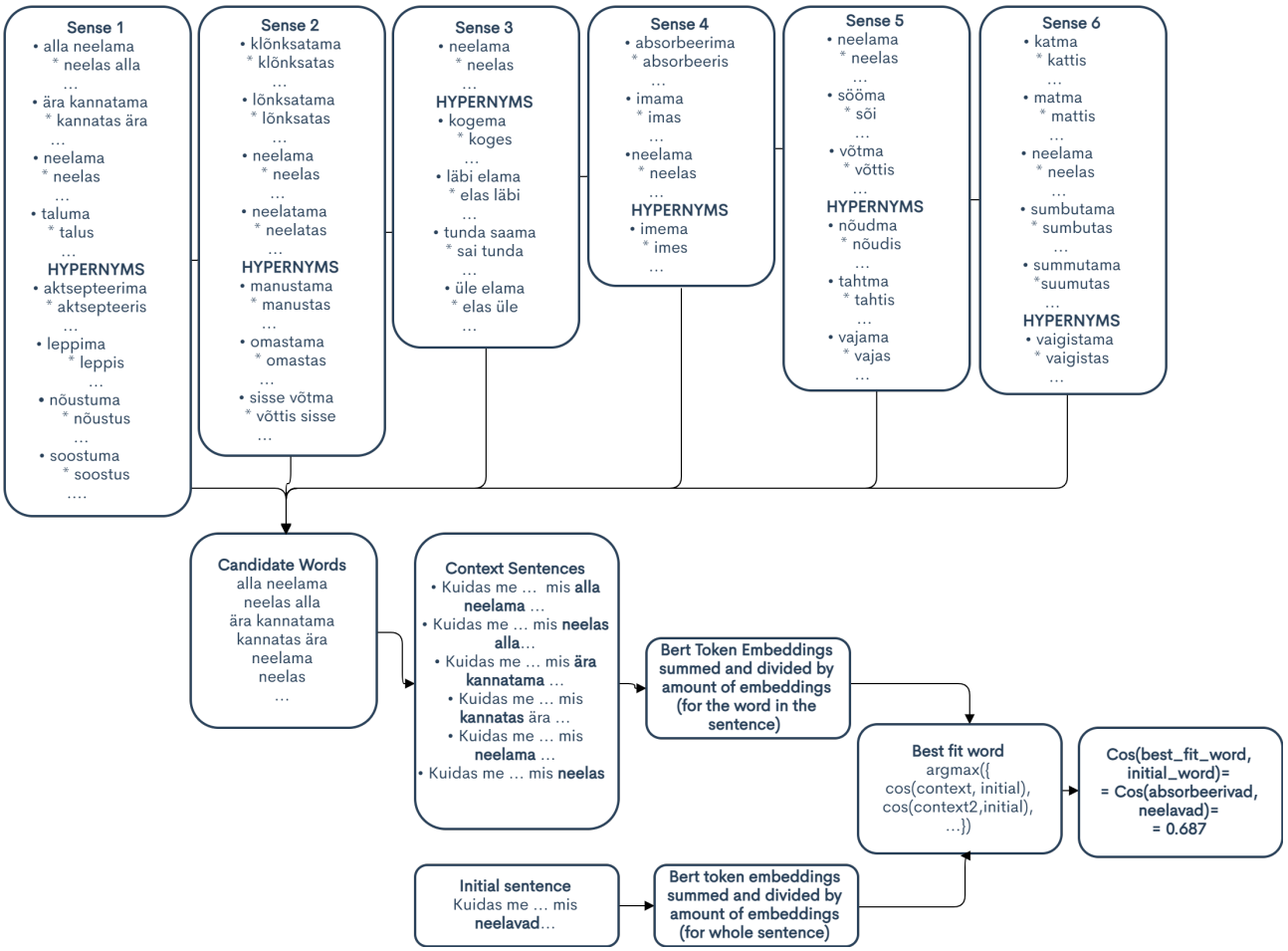

Figure 17. Example workflow with BERT embeddings.

5.3.1 Estonian dataset

A new evaluation dataset was created for the evaluation of the implemented metaphor identification methods.

As HGFC uses predefined sources and extracts only certain metaphorical expressions, the evaluation dataset contains phrases that the HGFC algorithm has information about. Because of that, the evaluation dataset was created using the phrases extracted from the HGFC models. All the sentences for the evaluation dataset were extracted from the Estonian National Corpus 2017 [19]. The author first roughly annotated extracted sentences to get about equal amounts of metaphorical and literal sentences. In total, 500 sentences were extracted.

Three human judges who are native speakers of Estonian were asked to evaluate the extracted sentences. The annotator's task was to classify the given phrase in the sentence as metaphorical or literal. They were also given a definition of a metaphor and examples of sentences containing noun-verb metaphors, where verb was used metaphorically and examples of literal sentences. Annotators were encouraged to rely on their intuition when annotating the phrases. The task description for the annotators is given in the Appendix.

Annotators annotated the dataset using Label Studio [55], which is an open-source data labelling tool. Label Studio supports data types like audio, text, images, videos and time series. The online version of the Label Studio app [11] was used for annotating. Figure 18 shows the interface that the annotators saw. The NOUN-VERB phrase in question was marked with red inside the sentence. Annotators had two options to choose from, either the phrase in this context is literal or metaphorical.

The inter-annotator agreement was measured in terms of Fleiss' kappa [48], which is a statistical method for accessing the reliability of agreement between annotators. Fleiss' kappa is used instead of Cohen kappa as it can be used to measure agreement between more than two annotators. Fleiss' kappa shows how much the annotators' annotations are better than random annotations.

Equation for the Fleiss' kappa:

$$\kappa = \frac{\overline{P} - \overline{P}_e}{1 - \overline{P}_e} \tag{24}$$

Here, the factor $1 - \overline{P}_e$ denotes the degree of agreement that is attainable above chance. $\overline{P} - \overline{P}_e$ is the actual degree of agreement above chance. If all the annotators

[11] `https://app.labelstud.io/`

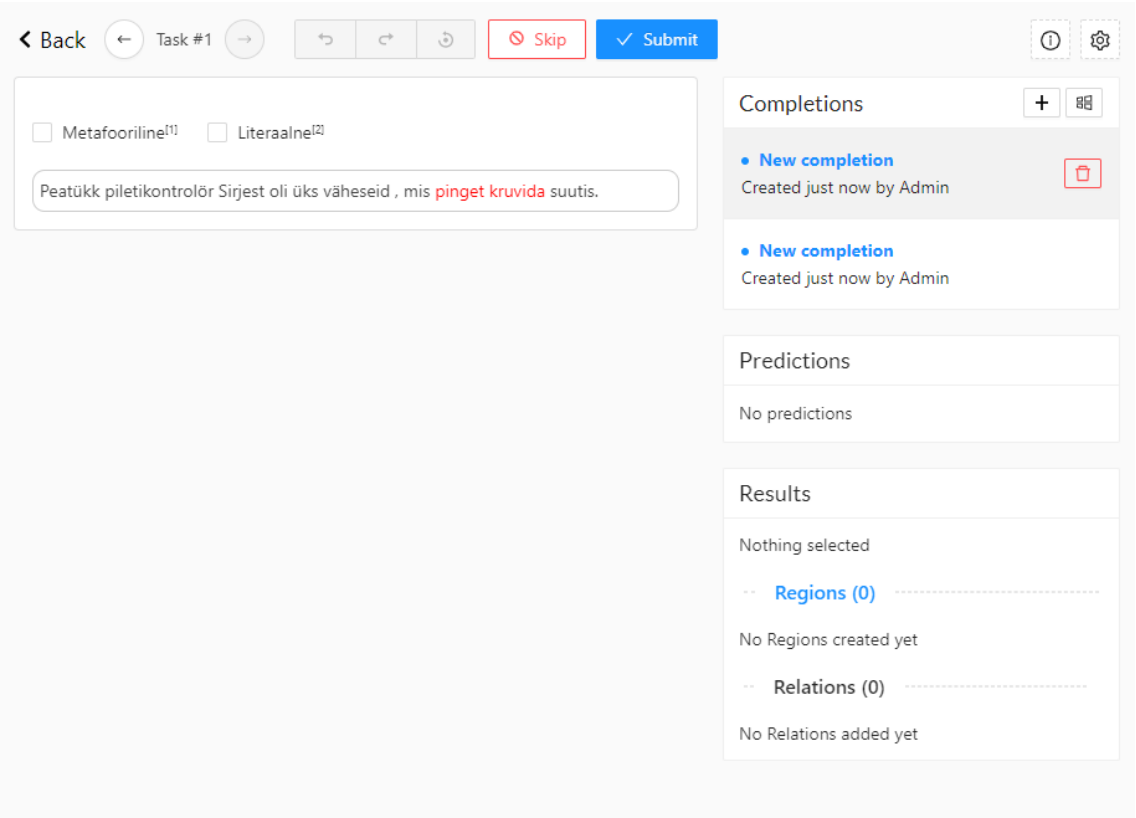

Figure 18. Label Studio interface

Kappa	Agreement
<0	Less than chance agreement
0.01-0.20	Slight agreement
0.21-0.40	Fair agreement
0.41-0.60	Moderate agreement
0.61-0.80	Substantial agreement
0.81-1.00	Almost perfect agreement

Table 12. Landis et al. [27] interpretations of κ values

are in complete agreement, then $\kappa = 1$ and if there is no agreement, then $\kappa \leq 0$.

Annotators agreement in this task was $\kappa = 0.78$ ($n = 3, N = 500, k = 2$). According to the Table 12 the agreement on the metaphor classification task falls into range 0.61 and 0.80, which is a substantial agreement.

The final dataset contained 262 sentences where the verb was used literally and 238 sentences where the verb was used metaphorically.

For example, the created dataset contained sentences like these:

- mõtted jooksevad mõte jooksma Selle filmi kohta on raske midagi mõistlikku kirja panna kuna mõtted jooksevad nagu kiirrong . M
- ravi kesta ravi kestma Samuti tasub arvestada , et Savisaare ravi võib kesta nädalaid või suisa kuid . L
- raha tiksus raha tiksuma Mintoses tiksus vähehaaval raha tagasi ja uusi laene sobivatel tingimustel läks üsna harva välja . M
- arve maksab arve maksma See on elementaarne viisakus , et mees maksab arve , kui ta on naise välja kutsunud . L

Every instance contains four items, all separated with a tab symbol. First is the phrase directly taken from the sentence, second is the lemmatised phrase, third is the sentence itself, and the last item is the label, where *M* denotes *Metaphorical*, and *L* denotes *Literal*.

5.3.2 English dataset

A subset of VU Amsterdam Metaphor corpus [51] is used for the evaluation of metaphor identification methods for English. VU Amsterdam Metaphor Corpus is the most extensive available corpus containing randomly selected texts from four

Type	No of lexical units	Fragments
News	45116	63
Fiction	44892	12
Conversations	48001	24
Academic	49561	16
Total	187570	115

Table 13. Statistics of the VU Amsterdam Metaphor corpus.

registers of the BNC-Baby, which is a subset of the BNC corpus [5]. This corpus contains academic texts, conversations, fiction and news texts. Table 13 shows the statistics of the VU Amsterdam Metaphor Corpus.

The corpus is hand-annotated for all metaphorical language use using the Metaphor Identification Procedure VU University Amsterdam (MIPVU) procedure, which is an expanded version of Metaphor Identification Procedure (MIP) [13]. MIP has a total of four steps:

1. Read the text for the general understanding of it.
2. Determine all the lexical units from the text.
3. Find the contextualised and basic meaning of all the units in the text.
4. If there is a contrast between the contextual and basic meanings, then the unit is metaphorical.

The MIPVU metaphor identification protocol is a systematic and transparent way of identifying linguistic but not conceptual metaphors. This protocol also identifies units that could be realised as a metaphor - potential metaphor.

Figure 19 shows all the annotations used in the VU Amsterdam Metaphor corpus alongside examples from the corpus.

Five hundred sentences were extracted using the extracted 2000 most frequent nouns and grammatical features from the HGFC method from the corpus. From the sentences, 250 contained NOUN-VERB phrase where the verb was used metaphorically, and 250 were used literally.

The created corpus contained sentences:

- grandmother promised grandmother promise l But you promised your grandmother not to learn to fly , Adam objected .

relation to metaphor	metaphor type	XML representation	corpus examples
metaphor	indirect	<mrw type="met">valuable</mrw>	Professional religious education teachers like Marjorie B Clark (Points of View, today) are doing *valuable* work in many secondary schools (…). (K58-fragment01)
	direct	<mrw type="lit">ferret</mrw>	(…) he's like a *ferret*. (KBD-fragment21)
	implicit	<mrw type="impl">it</mrw>	Naturally, to embark on such a *step* is not necessarily to succeed in realizing *it*. (A9J-fragment01)
WIDLII		<mrw type="met" status="WIDLII">up</mrw>	driven *up* the bumpy Forest Drive to East Kielder Farm, (…). (AHC-fragment60)
PP		<mrw type="met" status="PP">decide</mrw>	A party can't even *decide* its name (…). (A7W-fragment22)
UNCERTAIN		<mrw type="met" status="UNCERTAIN">appealed</mrw>	The council appealed by cases stated. (A7Y-fragment03)
signal		<mFlag type="lex">as if</mFlag>	It is *as if* it is walking through a minefield. (A9J-fragment01)
		mFlag type="morph">like</mFlag>	The wave-*like* pattern of the Intifada. (A9J-fragment01)
		<mFlag type="phrase" id="a9j-fragment01-mfp1">in</mFlag></w>(…)<mrw type="met">role</mrw>(…)<mFlag type="phrase" corresp=a9j-fragment01-mfp1">of</mFlag>	(…) acts *in* the *role of* field general (A9J-fragment01)

Figure 19. Overview of all annotations used in VU Amsterdam Metaphor Corpus. Image taken from Krennmayr et al. [21].

- election win election win l He told a Labour Co-ordinating Committee rally that it was not enough for Labour to win the next election .
- wishes reflect wish reflect m A high level of mobility in the structure allows for continuous interchange of roles and ideas , and retains for the Unified National Command the closeness to grass roots that is required to make it truly reflect people 's wishes and sentiments .
- risk carry risk carry m Another question is whether the criminal law ought not to be wider in its application to activities which carry some risk of causing death than in other spheres .

Each instance in the corpus has four items, all separated with a tab symbol. First is the phrase directly taken from the text, second is the lemmatised phrase, third is the label, where *m* denotes *Metaphorical* and *l* denotes *Literal* and fourth is the sentence itself.

5.3.3 Metrics

Traditional metrics were used to evaluate the metaphor identification task, as the metaphor identification task is essentially a classification task. Those metrics are accuracy, precision, recall and F1-score.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{25}$$

$$Precision = \frac{TP}{TP + FP} \tag{26}$$

$$Recall = \frac{TP}{TP + FN} \tag{27}$$

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{28}$$

where TP, TN, FP and FN denote the amount of true positives, true negatives, false positives and false negatives respectively.

The most important metric is precision, which shows the quality of the annotations from the implemented methods.

6 Results

This chapter presents the results from all of the implemented metaphor identification methods for Estonian and English. It should be mentioned that it is impossible to directly compare results obtained from Estonian and English evaluation datasets. This is because they contain different sentences (and in different languages) and therefore are not strictly interchangeable.

6.1 Results for Clustering-Based Approach

Table 14 shows the HGFC based metaphor identification systems' results on the Estonian and English data. Regular HGFC using grammatical features yielded better results on both languages compared with BERT based HGFC. Precision is lower than recall for all models, which shows that the methods classified phrases as metaphorical too sparingly.

Comparing the results between languages, it can be seen that both HGFC methods performed better on the English dataset.

	Estonian			English		
Model	**Recall**	**Precision**	**F1-score**	**Recall**	**Precision**	**F1-score**
HGFC	0.624	**0.544**	0.581	0.764	**0.672**	0.715
BERT HGFC	0.596	0.532	0.562	0.710	0. 608	0.655

Table 14. Results from the HGFC method. Best precision score shown in bold.

6.2 Results for Word Embedding and Visual Features Based Approach

Table 15 shows the best results obtained from embeddings and visual features based approach. All the obtained results can be seen from the Appendix. As it can be seen, the best method combines WordCos BERT embeddings and visual features (BERT-WC-ML) with middle fusion. From the Skip-gram based models, the best results for English were achieved with only using Skip-gram embeddings and classifying the phrases with WordCos method (Sg-WC). For Estonian language, the best combination was using Skip-gram embeddings and classifying the phrases with PhrasCos2 method (Sg-PC2).

From the overall results, it can be said that images alone are not meaningful enough for the task of metaphor identification.

Model	Estonian			English		
	Recall	**Precision**	**F1-score**	**Recall**	**Precision**	**F1-score**
Sg-WC	0.528	0.597	0.561	0.548	0.665	0.601
Sg-PC2	0.727	0.621	0.67	0.572	0.459	0.509
BERT-WC-ML	0.508	**0.645**	0.568	0.604	**0.719**	0.657

Table 15. Best results from distributional models using image embeddings for Estonian and English. Best precision marked in bold.

6.3 Results for Word Embedding and WordNet Based Approach

Table 16 shows the performance of the implemented models using WordNet. BERT base model outperforms the Word2vec based models for both languages. BERT gave the highest overall precision score of 0.65 for Estonian and 0.72 for the English dataset. As was found in [33], the models based on input and output vectors ($CBOW_{I+O}, SG_{I+O}$) achieved better results compared with the models that only used input vectors ($CBOW_I, SG_I$). It supports the [33] assumption that using both input and output vectors is beneficial for modelling similarity between words that have a different part-of-speech tag—comparing the Skip-gram and CBOW models; it can be seen that CBOW yields better results on both languages. It makes sense as CBOW averages the context word input vectors to maximise the probability for each context.

Model	Estonian			English		
	Recall	**Precision**	**F1-score**	**Recall**	**Precision**	**F1-score**
$BERT$	0.728	**0.65**	0.687	0.788	**0.721**	0.753
$CBOW_I$	0.716	0.565	0.631	0.812	0.611	0.698
$CBOW_{I+O}$	0.728	0.579	0.645	0.768	0.658	0.708
SG_I	0.664	0.548	0.600	0.672	0.654	0.663
SG_{I+O}	0.672	0.571	0.618	0.66	0.682	0.671

Table 16. Metaphor identification results. Best precision is marked in bold.

6.4 Discussion

The overall best method for Estonian and English language was the WordNet-based method using BERT contextualised embeddings. Even though the Estonian WordNet for verbs is small, it could still give more helpful information than the

visual embeddings and grammatical features in HGFC. It could be explained that WordNet is a hand-created knowledge source, which thus contains suitable information for the metaphor identification task.

The clustering method HGFC has many limitations compared with other implemented methods. First, it is constrained with pre-defined source concepts set. All the source-target domain mappings have to be done earlier, and then fixed metaphorical expressions could be extracted. This method will always identify the same metaphorical expressions no matter the context the phrase is in. It is a significant limitation compared with other methods implemented that do not have this restriction. In real-world settings, HGFC would not be comparable with other methods because of it. HGFC needs significant amounts of pre-processing at the beginning as well. Other methods could be initialized with pre-trained embeddings. Using BERT would also not alleviate this problem as the seed set of 2000 most frequent nouns and the grammatical features are still necessary. On the plus side - recognizing metaphors is much faster after all the pre-processing has been done. This is because it only needs to parse the sentence to find the correct syntactic constructions and then check if the phrase is included in the pre-defined metaphorical associations list. Other methods need to load a pre-trained model like Word2vec or BERT and then do many different similarity computations and checkups to different knowledge sources.

Using BERT contextualized embeddings were beneficial in the WordNet model and the model based on visual images. This could be explained by the fact that the BERT model - through masked language modelling and next sentence prediction - has learned the context and meaning of the words better than Word2vec embeddings. Using BERT embeddings with HGFC did not yield any better results. Only using phrases to construct the similarity matrix for HGFC was probably not enough to enhance the performance. The similarity space produced by BERT embeddings was an under-differentiated space with no significant clustering - this meant that any clusters generated were more likely noise. Selectional preference filtering alleviated this problem to some extent but not enough to enhance the model's performance.

Combining visual and linguistic features did not add any more meaningful information compared with the WordNet-based method. The issue could come down to the relevancy of the downloaded images. Even though the Estonian phrases and words were first translated into English, the translations contained weird constructions, which also led to images that did not represent the phrase or word correctly. Not translating lead to even worse performance as the images using Estonian keywords rarely contained the correct concept. As an example of this - the word *tee (tea - eng)* produced images of T-shirts as in English *tee* is often used instead of *T-shirt*. This

was a significant issue when working with using the features from images. It could be alleviated by using a better source for downloading images - for example Yandex Images[12]. The issue could have also been in the pre-trained model from which the features were extracted. In recent years many new architectures have been proposed for image classification that has yielded better results than ResNet-18 has.

Most errors came from the metaphorical phrases that were very common in language, like *leiba teenima*, *mägesid liigutama*. These kinds of metaphors cannot be detected using selectional preferences or distributional models. Distributional models have already learned that the word often occurs in those contexts so that the best appropriate word would be the same metaphorical word. This was mostly seen in the Word2vec based methods. For example, *pinget kruvima* is commonly used in Estonian; the best fit verb in this phrase was also *kruvima*, making the whole phrase literal even though it is metaphorical.

It could be said that Word2vec based methods' thresholds were more stable when comparing the computed thresholds between different models. Using different thresholds achieved consistent results for Word2vec whereas doing so in BERT-based produced worse performance. This could be explained by the fact that BERT-based methods produced similarities inside a small range. This means, that going below some threshold would mean that all phrases would be classified as literal because BERT-based methods considered all phrases somewhat similar.

All results from the implemented methods show that these methods are portable across languages. All results for the English language were higher than for Estonian which could be due to the differences in the evaluation dataset. The English dataset was extracted from the VU Amsterdam Metaphor Corpus [51], which is annotated by more than three persons and it could be more reliable than the dataset for Estonian produced for this thesis.

To conclude, BERT contextualized embeddings with WordNet information performed the best for metaphor identification in the Estonian language.

[12] `https://yandex.com/images/`

7 Conclusion

The goal of this thesis was to implement unsupervised or semi-supervised methods for metaphor identification for the Estonian language. As there were no previous attempts to solve this problem, this work creates a baseline for future works in this area. All methods were also implemented using English text data. Doing this helps to understand if some problems are language-specific.

In total, three types of methods were implemented. All methods were trained or using the Estonian National Corpus 2017 [19]. This thesis had a secondary goal of investigating the use of contextualised embeddings from the BERT language model. All three models were extended to be used with BERT embeddings instead of grammatical features or Word2vec non-contextualised embeddings.

All implemented methods were evaluated in the same dataset created in this thesis. The Estonian evaluation dataset contained 238 sentences where verb was used metaphorically and 262 literal sentences. The English evaluation dataset was extracted from the VU Amsterdam Metaphor Corpus. The final dataset contained 250 metaphorical and 250 literal sentences.

The WordNet-based method using BERT embeddings achieved the best results on Estonian and English datasets from the implemented models. As all other identification methods can also successfully identify metaphors - it could be said that the primary goal of the thesis was achieved. The secondary goal was also achieved, as models enriched with BERT's contextualised embeddings demonstrated improved results over the baseline methods.

This thesis made many contributions. First, a new evaluation set was created. This evaluation set contains 500 sentences for identifying verb-noun phrases where the verb was used metaphorically. This set could be used for future works to compare other methods. Second - a baseline 'best performing model' was established for identifying metaphors in the Estonian language. As a final contribution - this thesis demonstrated that data extracted from pre-trained BERT models can help existing models in solving the metaphor identification task.

This work focused on identifying noun-verb pairs. In the future these models should be extended to identify other metaphorical expressions in other syntactic constructions like adjective-noun expressions. A more significant step would be to create an annotated dataset for metaphorical expressions for the Estonian language. With such a dataset it would possible to experiment with supervised learning. With a relevant dataset - a possible solution would be to fine-tune BERT models for the task.

Multilingual BERT models have also been shown to produce good results in different

NLP tasks [20]. It would be interesting to see if multilingual learning could be beneficial in the task of metaphor identification as many metaphors are common in different languages.

8 Acknowledgements

First, I want to thank my supervisor Eduard Barbu for having faith in me and and supporting me throughout the thesis. I would also like to thank Al William Tammsaar for proofreading the thesis. Special thanks goes to the anonymous annotators for classifying sentences as metaphorical and literal. Finally, I would also like to thank my dog, pawsitivity specialist and personal human-walker - Tommi for keeping my spirits up.

References

[1] Eleri Aedmaa, Maximilian Köper, and Sabine Schulte im Walde. Combining abstractness and language-specific theoretical indicators for detecting non-literal usage of Estonian particle verbs. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 9–16, New Orleans, Louisiana, USA, June 2018. Association for Computational Linguistics.

[2] Lawrence W Barsalou. Grounded cognition. *Annu. Rev. Psychol.*, 59:617–645, 2008.

[3] J. R. Binder, C. F. Westbury, K. A. McKiernan, E. T. Possing, and D. A. Medler. Distinct brain systems for processing concrete and abstract concepts. *J. Cognitive Neuroscience*, 17(6):905–917, June 2005.

[4] Max Black. More about metaphor. *Dialectica*, pages 431–457, 1977.

[5] Lou Burnard. *Reference Guide for the British National Corpus (XML) version.* 2007.

[6] Sebastian Crutch and Elizabeth Warrington. Abstract and concrete concepts have structurally different representational frameworks. *Brain : a journal of neurology*, 128:615–27, 04 2005.

[7] Oier Lopez de Lacalle, Ander Salaberria, Aitor Soroa, Gorka Azkune, and Eneko Agirre. Evaluating multimodal representations on visual semantic textual similarity, 2020.

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[9] Dan Fass. met*: A method for discriminating metonymy and metaphor by computer. *Computational linguistics*, 17(1):49–90, 1991.

[10] J. R. Firth. A synopsis of linguistic theory 1930-55. 1952-59:1–32, 1957.

[11] Dedre Gentner. Structure-mapping: A theoretical framework for analogy. *Cognitive science*, 7(2):155–170, 1983.

[12] A. Goatly. *The Language of Metaphors.* English language, literature, psychology. Routledge, 2011.

[13] Pragglejaz Group. Mip: A method for identifying metaphorically used words in discourse. *Metaphor and Symbol*, 22:1–39, 10 2007.

[14] E. Dario Gutiérrez, Ekaterina Shutova, Tyler Marghetis, and Benjamin Bergen. Literal and metaphorical senses in compositional distributional semantic models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 183–193, Berlin, Germany, August 2016. Association for Computational Linguistics.

[15] Zellig S Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

[16] Reyhaneh Hashempour and Aline Villavicencio. Leveraging contextual embeddings and idiom principle for detecting idiomaticity in potentially idiomatic expressions. In *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, pages 72–80, Online, December 2020. Association for Computational Linguistics.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[18] Leslie Hogben, editor. *Handbook of Linear Algebra*. CRC Press, Boca Raton, FL, USA, 2006.

[19] J. Kallas and K. Koppel. Estonian National Corpus 2017, 2018. Center of Estonian Language Resources.

[20] Claudia Kittask, Kirill Milintsevich, and Kairit Sirts. Evaluating multilingual bert for estonian, 2021.

[21] T. Krennmayr and G.J. Steen. *VU Amsterdam Metaphor Corpus*, pages 1053–1071. Springer Verlag, 2017.

[22] Saisuresh Krishnakumaran and Xiaojin Zhu. Hunting elusive metaphors using lexical resources. In *Proceedings of the Workshop on Computational Approaches to Figurative Language*, pages 13–20, Rochester, New York, April 2007. Association for Computational Linguistics.

[23] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[24] Murathan Kurfalı and Robert Östling. Disambiguation of potentially idiomatic expressions with contextual embeddings. In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 85–94, online, December 2020. Association for Computational Linguistics.

[25] G. Lakoff. *Master Metaphor List.* University of California, 1994.

[26] George Lakoff and Mark Johnson. *Metaphors we live by.* University of Chicago press, 2008.

[27] J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174, 1977.

[28] Sven Laur, Siim Orasmaa, Dage Särg, and Paul Tammo. Estnltk 1.6: Remastered estonian nlp pipeline. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 7154–7162, Marseille, France, May 2020. European Language Resources Association.

[29] Chee Wee Leong and Rada Mihalcea. Measuring the semantic relatedness between words and images. 01 2011.

[30] Samuel R. Levin. Aristotle's theory of metaphor. *Philosophy Rhetoric*, 15(1):24–46, 1982.

[31] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

[32] Edward Loper and Steven Bird. Nltk: The natural language toolkit. *CoRR*, cs.CL/0205028, 2002.

[33] Rui Mao, Chenghua Lin, and Frank Guerin. Word embedding and WordNet based metaphor identification and interpretation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1222–1231, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[34] Zachary J. Mason. Cormet: A computational, corpus-based conventional metaphor extraction system. *Comput. Linguist.*, 30(1):23–44, March 2004.

[35] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[36] George Miller. *WordNet: An electronic lexical database.* MIT press, 1998.

[37] Heili Orav, Kadri Vare, Sirli Parm, Liisi Pool, Lauri Eesmaa, Piia Taremaa, Maria Reile, Katrin Alekand, Ingmar Jaska, Helen Türk, Eleri Aedmaa, and Ahti Lohk. Estonian wordnet (2.1), 2015.

[38] Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online, July 2020. Association for Computational Linguistics.

[39] S. Pernes. Metaphor mining in historical german novels: An unsupervised learning approach. In *2015 IEEE International Conference on Big Data (Big Data)*, pages 1650–1652, 2015.

[40] Wim Peters and Ivonne Peters. Lexicalised systematic polysemy in WordNet. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*, Athens, Greece, May 2000. European Language Resources Association (ELRA).

[41] Malay Pramanick and Pabitra Mitra. Unsupervised detection of metaphorical adjective-noun pairs. In *Proceedings of the Workshop on Figurative Language Processing*, pages 76–80, 2018.

[42] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages, 2020.

[43] P. Resnik. Selection and information: a class-based approach to lexical relationships. 1993.

[44] Ekaterina Shutova, Douwe Kiela, and Jean Maillard. Black holes and white rabbits: Metaphor identification with visual features. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 160–170, San Diego, California, June 2016. Association for Computational Linguistics.

[45] Ekaterina Shutova and Lin Sun. Unsupervised metaphor identification using hierarchical graph factorization clustering. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 978–988, 2013.

[46] Ekaterina Shutova, Lin Sun, and Anna Korhonen. Metaphor identification using verb and noun clustering. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1002–1010, 2010.

[47] Ekaterina Shutova and Simone Teufel. Metaphor corpus annotated for source - target domain mappings. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May 2010. European Language Resources Association (ELRA).

[48] Sidney Siegel and N John Castellan Jr. Nonparametric statistics for the behavioral sciences. 1988.

[49] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[50] G. Steen. Finding metaphor in discourse: Pragglejaz and beyond. *Cultura, Lenguaje y Representación*, 5:9–25, 2007.

[51] Gerard Steen. *A method for linguistic metaphor identification: From MIP to MIPVU*, volume 14. John Benjamins Publishing, 2010.

[52] Chang Su, Shuman Huang, and Yijiang Chen. Automatic detection and interpretation of nominal metaphor based on the theory of meaning. *Neurocomputing*, 219(September 2016):300–311, 2017.

[53] Lin Sun and Anna Korhonen. Improving verb clustering with automatically acquired selectional preferences. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 638–647, Singapore, August 2009. Association for Computational Linguistics.

[54] Hasan Tanvir, Claudia Kittask, and Kairit Sirts. Estbert: A pretrained language-specific bert for estonian, 2020.

[55] Maxim Tkachenko, Mikhail Malyuk, Nikita Shevchenko, Andrey Holmanyuk, and Nikolai Liubimov. Label Studio: Data labeling software, 2020-2021. Open source software available from https://github.com/heartexlabs/label-studio.

[56] Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. Metaphor detection with cross-lingual model transfer. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 248–258, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[57] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[58] T. Veale, E. Shutova, and B. Klebanov. *Metaphor: A Computational Perspective*. Online access: Morgan & Claypool Synthesis Collection Six. Morgan & Claypool, 2016.

[59] Tim vor der Brück and Marc Pouly. Text similarity estimation based on word embeddings and matrix norms for targeted marketing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1827–1836, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[60] Yorick Wilks, Lucian Galescu, James Allen, and Adam Dalton. Automatic Metaphor Detection using Large-Scale Lexical Resources and Conventional Metaphor Extraction. *First Workshop on Metaphor in NLP*, (June):36–44, 2013.

[61] Mike Wu and Noah Goodman. Multimodal generative models for compositional representation learning, 2019.

[62] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation, 2016.

[63] S. Yu, Kai Yu, and Volker Tresp. Soft clustering on graphs. In *NIPS*, 2005.

[64] Asta õim. *Komistusi metafooridega*. Keelehooldekeskus, 2011.

Appendix

I. Instructions for Annotators

This section is presenting the instructions, that were given to the annotators who classified the NOUN-VERB phrases as metaphorical or literal.

Metafooride Annoteerimine

Metafoor on defineeritud kui sõna või väljendi kasutamine sarnasuse alusel uudses, ülekantud tähenduses.

Näiteid fraasidest, mis on metafoorilised: kirgi kütma - Võhmas kütab kirgi kavandatav laululava. ajudele hakkama - See subtroopiline suvi hakkas ajudele hammas hakkama - Linnamaja peale minu hammas ei hakka. ämbrisse astuma - Mõnigi poliitik on korduvalt ämbrisse astunud

Näiteid fraasidest, mis on literaalsed: teksti kirjutama - Teksti kirjutades mõtlesin ma palju ka Hollandi erakordselt intensiivsele kunstielule 17. sajandil mees kõndima - Mees kõndis päev otsa stuudios ringi , kordagi maha istumata . sõitu võitma - Ka päeva teise sõidu võitis Asmer piiri ületama - " Ärge kahelge piiri ületada proovivate inimeste suhtes kasutada tulirelva , isegi juhul kui põgenikeks on naised või lapsed , " seisab Stasi dokumendis , mis kannab kuupäeva 1. oktoober 1973 .

Teie ülesanne on järgnev: 1. Loe lause mõttega üle ja proovi aru saada, mida see öelda tahab.

2. Vaata punasega märgitud fraasi, määra kindlaks, kas see fraas oleks muus kontekstis erineva tähendusega. Nt lauses "Tooraineturu mull lõhkeb" fraas "mull lõhkeb" on metafoorline aga lauses "Lapse puhutud mull lõhkes" literaalne.

3. Otsusta, kas fraasis leiduv tegusõna on lause kontekstis literaalne või metafoorile (näited metafoorsetest ja literaalsetest fraasidest on toodud üleval)

4. Kui kahtled fraasi metafoorilisuses, klassifitseeri fraas kui literaalne.

II. Extracted Source Domains

1. TEEKOND - JOURNEY
2. LIIKUMINE - MOTION
3. VIGASTUS - INJURY
4. ASUKOHT - LOCATION
5. VÕISTLUS - RACE
6. TULI - FIRE
7. LAPS - CHILD
8. KIIRUS - SPEED
9. SÕDA - WAR
10. HAIGUS - DISEASE
11. LAHING - FIGHT
12. EHITUS - CONSTRUCTION
13. SÕIDUK - VEHICLE
14. SÜSTEEM - SYSTEM
15. ÄRI - BUSINESS
16. MASIN - MACHINE
17. ISIK - PERSON
18. KONTROLL - CONTROL
19. TEE - PATH
20. LUGU - STORY
21. VEDELIK - LIQUID
22. KASV - GROWTH
23. TAKISTUS - OBSTACLE
24. KAUGUS - DISTANCE
25. VÄRAV - GATEWAY
26. VAENLANE - ENEMY

27. KONKURENTS - COMPETITION
28. SUUND - DIRECTIONALITY
29. OBJEKT - OBJECT
30. NÄGEMUS - VISION
31. SURM - DEATH
32. ELU - LIFE
33. RAHA - MONEY
34. KEHA - BODY
35. TOIT - FOOD
36. TAIM - PLANT
37. HOONE - BUILDING
38. RELV - WEAPON
39. OMAND - POSESSION
40. VALGUS - LIGHT
41. ISU - HUNGER
42. LOOM - ANIMAL
43. HÄVITAMINE - DESTRUCTION
44. KUJU - SHAPE
45. RIIE - CLOTHES
46. RESSURSS - RESOURCE
47. MÄNG - GAME
48. KINGITUS - GIFT
49. KAOTUS - LOSS
50. OMAND - POSSESSION

III. Results from Word Embedding and Visual Features Based Methods

	Estonian			English		
Method	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
L-WC	0.606	0.624	0.589	0.496	0.732	0.591
L-PC1	0.484	0.788	0.599	0.5	0.884	0.639
L-PC2	0.463	0.784	0.582	0.487	0.892	0.63
L-PC3	0.582	0.624	0.602	0.51	0.836	0.633
V-WC	0.522	0.484	0.565	0.496	0.796	0.608
V-PC1	0.523	0.544	0.533	0.492	0.524	0.508
V-PC2	0.622	0.736	0.623	0.501	0.728	0.594
V-PC3	0.546	0.656	0.596	0.492	0.844	0.622
WC-M	0.645	0.508	0.568	0.719	0.604	0.657
PC1-M	0.473	0.532	0.501	0.505	0.6	0.548
PC2-M	0.45	0.52	0.482	0.494	0.884	0.634
PC3-M	0.602	0.624	0.613	0.51	0.692	0.587
WC-L	0.634	0.604	0.619	0.502	0.84	0.629
PC1-L	0.517	0.468	0.492	0.478	0.768	0.589
PC2-L	0.517	0.664	0.581	0.487	0.76	0.594
PC3-L	0.581	0.664	0.587	0.505	0.796	0.618

Table 17. All metaphor identification results using BERT contextualised embeddings. L (in the beginning) -linguistic embeddings, V - visual embeddings, WC - WordCos, PC - PhrasCos, M - middle fusion, L - late fusion

Method	Estonian			English		
	Precision	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**
L-WC	0.597	0.528	0.561	0.665	0.548	0.601
L-PC1	0.547	0.672	0.603	0.514	0.72	0.6
L-PC2	0.621	0.727	0.67	0.459	0.572	0.509
L-PC3	0.506	0.66	0.573	0.47	0.376	0.427
WC-M	0.472	0.779	0.588	0.496	0.668	0.569
PC1-M	0.546	0.701	0.614	0.475	0.724	0.574
PC2-M	0.571	0.761	0.653	0.474	0.712	0.569
PC3-M	0.438	0.567	0.494	0.479	0.744	0.581
WC-L	0.472	0.779	0.587	0.495	0.668	0.569
PC1-L	0.503	0.686	0.58	0.501	0.724	0.592
PC2-L	0.512	0.727	0.601	0.509	0.624	0.561
PC3-L	0.453	0.507	0.478	0.463	0.584	0.517

Table 18. All metaphor identification results using Word2vec contextualised embeddings. L (in the beginning) -linguistic embeddings, WC - WordCos, PC - PhrasCos, M - middle fusion, L - late fusion

IV. Code

Code used in this thesis and created evaluation dataset for Estonian can be accessed from public GitHub repository[13].

[13] https://github.com/ckittask/metaphor_identification_for_estonian

V. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Claudia Kittask**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Metaphor Identification for Estonian,

   supervised by Eduard Barbu, PhD.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Claudia Kittask
04/08/2021