Blending Words or: How I Learned to Stop Worrying and Love the Blendguage

A computational study of lexical blending in Swedish.

Adam Ek

Department of Linguistics
Examensarbete 15HP
Datorlingvistik - Magisterkurs 15HP
Vårterminen
Handledare: Mats Wirén, Robert Östling
English title: Blending words or: How I Learned to Stop Worrying and Love the Blendguage

Stockholms universitet

Blending Words or: How I Learned to Stop Worrying and Love the Blendguage

A computational study of lexical blending in Swedish.

Abstract

This thesis investigates Swedish lexical blends. A lexical blend is defined as the concatenation of two words, where at least one word has been reduced. Lexical blends are approached from two perspectives. First, the thesis investigates lexical blends as they appear in the Swedish language. It is found that there is a significant statistical relationship between the two source words in terms of orthographic, phonemic and syllabic length and frequency in a reference corpus. Furthermore, some uncommon lexical blends created from pronouns and interjections are described. A description of lexical blends through semantic construction and similarity to other word formation processes are also described. Secondly, the thesis develops a model which predicts source words of lexical blends. To predict the source words a logistic regression model is used. The evaluation shows that using a ranking approach, the correct source words are the highest ranking word pair in 32.2% of the cases. In the top 10 ranking word pairs, the correct word pair is found in 60.6% of the cases. The results are lower than in previous studies, but the number of blends used is also smaller. It is shown that lexical blends which overlap are easier to predict than lexical blends which do not overlap. Using feature ablation, it is shown that semantic and frequency related features have the most importance for the prediction of source words.

Nyckelord/Keywords

Lexical blends, regression, word formation, feature ablation

Sammanfattning

Denna uppsats undersöker svenska teleskopord. Teleskopord definieras som sammansättningen av två eller fler ord, där minst ett av orden har reducerats. Uppsatsen undersöker teleskopord från två olika aspekter. Från den första aspekten så undersöks de statistiska egenskaperna hos teleskopord. Vi finner att det finns ett signifikant statistiskt samband mellan de två källorden vad gäller ortografisk, fonetisk och stavelse längd samt frekvens i referenskorpus. Vidare beskrivs några ovanliga teleskopord som har skapats från pronomen och interjektioner. Teleskopord undersöks även i termer av hur de konstrueras semantiskt, samt likheter med andra ordbildningsprocesser. Från den andra aspekten så har en logistisk regressions modell skapats vilket predicerar källorden hos teleskopord. Utvärderingen av modellen visar att för det högst rankade exemplen finner vi det korrekta ordparet i 32.2% av fallen. För de 10 högst rankade exemplen finner vi det korrekta ordparet i 60.6% av fallen. Modellens resultat är lägre än i tidigare studier, men mängden av data är även mindre än i tidigare studier. Vi visar även att teleskopord som överlappar är enklare att predicera är icke överlappande teleskopord. Vidare utvärdering visar att de särdrag relaterade till ords semantik och frekvens har högst påverkan på modellen.

Nyckelord/Keywords

Teleskopord, regression, ordbildning, särdragsablation

Table of Contents

1 Introduction

This thesis aims at exploring lexical blends[1] in Swedish. Blending is a word formation process in which two or more words are concatenated, where at least one of the words is reduced. An example of a blend is *motell* ('motel'), where the reduced forms of *motor* ('motor') and *hotell* ('hotel') has been concatenated, e.g. `mot-el`, `mo-tell`, or `m-otell`.

Blending as a word-formation process has gained popularity in recent years (Mattiello, 2013, p. 111). This increase in popularity is also seen in linguistic research, where blends have received attention in recent years (Gries, 2004a,b; Lehrer, 2007; Mattiello, 2013; Renner, 2015; Ronneberger-Sibold, 2012).

As blending has become more popular, the question regarding its treatment in natural language processing (NLP) has been explored. The approaches towards blends from a computational perspective generally come in three forms:

1. Generation: is it possible to generate satisfactory blends given two source words (Gangal et al., 2017; Schicchi and Pilato, 2017)
2. Detection: can a system be constructed which is able to distinguish blends from other types of words (Cook, 2010, 2012)
3. Identification: can a system be constructed which is able to predict the source words of a blend (Cook, 2010; Cook and Stevenson, 2007, 2010)

This thesis will focus on (3) using machine learning, e.g. can a system be constructed that is able to identify the source words of blends. In Swedish, compounding is a popular and productive word formation process (Bolander, 2005, p. 86). Many natural language processing systems require some type of compound splitting to analyze Swedish (Sjöbergh and Kann, 2004). This is because most compounds are not present in any lexicon, which many NLP systems rely on as a source for word information. The ability to identify which words formed a compound allows the system to derive information regarding the compound based on the source words.

Blends are similar to compounds in their construction, as two or more words are concatenated as in compounding, but with the addition of a reduction to some of the words. It is also the case that most blends do not appear in any lexicon. The simplest way of analyzing blends would be to treat them as compounds, where information regarding the blend can be derived from the words used to create the blend.

The primary aim of this thesis is to identify the source words of blends using machine learning so that information regarding the blend can be derived from its source words. The secondary aim of the thesis is to explore the dataset of blends. To explore Swedish blends, experiments performed on English blends will be performed on Swedish blends.

The contributions of this thesis to research regarding blends are the following:

- Exploration of previous experimental results regarding blends.
- A description of Swedish blends.
- The development of a classifier which is able to predict source words of blends.
- Evaluation of feature importance in classification.

[1] *blend* will be used instead of *lexical blend* in the running text.

2 Background

2.1 Word formation

Affixation: Affixation is a word formation process that creates new words by appending an affix to an existing word. Different affixes have different properties when used to create a new word. Some affixes change the part-of-speech of a word and while others change the semantic meaning (Bolander, 2005, p. 94-102).

In (1) a verb is changed into a noun with the suffix *-an*, in (2) a noun is changed into a verb with the suffix *-a*, in (3) a noun is changed into an adjective by the suffix *-ig* and in (4) a noun is changed into an adverb by the suffix *-vis*.

(1) *önska* (en. 'to wish') (Verb) → *önsk-an* ('wish') (Noun)
(2) *nätverk* ('network') (Noun) → *nätverk-a* ('networking') (Verb)
(3) *nörd* ('nerd') (Noun) → *nörd-ig* ('nerdy') (Adjective)
(4) *grupp* ('group') (Noun) → *gruppvis* ('groupwise') (Adverb)

The above affixations were created through the addition of a suffix, but an affixation may also use a prefix. In contrast to suffixes, prefixes tend to change the semantic meaning rather than the part-of-speech as shown in (5-6).

(5) *uppskattad* ('appreciated') (Adjective) → *o-uppskattad* ('unappreciated') (Adjective)
(6) *kämpa* ('fight') (Verb) → *be-kämpa* ('fight against') (Verb)

In (5) the meaning of *o-uppskattad* ('unappreciated') is an antonym of *uppskattad* ('appreciated'). In (6) the meaning of *kämpa* ('fight') is specified further. In (6) the verb *kämpa* ('fight') is also transformed from an intransitive verb to a transitive verb (Bolander, 2005, p. 99).

Compounding: Compounding is a word formation process in which two or more words are combined to form a new word (Plag, 2003).

In Swedish compounding is a common process for introducing new words (Bolander, 2005, p. 86). An example of compounding is *slutdiskussion* ('final discussion') that is formed by concatenating *slut* ('final') with *diskussion* ('discussion').

(7) *Slutdiskussion* ('final discussion') = *Slut* ('final') + *diskussion* ('discussion')

Some cases of Swedish compounding requires an infix called an interfix[2] to combine the words, such as in (8) where the interfix *-s-* is inserted between the first and second part of the compound.

(8) *Fotbollsmatch* = *Fotboll* ('football') + s + *match* ('game')

Compounds can be classified based on the semantic relationship between the source words. Two types of relationships is described in Bolander (2005, p. 87-88): copulative and determinative compounds. Copulative compounding is when both words in the compound act as semantic heads, e.g. *blå+grön* ('blue-green'), meaning something that is blue and green. Determinative compounds are when one word is the semantic head and the other word modifies it (Bolander, 2005, p. 88). For example *kaffe+maskin* ('coffee machine') is a machine that produces coffee, in the compound, *maskin* ('machine') acts as the head being modified by *coffee* ('coffee').

Clipping: Clipping is a word formation process that removes some parts of a word. For example, *laboratory* can be clipped to *lab* by removing the ending characters *-oratory*. Another type of clipping removes characters both in the beginning and in the ending of a word. For example, a common internet slang for *okay* is *k*. In this case, the clipping has not only removed the ending of the word as in *lab* but the initial character as well.

[2] sv. fogemorfem

2.2 Lexical Blends

2.2.1 Properties and classification of lexical blends

A blend is created by taking two words and concatenating them, where one of the words is reduced (Mattiello, 2013, p. 112). The process has no regular or clear rules according to Mattiello (2013, p. 111). Primarily the process seems to be driven by loose heuristics whose purpose is to combine two words in a satisfactory manner, no matter the regular word formation constraints (Renner, 2015).

Even if the process is irregular and seems random, blends can be classified according to some properties. First, blends can be classified based on how the source words are reduced. Example (1) and (2) show how the beginning or the ending string in the first or second source word may be reduced.

(1) Start of word reduction: brunch = br(eakfast) + (l)unch
(2) End of word reduction: brunch = br(eakfast) + (l)unch

Secondly, blends may be categorized based on whether the characters from the source words overlap in the blend or not. Overlapping characters can be seen as indeterminacy of membership, where it is impossible to determine if the overlapping characters belong to the first or second source word. For example, *hemester* = ***hem*** ('home') + *[s]**em**ester* ('vacation')[3] may contain a reduction of both source words, or only one. In *h**em**ester*, the two overlapping characters `em` can come from either *hem* ('home') or *semester* ('vacation')[4]. From the word form of the blend, it is impossible to determine which of the words contributed the characters `em`.

A curious case appears when considering the blend *noverlap*, which is the combination of *no* and *overlap* = *n**o*** + ***o**verlap*. This is a special blend, as both of the source words can be recovered from the word form of the blend. In the blend *noverlap*, it can not be determined which of the source words have been reduced, e.g. which word contributed the ○[5].

Similarly to compounds, the source words in blend stand in a semantic relation to each other according to Mattiello (2013, p. 123-125). The source words can stand in a determinative relation to each other, where one word is the semantic head and the other word the modifier, as in *funderwear (= f**un*** + ***un**derwaer)*, which is underwear with fun bright colors according to (Mattiello, 2013, p. 123). A Swedish example of a determinative blend would be *bloppis* = *b**lo**[gg]* ('blog') + ***lo**ppis* ('flea market'). The words can also stand in a copulative semantic relation to each other. Source words in a copulative relationship are both semantic heads and have the same semantic status according to Mattiello (2013, p. 125). A Swedish example of a copulative blend is *blok* = *bl**o**[gg]* + *[b]**o**k* ('book'). The blend *blok* denotes a blog that is a book, or a book that is a blog. Table 1 show the different combinations of the properties described above.

2.2.2 Quantitative studies of lexical blends

Several studies regarding the statistical properties of English blends has been performed. Relating to the recognizability of the source words in blends, Gries (2004a) finds that the shorter word contributes more to the blend than the longer source word. Gries (2004a) also investigate the similarity in terms of graphemes and phonemes between the source words in blends. The study shows that the source words show a higher graphemic than phonemic similarity. In a later study, Gries (2012) investigates several hypotheses that have been put forward by previous studies. The study shows that the first source word is significantly shorter than the second source word in terms of phonemes, characters, and syllables. The study also investigates the frequency of the source words in a reference corpus (Reuters corpus, English). It is found that the first source word is significantly more frequent than the second source word.

[3]Overlap in blends will be indicated by bold letters and word reduction will be indicated by enclosing the reduced part of a word in brackets.

[4]The case may also be that parts of the overlap come from the first word and the other part from the second word.

[5]Noverlap will be used to denote the blends whose source word does not overlap.

Table 1: Categorization of blends based on reduction, overlap and semantic relation. Cop. = copulative relation, Det. = determinative relation.

I	REDUCTION	OVERLAP	RELATION	SW_1	SW_2	BLEND
1	Both	True	Cop.	bl**o**[gg] ('blog')	[b]**o**k ('book')	Blok
2	One	True	Cop.	bl**o**[nd] ('blonde')	**o**range ('orange')	Blorange
3	Both	True	Det.	m**ot**[or] ('motor')	[h]**ot**ell ('hotel')	Motell
4	One	True	Det.	b**lo**[gg] ('blog')	**lo**ppis ('flea market')	Bloppis
5	Both	False	Cop.	sk[ed] ('spoon')	[g]affel ('folk')	Skaffel
6	One	False	Cop.	dans ('dance')	[bal]ett ('ballet')	Dansett
7	Both	False	Det.	prome[nad] ('stroll')	[mi]nut ('minute')	Promenut
8	One	False	Det.	alko[hol] ('alcohol')	läsk ('soda')	Alkoläsk

The identification of source words of blends have been done in (Cook, 2010; Cook and Stevenson, 2007, 2010). The work in Cook and Stevenson (2010) builds on the work in (Cook and Stevenson, 2007), thus only (Cook and Stevenson, 2010) will be reviewed.

The dataset of blends used in Cook and Stevenson (2010) was collected from `www.wordspy.com` and from previous studies. The dataset contains 1186 blends, the dataset used in the study is a subset of 342 blends. To generate candidate word pairs each blend was split into n parts. Each split contains a prefix and a suffix part, of minimum length 2. For example, the blend *motel* is split into two parts: (`mo, tel`) and (`mot, el`). A set of candidate source words is generated from two sources: (1) CELEX lexicon (Baayen et al., 1993) and (2) the 100k most common words from the Web 1T 5-gram corpus (Brants and Franz, 2006). The candidates for the first source word is all words that have the identical beginning string as the first part of the word split. The candidates for the second source words are all words that have the identical substring as the second part of the split.

To predict the correct word pair each word pair is associated with a set of features. The features capture a variety of frequency measurements from the Web 1T 5-gram corpus (Brants and Franz, 2006), the contribution to the blend from the source words, the semantic relationship between the source words, and the syllable structure of the source words.

The features were used to train a feature ranking model and a perceptron model. The feature ranking model calculates a real-value for each word pair in the following manner:

$$score(sw1, sw2) = \sum_{i}^{len(f)} \frac{arctan(f_i) - mean(f_i, cs)}{sd(f_i, cs)} \quad (1)$$

The model assigns scores to each word pair in the following way: for each feature, the mean and standard deviation is calculated. To calculate the score for a particular candidate pair P, the mean of feature i is subtracted from the arctan of feature i for P, this value is then divided by the standard deviation of feature i. The score is calculated in this manner to normalize each value, and to reduce the influence of outliers through using the arctan instead of the feature value (Cook, 2010). The feature ranking model output is a list of word pairs ranked according to the score assigned by Equation (1).

To evaluate the feature ranking model and the perceptron model, 10 fold cross-validation was used. The performance was measured by the accuracy at rank 1. The feature ranking model and the perceptron model had the same performance, wherein 40% of the cases a correct candidate pair was scored the highest. This was compared to two baselines: a random baseline which achieved an accuracy of 6% and an informed baseline with an accuracy of 27%.

2.3 Statistical learning

2.3.1 Regression models

Regression analysis is a method of predicting a real-value for a set of observations $x \in X$ each associated with some feature vector $\vec{f}$.

Linear regression: Linear regression is a model which predicts a real-value y for a observation x. The output is based on the relationship between values in the feature vector $\vec{f}$ of observation x and a set of learned weights w (Jurafsky and Martin, 2009, p. 228-229).

$$score(f) = w_0 + \sum_{i=1}^{N} w_i * f_i \tag{2}$$

Each feature in $\vec{f}$ has a weight associated with it, learned during training. The weights are estimated by minimizing the sum-squared error (Jurafsky and Martin, 2009, p. 230). The model will assign weights that as closely as possible capture the relationship between the feature values and the actual value.

Logistic regression: Linear regression outputs a real-value for each observation. In classification, the task is to assign a class to each observation (Jurafsky and Martin, 2009, p. 231).

Logistic regression aims at converting a real-valued output from a linear combination of the features and their coefficients in $\vec{f}$ to a binary value $[0,1]$ representing the True and False class and the probability that the feature vector $\vec{f}$ of observation x belongs to 1 or 0. To estimate the probability that observation x belongs to the true or false class, the inverse *logit* function is used. The inverse *logit* function is a way of mapping a real-value to a binary value.

$$predict(f) = logit(w_0 + \sum_{i=1}^{N} w_i * f_i) \tag{3}$$

The predict function will output the probability that observation x with vector $\vec{f}$ belongs to the true class. If the probability that the observation belongs to the true class is larger than the probability that x belongs to the false class, x is classified as true, else false.

2.3.2 Word and character embeddings

Word and character embeddings rely on unsupervised learning to model the meaning of words based on distributional semantics, summarized famously by Firth (1961): *You shall know a word by the company it keeps*.

There are two types of embeddings used in this thesis, word embeddings, and character embeddings. Both word and character embeddings use the same context extraction algorithms, CBOW (continuous bag of words) or skip-gram.

The CBOW models take as input a set of items (words in a sentence or character in a word) that surrounds the target unit (a word or a character). The model aims at predicting the target word or character based on the context surrounding it.

In the skip-gram model, the input is a target unit (a word or a character) and the output is the words or characters that surround it. The difference between the two techniques can be seen in Figure 1 reproduced from (Mikolov et al., 2013).

Word embeddings: Word embeddings are extracted from a corpus. Each word in the corpus is associated with a context consisting of the k preceding and succeeding words constrained to the current sentence. The words in the context are then encoded in a vector associated with the word (Mikolov et al., 2013).

Character embeddings: Character embeddings are created by partitioning a word into character n-grams. Each n-gram is then given a vector that is the outer product of all character vectors in the n-gram.

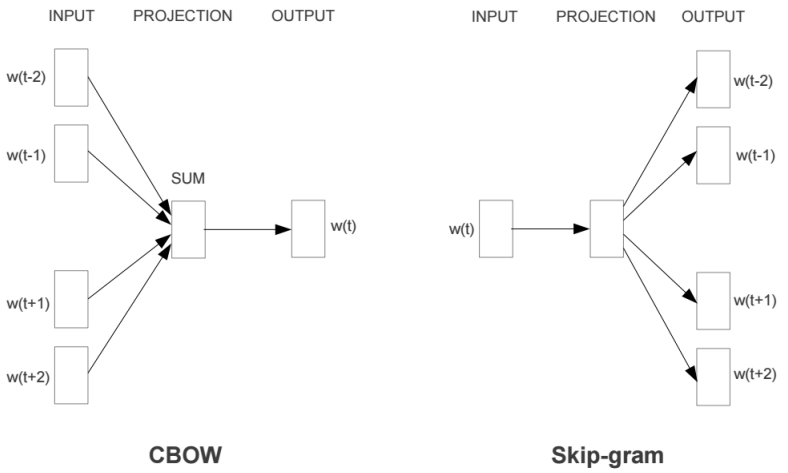

Figure 1: CBOW and skip-gram models. Reproduced from (Mikolov et al., 2013)

Full words are represented by the resulting vector from taking the outer product of all n-grams in the word (Bojanowski et al., 2016).

2.4 Evaluation

This section presents the metrics used for the evaluation of the system. The system operates on pairs of words and *word pair* will be used to denote the data points (or observations). For a particular blend, the correct word pair is used to create the blend, and an incorrect word pair is not used to create the blend.

2.4.1 Classification

The classification experiments are evaluated with precision, recall, and F_1-score. These metrics are used when the system retrieves word pairs from a set of word pairs. This type of evaluation relies on a confusion matrix, where each prediction by the system is classified as either True Positive, False Positive, False Negative or True Negative.

A True Positive is when a correct word pair is retrieved. A False Positive is when an incorrect word pair is retrieved. A False Negative is when a correct word pair is not retrieved. A True Negative is when an incorrect word pair is not retrieved (Schütze et al., 2008, p. 155). The relationship between the different classifications is often visualized in a confusion matrix, as shown below in Table 2.

Table 2: Confusion matrix showing the categorization of binary classifications.

	RELEVANT	NOT RELEVANT
RETRIEVED	TP	FP
NOT RETRIEVED	FN	TN

From the confusion matrix, the accuracy, precision, and recall of the system can be calculated. Accuracy is the number of correctly retrieved and correctly not retrieved word pairs, divided by the total number of word pairs, e.g. the fraction of correct retrievals. Precision is the number of relevant word pairs retrieved divided by the number relevant and not relevant word pairs retrieved. The recall is the

number of relevant word pairs retrieved divided by the number of relevant word pairs retrieved plus the number of relevant word pairs not retrieved.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

$$precision = \frac{TP}{TP+FP} \tag{5}$$

$$recall = \frac{TP}{TP+FN} \tag{6}$$

The harmonic mean between precision and recall is called the F_1-score and is calculated as follows:

$$F_1\text{-score} = 2 * \frac{(precision * recall)}{precision + recall} \tag{7}$$

The F_1-score shows the performance of the system, where both recall and precision is taken into account.

2.4.2 Ranking

Rankings may be evaluated by various metrics which capture the position of the correct word pair in a ranked list. Two measurements will be used in this thesis, Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR). MAP will be used for rankings in which there are several correct word pairs and MRR will be used for rankings where there is only one correct word pair.

Mean Reciprocal Rank: The metric MRR captures the highest ranked correct word pair and calculates its score by dividing 1 by the rank. The MRR for several rankings is obtained by calculating the mean of the rankings (Jurafsky and Martin, 2009, p. 821), e.g:

$$MRR = \frac{1}{N}\sum_{k=1}^{N}\frac{1}{R_k} \tag{8}$$

Where R_k is the index k of the highest ranking correct word pair in the ranking, and N is the number of rankings. This means that the MRR of multiple lists simple is the mean ranking of the correct word pair. For example, the mean ranking of the first `1` in the four rankings: `(0,1,1)`, `(1,0,0,0)`, `(0,0,0,1)` and `(0,1,0,1)` is:

$$MRR = \frac{\frac{1}{2}+\frac{1}{1}+\frac{1}{4}+\frac{1}{2}}{4} = \frac{0.5+1.0+0.25+0.5}{4} = 0.5625 \tag{9}$$

Mean Average Precision: For a ranking R, extract the position of each correct word pair $N = \{r_i | r_i \in R \wedge r_i = True\}$. For each $n \in N$, calculate the precision at ranking $1...n$. The MAP is then calculated by dividing the mean precision by $|N|$, e.g. the number of correct word pairs (Schütze et al., 2008, p. 160).

$$MAP = \frac{1}{|N|}\sum_{j=1}^{|N|}(\frac{1}{m_j}\sum_{k=1}^{m_j} precision(R_{jk})) \tag{10}$$

For example, let's consider the following ranking: $(0,1,0,1)$. For this ranking, the MAP score is calculated for two indices in the ranking, 2 and 4. Thus, the precision is calculated between the indices $[1,2] = \frac{1}{2} = 0.5$ since the first correct word pair appear at rank 2. The precision is also calculated between the indices $[1,4] = \frac{2}{4} = 0.5$ since the second correct word pair appears at rank 4. The MAP is calculated by dividing the sum of precision scores by the number of measurements, e.g.

$$MAP = \frac{0.5+0.5}{2} = 0.5 \tag{11}$$

MAP for multiple rankings is calculated as the mean MAP over all rankings. It should be noted that performing MAP with datasets containing only one correct word pair is equivalent to using MRR.

2.4.3 Cross-validation

Cross-validation is an evaluation technique for estimating the performance of a system on the dataset. Cross-validation if performed by randomly sorting the word pairs in the dataset into n different sets, denoted folds. The system is evaluated by using each fold once as testing set and the remaining folds as training. The final performance is given by the mean of the results for each fold, as shown below:

$$CV_{score} = \frac{1}{n}\sum_{i=1}^{n} result(test_i, train_{\{j\in 1\ldots n | j\neq i\}}) \tag{12}$$

Traditionally, datasets are divided into a training set and a testing set. A drawback of this is that the evaluation is only performed on a subset of the dataset. The use of cross-validation allows the system to be evaluated on the complete dataset.

3 Aims and research questions

The aim of this thesis is to investigate Swedish blends. The thesis consists of two parts, first, a corpus/lexicon study in which the properties of blends is investigated. This thesis aims to investigate the following questions regarding the structure of blends:

1 How does the first and second source words in blends relate to each other in terms of:
  1.1 Orthographic and phonemic length.
  1.2 Word frequency in a corpus.
  1.3 Orthographic and phonetic contribution to the blend.

Secondly, the thesis aims at constructing a model which is able to predict the source words of the blends within the dataset. Primarily, the importance of the features will be investigated. The feature set used will be motivated by the results from research questions 1.1 - 1.3. In addition to exploring the feature importances, the difference between overlapping blends, *motell* = *m**ot**[or]* ('motor') + *[h]**ot**ell* ('hotel'), and noverlapping blends, *alfanummer* = *alfa[bet]* ('alfabet') + *[telefon]nummer* ('telephone number') will be explored.

2 Given a model, the following three research questions are posed:
  2.1 Which features provide the best predictors of the source words in blends?
  2.2 Is there a difference in performance between the overlapping and noverlapping blends?
  2.2 Is there a difference between the feature importance for overlapping and noverlapping blends?

4 Data & Method

This section describes the data and resources used, the model creation method and how the experiments are set up. The code for the project is available at
https://github.com/adamlek/swedish-lexical-blends

4.1 Data

4.1.1 Corpus

To extract candidate source word frequencies and word embeddings a corpus was compiled from news texts. The corpus contains sentences from GP (Göteborgsposten) and Webbnyheter (Webnews) gathered between the years 2001 and 2013. The data was obtained from Språkbanken[6]. The number of tokens and types in the corpora is described in Table 3.

Table 3: The number of tokens and types in the corpora.

CORPUS	TOKENS	TYPES
Webnews	218 298 739	634 700
Göteborgsposten	188 594 777	683 265
Combined	406 893 516	964 070

4.1.2 SALDO

SALDO is a morphological and semantic lexicon for Swedish (Borin et al., 2008). SALDO contains Swedish lemmas with information regarding the conjugations and the semantics of words. SALDO have been used to extract candidate source words for the blends (as described in Section 4.2.1.).

4.1.3 Embedding models

The word embeddings model was created from the corpus described above. The model uses CBOW contexts, with a window of 5 words and a minimum frequency of 1. The character embeddings model was trained on the Swedish Wikipedia and Common Crawl (Grave et al., 2018), using CBOW contexts and a window of 5. The minimum frequency was set at 1 and the model uses negative sampling set at 10.

4.1.4 Dataset of Lexical blends

A list of blends has been collected from the following sources: (a) Nyordslistan [7], (b) Kiddish[8], (c) Slangopedia[9], (d) Språktidningen[10] and (e) personal correspondence. The full list of lexical bends and their relative frequency in the corpus is described in Appendix A. It is possible that more Swedish blends exist, but due to the scope of the thesis, this amount was deemed sufficient.

In total, the list of blends contains 223 blends, of which 158 have both source words present in the SALDO lexicon. Only the blends with both source words in the SALDO lexicon have been used in the machine learning model.

The 65 blends without source words in SALDO are included in the statistical analysis of blends. Thus, when investigating the statistical properties of Swedish blends the full dataset of 223 blends are used.

[6] https://spraakbanken.gu.se/
[7] http://www.sprakochfolkminnen.se/sprak/nyord/nyordslistor.html
[8] http://www.kidish.se/vad-ar-kidish/
[9] http://www.slangopedia.se/
[10] http://spraktidningen.se/

During the data collection, some blends were excluded based on the structure of the source words. Blends where the second source word contains a reduction of the ending substring, and blends where the first source word contain a reduction of the beginning substring have been excluded. For example, the blend *mokus* = *mo[r]* ('mother') + *kus[in]* ('cousin') is excluded since the ending substring of the second source word *kusin* ('cousin'), `-in`, have been reduced. Additionally, blends where one source word is inserted into the other are also excluded, for example the blend *samargbete* = *sam**ar**bete* ('collaborative work') + ***ar**[g]* ('angry') have been excluded as the second source word *arg* ('angry') is inserted into the first source word. The exclusions were done to reduce the number of ways that a blend can be split when searching for candidates (see Section 4.2.1.),

The 158 blends with both source words in the SALDO lexicon are divided into two subsets, one containing the blends where the source words overlap, and one containing the blends whose source words do not overlap. The list of blends used for machine learning is summarized in Table 4.

Table 4: Summary of the blend dataset.

BLEND TYPE	COUNT
Overlap	63
Noverlap	95
All	158

4.1.5 Gold standard

The gold standard source words for each blend is determined by the description of the blend given. For example, the blend *tvångla* = *tv[inga]* ('force') + *[h]ångla* ('make-out') is described as (translation by the author):

(1) Att hångla med någon mot hens vilja. Ett sammansatt ord av verben *tvinga* och *hångla*. ('To make-out with someone against their will. A compound word from the verbs *force* and *make-out*.') [11]

The blend is described as a compound, but through observation, it can be seen that the source words must have been reduced to form the blend. In other cases the source words are not explicitly stated but a description of the meaning is given. For example the blend *göteburgare* = *Göte**b**[org]* ('Gothenburg') + *[ham]**b**urgare* ('hamburger') is described as follows (translation by the author):

(2) En återuppvärmd hamburgare. Uttrycket kommer sig av det Göteborgska fenomenet att grillkioskerna där ofta har den ovanan att steka på en hög med burgare på morgonen, för att sedan värma på dem precis innan servering. Om man skall käka Göteburgare så kan man lika gärna gå och käka på Donken. ('A reheated hamburger. The expression comes from the gothenburgian phenomena where fast food stands have the habit of pre-cooking burgers in the morning, to later re-heat them before serving. Example: If you're going to eat a Göteburgare you might as well go to McDonalds.') [12]

The description states that the word originates from *hamburgare* ('hamburger') and from a phenomenon observed in Göteborg. The gold standard source words for *göteburgare* is thus recorded as *Göteborg* (Gothenburg') and *hamburgare* ('hamburger'). A similar example is the blend *trånglig* = ***trång*** ('tight') + *[k]**ång**lig* ('troublesome'), described as (translation by the author):

[11] http://www.kidish.se/#ordlista, accessed 2018-09-11
[12] http://www.slangopedia.se/ordlista/?ord=G%F6teburgare, accessed 2018-09-10

(3) Kläder som är trånga och krångliga att ta på sig är trångliga. ('Clothes which are tight and troublesome to put on are tightysome.') [13]

The description of the blend explains its meaning. The words *trång* ('tight') and *krångliga* ('troublesome') are selected as the source word. In the list of blends *trångliga* is saved as *tranglig*, e.g. as indefinite. The source word *krångliga* ('troublesome') is reduced to its indefinite form, *krånglig* ('troublesome'), to fit the blend. In general, when the meaning of a blend is described with affixes and affixed source words, the non-essential affixes have been removed from the gold standard.

In some cases neither of source word is mentioned in the description, for example *skypebo* = *skype* ('Skype') + *[sam]bo* ('domestic parter'/'people cohabiting'):

(4) När man är Skypebo så sitter man med varsin 'padda' och umgås. Det känns faktiskt mera fysiskt än den kontakt man får via ett telefonsamtal. Man behöver inte alltid ha kameran på för att känna den närheten. ('When you are a skypebo you each sit and use a pad. It feels more physical than the connection you get when talking on the phone. You don't always need the camera on to feel the closeness.')[14]

The selected source words are the most reasonable words according to the current author. In this case *skype* ('Skype') based on the characters in the blend, and *sambo* ('domestic parter'/'people cohabiting') based on the meaning and other blends using *sambo* ('domestic parter'/'people cohabiting'), specifically the final substring `bo`[15].

In essence, the source words are taken from the description and modified in such a way that they fit the word form of the blend. If any changes are made to the source words in the description, as few changes as possible have been done. For the cases where the source word is not present in the description, the source word have been selected by the current author. Only one correct word pair is selected for each blend.

4.2 Method

4.2.1 Candidate selection

For each blend, two sets of words are extracted from SALDO. The first set, denoted as the prefix set corresponds to the candidates of the first source word in the blend. The second set denoted as the suffix set, correspond to the candidates of the second source word of the blend.

The prefix set consist of each word which has the same two starting characters as the blend. The suffix set consist of each word which has the same two ending characters. There is no need to search for longer substrings, as these will already be included in the prefix and suffix set. To generate candidate pairs which can form the blend, the product of these two sets is taken. The product set is then filtered to remove the word pairs which cannot create the blend. The remaining candidate word pairs are filtered based on if the blend overlap, or not. The candidate word pairs of overlapping blends can be combined in two or more ways to create the blend. The candidate word pairs for noverlapping blends can create the blend in one and only one way.

For example, the blend *motell* ('motel') starts with `mo` and ends with `ll`, thus the candidate word pairs is all word combinations where the first word starts with `mo` and the second word ends with `ll`. For example, the pair *mormor* ('grandmother') and *Kjell* (First name) is extracted, but is filtered out because the words cannot be combined into the blend. The pair (*motor, Kjell*) ('motor'/First name) is extracted and can be merged into the blend. This pair is discarded as *motell* ('motel') is an overlapping blend and (*mormor, Kjell*) ('grandmother', First name) can only create *motell* ('motel') in one way. The

[13] http://www.sprakochfolkminnen.se/sprak/nyord/inskickade-nyord/inskickade-nyord-2013, accessed 2018-09-10.
[14] http://www.sprakochfolkminnen.se/sprak/nyord/inskickade-nyord/inskickade-nyord-2013, accessed 2018-09-11
[15] See Appendix A: *exbo*, *flerbo* and *helgbo*.

pair (*motala, kartell*) (City, 'cartel') can create the blend in two ways: (`mot, ell`) and (`mo, tell`) and is saved as a candidate pair.

Each entry in the dataset used by the model is the splits of the word pair that create the blend. Thus, the noverlapping dataset contains only one correct word pair, because the correct pair can be split in one and only one way. The overlapping dataset contains *n* correct word pairs, where *n* is the number of splits which create the blend. For example, the gold standard word pair *(motor, hotell)* ('motor, hotel') for the blend *motell* ('motel') contain two correct splits: `(mot,ell)` and `(mo,tell)`. The motivation for separating the word pairs in this manner is because the different splits provide different information in regards to contribution and frequency (see features 30-31, 35 and 37 in Section 4.2.2.).

4.2.2 Features

This section presents the features included in the model. The features are classified into four categories determined by their target domain. A summary of the categories and a definition is given in Table 5.

Table 5: Classification of features according categories based on the target domain.

CATEGORY	TARGET DOMAIN
SEMANTICS	Semantics of the source words and the blend.
SOURCE WORDS	The relationship between the source words.
BLEND	The relationship between the source words and the blend.
FREQUENCY	The frequency of the source words in the corpus.

Below, each feature and its motivation is explained, followed by a summary of all features in Table 6 (Page 17). It should be noted that none of the features are developed specifically for Swedish. Only the resources (corpus, lexicon, embedding models) are language dependent.

Embedding score (1-3, 7-9): For the source words and the blend, the context vector itself is captured as a feature. The feature is represented by the sum of the word vector.

Embedding similarity (4-6, 10-12): The similarity between the two source words using word embeddings is captured, if the blend is in the vocabulary (e.g. in the news corpus) the similarity between the source words and the blend is calculated. Typically, the blend will not be present in the word embedding model. In the character embeddings model, the blend can be recreated from the *n*-grams in the model which allow the feature to measure the similarity between the source words and the blend for all blends. The models produce a vector of *n* dimensions for each word, and the similarity between the vectors are measured using cosine similarity, shown below:

$$cos(a,b) = \frac{a \cdot b}{|a| \cdot |b|} = \frac{\sum_{i=1}^{n} a_i b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\sqrt{\sum_{i=1}^{n} b_i^2}} \tag{13}$$

Bi- and trigram similarity (13-16): The bi- and trigram similarity capture the amount of shared bi- and trigrams between the two candidate words. The bi- and trigram similarity is calculated in the following manner, where $ngrams(w)$ is a sequence containing all *n*-grams of the word *w*:

$$\frac{|ngrams(w_1) \cap ngrams(w_2)|}{|ngrams(w_1)|} \qquad \frac{|ngrams(w_2) \cap ngrams(w_1)|}{|ngrams(w_2)|} \tag{14}$$

This feature captures the orthographic similarity of the words for substrings of length 2 and 3.

Longest common substring (17): The longest shared substring of the first and second source word is captured as a feature. For example, the LCS of the candidate pair *(spöke, psyke)* ('ghost', 'psyche') for

the noverlapping blend *spyke* is `ke`, thus $LCS(\text{spöke}, \text{psyke}) = 2$.

Levenshtein distance (21-24): It may be useful to know how many transformation operations are needed to convert the source words into each other and the blend. To measure this, the Levenshtein distance between the source words and the blend is measured.

IPA Levenshtein distance (18-20): In addition to orthographic Levenshtein distance, the Levenshtein distance between IPA representations are captured. This is done by translating the string into IPA symbols using the python package `epitran` (Mortensen et al., 2018). From the IPA representation, the Levenshtein distance between the source words and the blend is calculated.

Phonemes (24-25): The number of phonemes in the source words relative to the number of phonemes in the blend is captured as a feature. The feature is calculated in the following manner, where $phonemes(w)$ is a sequence of all phonemes in the word w:

$$\frac{|phonemes(w)|}{|phonemes(blend)|} \tag{15}$$

Syllables(26-27): The number of syllables in source words relative to the number of syllables in the blend is captured as a feature. The number of syllables in a word is estimated by counting the number of vowels.

This method will contain some errors as certain neighboring vowels count as the nucleus of one syllable, while other times the boundary between syllables is between the vowels. A simple test on the NST lexicon which contains syllable boundaries encoded in X-SAMPA showed that for 94% of the words, counting the number of syllables in this manner yielded the correct amount. For the scope of this thesis, this is deemed sufficient.

Word length(28-29): For each source word, its character length relative to the character length of the blend is captured.

$$\frac{len(sw_n)}{len(blend)} \tag{16}$$

The character length of the different source words is an important factor (Cook, 2010), and this feature aims to capture the relative character length of the source words in relation to the blend.

Contribution (30-31): The contribution of characters from each source word to the blend is calculated. The contribution is calculated by dividing the number of character contributed by the first and second source word to the blend by the number of characters in the blend. For example the contribution of the split (`mo, tell`) is 2 for `mo` and 4 for `tell`.

The contribution of the different source words has shown to be an important factor in Cook (2010). This feature aims at capturing simple contribution to the blend in terms of characters contributed to the blend.

Removal (32): This feature takes the length of both candidate words divided by the length of the blend. This feature capture how much of the candidate words combined is removed to create the blend.

$$\frac{len(sw_1 + sw_2)}{len(blend)} \tag{17}$$

This feature is aimed at excluding candidate pairs which are very long when combined, such as the combination of two compounds.

Source word splits (33): This feature capture the number of ways to split the candidate pair to create the blend. The feature is primarily aimed at aiding the prediction of overlapping blend, Many possible splits would indicate that the correct interpretation of the blend can be arrived at in many different ways.

Affix frequency (35, 37): The affix frequency is calculated by dividing the frequency of the source words by the frequency of all words with the identical beginning or ending substring. For example, the affix frequency for the pair (*motor, hotell*) ('motor', 'hotel') with the split (`mot, ell`) is calculated in the following manner, where C is the set containing all the word beginning/ending with the acceptable word split (e.g. C_{prefix} contain all candidate words that begin with `mot`):

$$af(motor) = \frac{freq(motor)}{\sum freq(w \in C_{prefix_{mot}})} \tag{18}$$

$$af(hotell) = \frac{freq(hotell)}{\sum freq(w \in C_{suffix_{ell}})} \tag{19}$$

This feature is intended to capture the prominence of the word given other words with the same structure in the beginning/end of the word. Lehrer (2007) and Cook (2010) found evidence that more frequent word given its affix context tends to be more likely source words.

Corpus frequency (34, 36): This feature captures the frequency of the word relative to the corpus (where N is the total number of tokens in the corpus):

$$\frac{freq(w)}{N} \tag{20}$$

The feature intends to capture the relative frequency of the word in the corpus.

4.2.3 Baselines

Random baseline: A random baseline is constructed in the following manner: given a blend, select n candidate pairs at random.

Feature ranking baseline: A baseline system is constructed based on the feature ranking approach used by (Cook, 2010; Cook and Stevenson, 2010). In this approach, each word pair is scored by subtracting the *arctan* value for feature i with the mean of that feature over the whole dataset (the whole dataset is denoted as cs in (20)). This value is then divided by the standard deviation of that feature, e.g.

$$score(sw1, sw2) = \sum_{i}^{len(f)} \frac{arctan(f_i) - mean(f_i, cs)}{sd(f_i, cs)} \tag{21}$$

For each blend, the word pairs are sorted according to the score given by the model. The highest scoring candidate pair is then selected as the correct word pair for the blend.

4.2.4 Experimental setup

The logistic regression model used in the experiments is the implementation available in the Python 3 package `sklearn`[16].

Three experiments are performed, all using cross-validation. The amount of folds for the overlapping blends is 6, for noverlapping blends, the number of folds is 9 and for the combined dataset the number of folds is 10. The number of folds is selected to create roughly evenly size folds, where n should be as close to 10 as possible. During the development, the first fold of the noverlapping blends was used as

[16] www.scikit-learn.org/

the test set. For this reason, the overlapping dataset use 6 folds such that there are 10-11 blends in each fold.

Three types of experiments are performed to evaluate the model:

1. Ranking experiment: The logistic regression model is compared against the two baselines described in Section 4.2.3. The purpose of these experiments is to evaluate the model's performance in comparison to the baselines. To perform the evaluation, the models will rank the word pairs according to the probability that they belong to the true class.
2. Classification experiment: The logistic regression model is evaluated with precision, recall, and F_1-score. The precision, recall, and F_1-score is calculated on the top n ranking word pairs. The systems will be evaluated on the top 3 and top 5 ranking word pairs.
3. Feature ablation experiment: Feature ablation in three variants is performed to investigate the impact of the features. The feature ablation will be performed on the features individually (see Table 6), groupings of features (a set of features which capture the same type of information) and by categories of features (as defined in Table 5).

Table 6: Summary of the features used and their intended target domain. SEMANTIC features aims at capturing similarity between words, SOURCE WORDS aims at capturing the relationship between the source words, BLEND aims at capturing the relationship between the source words to the blend, and FREQUENCY aims at capturing the frequency of the source words.

ID	FEATURE	CATEGORY
1	sw_1 character score	SEMANTICS
2	sw_2 character score	SEMANTICS
3	*blend* character score	SEMANTICS
4	sw_1, sw_2 character similarity	SEMANTICS
5	sw_2, *blend* character similarity	SEMANTICS
6	sw_1, *blend* character similarity	SEMANTICS
7	sw_1 word score	SEMANTICS
8	sw_2 word score	SEMANTICS
9	*blend* word score	SEMANTICS
10	sw_1, sw_2 word similarity	SEMANTICS
11	sw_2, *blend* word similarity	SEMANTICS
12	sw_1, *blend* word similarity	SEMANTICS
13	sw_1, sw_2 character bigram similarity	SOURCE WORDS
14	sw_2, sw_1 character bigram similarity	SOURCE WORDS
15	sw_1, sw_2 character trigram similarity	SOURCE WORDS
16	sw_2, sw_1 character trigram similarity	SOURCE WORDS
17	sw_1, sw_2 LCS	SOURCE WORDS
18	sw_1, sw_2 IPA levenshtein distance	SOURCE WORDS
19	sw_2, *blend* IPA levenshtein distance	BLEND
20	sw_1, *blend* IPA levenshtein distance	BLEND
21	sw_1, sw_2 levenshtein distance	SOURCE WORDS
22	sw_2, *blend* levenshtein distance	BLEND
23	sw_1, *blend* levenshtein distance	BLEND
24	sw_1 phonemes	BLEND
25	sw_2 phonemes	BLEND
26	sw_1 syllables	BLEND
27	sw_2 syllables	BLEND
28	sw_1 length	BLEND
29	sw_2 length	BLEND
30	sw_1 contribution	BLEND
31	sw_2 contribution	BLEND
32	sw_1, sw_2 removal	BLEND
33	source word splits	BLEND
34	sw_1 corpus frequency	FREQUENCY
35	sw_2 affix frequency	FREQUENCY
36	sw_1 corpus frequency	FREQUENCY
37	sw_2 affix frequency	FREQUENCY

5 Results

5.1 Statistical properties of lexical blends

This section presents the statistical tests performed on the list of blends. Three experiments are performed: (1) the relationship between the source words in terms of length, (2) the contribution in terms of symbols to the blend and (3) the relationship between the source words in terms of frequency in a corpus.

5.1.1 Source word lengths

The relationship between the number of characters, phonemes, and syllables of the first and second source word is investigated. For each category *C* (characters, phonemes, syllables), the null hypothesis is "the first source word and the second source word have the same length in terms of *C*", and the hypothesis to test is "the first source word is shorter than the second source word in terms of *C*".

To test the hypothesis a one-tailed students t-test with an alpha level of 0.05 is performed. The results are shown in Table 7, which shows the mean length of the source words for each category, and the p-value of the test below.

Table 7: Mean, standard deviation and p-value from the t-test comparing the first and second source word in terms of characters, phonemes, and syllables. Bold indicates that the result is significant.

	CHARACTERS		PHONEMES		SYLLABLES	
	SW_1	SW_2	SW_1	SW_2	SW_1	SW_2
Mean	6.31	6.87	5.91	6.36	2.21	2.54
SD	2.86	2.39	2.68	2.20	1.23	1.07
p-value	**0.018**		**0.003**		**0.002**	

The tests show that for character, phonemes, and syllables, the first source words tends to be significantly shorter than the second source word.

5.1.2 Contribution

The contribution of characters to the blend is investigated in two tests. In the first test, the contribution to the blend from the first and second source word is compared. In the second test, the contribution to the blend from the shorter source word is compared to the contribution from the longer source word.

To calculate the contribution, the mean of all word splits is calculated. For example, the word *motell* ('motel') has two possible splits, (`mot, ell`) and (`mo, tell`). The contribution of *motor* ('motor') is thus $\frac{3+2}{2} = 2.5$ and the contribution of *hotell* ('hotel') is $\frac{3+4}{2} = 3.5$.

The null hypotheses are "the contribution from sw_1 and sw_2 are equal" and "the contribution from the shorter source word is equal to the contribution of the longer source word". The hypotheses we wish to test are "the first source word contributes more than the second source word" and "the shorter source word contributes more than the longer source word". The hypotheses are tested using a one-sided paired students t-test with an alpha level of 0.05. The mean, standard deviation and p-values are shown in Table 8.

The t-tests show that neither the first or shorter source word contribute more than the second or longer source word. When testing if the second and longer source word contributes more to the blend, it is shown to be significant with p-values of 3.66-e07 and 2.62e-07 respectively.

Table 8: Mean, standard deviation and p-value from the t-test exploring the contribution of the first and second source word, and the short/long source word. Bold indicates that the result is significant.

	SW_1	SW_2	SHORT	LONG
Mean	3.42	4.71	3.75	4.55
SD	0.44	1.72	1.78	1.83
p-value	1		1	

5.1.3 Frequency

The frequency of the first and second source word in the news corpus (Section 4.1.1.) is compared. The relationship between first and second source word is tested using a one-sided paired students t-test with an alpha value of 0.05. The variance between the list of first and second source words is not equal, thus the t-test does not assume that the lists have the same variance.

The null hypothesis is the following: "The first and the second source words have the same frequency", this is tested against the alternative hypothesis that "the first source word is more frequent than the second source word". The mean, median and p-value from the t-test are shown in Table 9.

Table 9: Mean type frequency, median, and p-value from the t-test investigating the relationship between frequencies of the first and second source word. Bold indicates that the result is significant.

	SW_1	SW_2
Mean	89 387	16 022
Median	6 233	3 140
p-value	**0.02**	

The result of the t-test indicates that the first source word is significantly more frequent than the second source word in the corpus used in this study.

5.2 Source word identification of lexical blends

In this section, the performance of the linear regression model is evaluated in three experiments. The first experiment investigates the accuracy at *n*, the second experiment the precision, recall, and f-score of the top ranking word pairs, and the third experiment the importance of different features using feature ablation.

5.2.1 Ranking experiments

The first experiments evaluate the performance of the system based on the ranking of the word pairs. For each blend, the word pairs are ranked according to the probability that they belong to the true class as estimated by the logistic regression.

The logistic regression is compared against the two baselines described in section 4.2.4. For each blend, the word pairs are ranked and the performance is measured on four different thresholds. For each threshold, the system is regarded as correct if a correct word pair occurs within the top *n* ranking word pairs. The system is tested on the thresholds: 1, 3, 5 and 10, The results for the overlapping blends is shown in Table 10, and the results for the noverlapping blends are shown in Table 11.

Table 10: Model evaluation of overlapping blends and comparison to the baselines. The evaluation is performed by considering the system to be correct if the top n ranking word pairs contain a correct word pair.

SYSTEM	ACC_1	ACC_3	ACC_5	ACC_{10}
Random	0.031	0.063	0.126	0.158
Feature ranking baseline	0.190	0.349	0.365	0.428
Logistic Regression	0.444	0.611	0.666	0.740

Table 11: Model evaluation of noverlapping blends and comparison to the baselines. The evaluation is performed by considering the system to be correct if the top n ranking word pairs contain a correct word pair.

SYSTEM	ACC_1	ACC_3	ACC_5	ACC_{10}
Random	0.021	0.052	0.063	0.094
Feature ranking baseline	0.021	0.063	0.115	0.168
Logistic Regression	0.234	0.416	0.437	0.541

For both overlapping and noverlapping blends, the feature ranking baseline performs better than the random baseline. The logistic regression model performs better than both baselines. It can be observed that the performance of the model is higher for overlapping than noverlapping blend.

The ranking experiment was also performed with the two data sets combined. The results for all blends in the dataset is shown in Table 12.

Table 12: Model evaluation of all blends and comparison to the baselines. The evaluation is performed by considering the system to be correct if the top n ranking word pairs contain a correct word pair.

SYSTEM	ACC_1	ACC_3	ACC_5	ACC_{10}
Random	0.031	0.044	0.088	0.107
Feature ranking baseline	0.069	0.145	0.196	0.240
Logistic Regression	0.322	0.492	0.537	0.606

5.2.2 Classification experiments

In addition to measuring if the correct word pairs are selected in the top n results, a more fine-grained analysis is performed on the top ranking word pairs. In this analysis the precision, recall and F_1-score is calculated on the top 3 and top 5 ranking word pairs.

For each blend, there is only a small amount of correct word pairs, and many more incorrect word pairs, e.g. if there is only one correct word pair and the system selects 5 word pairs, the remaining for word pairs will be incorrect.

To estimate a realistic performance of the system, an upper bound is estimated and compared against. The upper bound is calculated by considering all the correct word pairs as being in the top n suggestions and populating the remaining slots with incorrect word pairs. The normalized score is calculated by dividing the performance of the logistic regression by the upper bound. The normalized score can be viewed analogously to intrinsic evaluation, where the performance of the model is evaluated independently of any application (Jurafsky and Martin, 2009, p. 129). The performance of the logistic regression disregarding the upper bound can be viewed as an extrinsic evaluation, e.g. as part of a pipeline, how good is the top n retrieved word pairs.

The experiment is performed on both the overlapping and noverlapping blends as well as the combined dataset, on the top 3 and top 5 suggestions. The results for the overlapping blends are shown in Table 13 and the results for the noverlapping blends are shown in Table 14.

Table 13: Classification experiment on overlapping blends. The top n ranking word pairs are considered as suggestions by the system and is evaluated using precision (P), recall (R) and F_1-score (F).

	Top 3			Top 5		
System	P	R	F	P	R	F
Logistic Regression	0.364	0.437	0.396	0.285	0.573	0.379
Upper bound	0.777	0.945	0.850	0.498	1.000	0.663
Difference	−0.413	−0.508	−0.454	−0.213	−0.427	−0.284
Normalized score	0.468	0.462	0.465	0.572	0.573	0.571

Table 14: Classification experiment on noverlapping blends. The top n ranking word pairs are considered as suggestions by the system and is evaluated using precision (P), recall (R) and f-score (F).

	Top 3			Top 5		
System	P	R	F	P	R	F
Linear Regression	0.141	0.416	0.210	0.089	0.437	0.148
Upper bound	0.339	1.000	0.507	0.206	1.000	0.342
Difference	−0.198	−0.584	−0.297	−0.117	−0.563	−0.194
Normalized score	0.415	0.416	0.414	0.432	0.437	0.432

The experiments for overlapping and noverlapping blend show that the systems roughly have the same performance in the top three suggestions. For the top five suggestions, the overlapping model performs much better with an f-score 13.9 percentage points above that of noverlap blends.

The precision, recall and F_1-score experiments were also performed on the combined dataset. The results for the combined dataset is shown in Table 15.

Table 15: Classification experiment on all blends. The top n ranking word pairs are considered as suggestions by the system and is evaluated using precision (P), recall (R) and F_1-score (F).

	Top 3			Top 5		
System	P	R	F	P	R	F
Linear Regression	0.240	0.448	0.311	0.180	0.556	0.271
Upper bound	0.513	0.966	0.668	0.321	1.000	0.485
Difference	−0.273	−0.518	−0.357	−0.141	−0.444	−0.214
Normalized score	0.467	0.463	0.465	0.560	0.556	0.558

The performance on the combined dataset shows that the performance is similar to the performance of the overlapping blends.

5.2.3 Feature ablation

The importance of the features is investigated in three feature ablation experiments. The noverlapping blends are evaluated using MRR since there is only one correct word pair among these. The MAP is used for overlapping blends, where there are two or more correct word pairs. The motivation to change

metric is that MAP and MRR allow us to more easily track what effect the different features have on the correct word pairs.

In the first feature ablation experiment, feature ablation is performed on groups of features. The results and feature groups are shown in Table 16.

Table 16: Feature ablation experiments with groups of features removed. Numbers indicate the performance change in percentage points. Changes such as -0.0 or $+0.0$ indicate that the change is positive or negative by an amount ($0.01 > c$) and ± 0 indicate no change.

		OVERLAP	NOVERLAP	ALL
GROUP	FEATURE GROUP	MAP	MRR	MAP
All		48.6	34.7	40.5
1, 2, 3	Character score	$+1.5$	-1.6	-0.0
4, 5, 6	Character similarity	-5.4	-5.1	-4.3
7, 8, 9	Word score	$+0.3$	-1.3	$+0.3$
10, 11, 12	Word similarity	-4.4	-5.4	-3.2
13, 14	Character bigram similarity	-0.7	$+0.0$	-0.2
15, 16	Character trigram similarity	$+0.2$	-0.5	$+0.3$
18, 19, 20	IPA levenshtein distance	$+1.8$	-1.7	$+0.1$
21, 22, 23	Levenshtein distance	$+2.5$	-0.5	$+0.6$
24, 25	Phonemes	-0.5	-0.6	-0.2
26, 27	Syllables	-1.6	$+0.1$	$+0.7$
28, 29	Length	-0.0	-0.5	$+0.4$
30, 31	Contribution	$+0.0$	$+0.4$	$+0.1$
34, 36	Corpus frequency	± 0.0	-0.0	$+0.0$
35, 37	Affix frequency	-2.7	-8.7	-5.4

Table 17: Feature ablation experiments with categories of features removed. Numbers indicate the performance change in percentage points. Changes such as -0.0 or $+0.0$ indicate that the change is positive or negative by an amount ($0.01 > c$) and ± 0 indicate no change.

	OVERLAP	NOVERLAP	ALL
CATEGORY	MAP	MRR	MAP
All	48.6	34.7	40.5
SEMANTICS	-8.2	-11.9	-9.8
SOURCE WORDS	-2.7	-1.0	-11.2
BLENDS	-7.0	-4.3	-4.7
FREQUENCY	-2.7	-8.7	-5.4

From the first experiment, it can be seen that generally, the changes appear to be rather small. The most notable performance changes are for character similarity, word similarity, and affix frequency. These features show performance losses between 2.7 percentage points and up to 8.7 percentage points.

The largest positive changes can be seen in the Levenshtein categories, where the removal of these features show an increase of 1.8 and 2.5 percentage points for overlapping blends. For noverlapping blends, however, the removal of IPA Levenshtein distance resulted in a loss of 1.7 percentage points.

In the second experiment, complete categories are removed (as described in section 4.2.2.) and the change in performance is measured. The results from this experiment are shown in Table 17.

The results from the second experiment show that the SEMANTIC features seem to have the most importance for overlapping and noverlapping blends, while the SOURCE WORD features seem to be the most important for the complete dataset. The FREQUENCY and BLEND features show a similar performance loss for the complete dataset. With the FREQUENCY features showing a larger loss for noverlapping blends and the blend features showing a larger loss for overlapping features.

In the third experiment, feature ablation is performed on the features individually. The results are shown in Table 18.

Table 18: Feature ablation experiments with features removed individually. Numbers indicate the performance change in percentage points. Changes such as -0.0 or $+0.0$ indicate that the change is positive or negative by an amount ($0.01 > c$) and ± 0 indicate no change.

		OVERLAP	NOVERLAP	ALL
ID	FEATURE	MAP	MRR	MAP
	Full featureset	48.6	34.7	40.5
1	sw_1 character score	± 0.0	± 0.0	± 0.0
2	sw_2 character score	$+1.0$	-0.6	$+0.1$
3	*blend* character score	-0.3	$+0.0$	$+0.3$
4	sw_1, sw_2 character similarity	-3.6	-5.1	-3.9
5	sw_2, *blend* character similarity	-0.1	-1.6	$+0.4$
6	sw_1, *blend* character similarity	-0.6	-1.1	$+0.2$
7	sw_1 word score	$+0.2$	-0.1	$+0.2$
8	sw_2 word score	$+0.1$	-0.7	$+0.1$
9	*blend* word score	$+0.3$	$+0.0$	$+0.0$
10	sw_1, sw_2 word similarity	-2.2	-5.4	-2.2
11	sw_2, *blend* word similarity	-0.6	-0.2	-0.0
12	sw_1, *blend* word similarity	-2.7	-0.0	-0.0
13	sw_1, sw_2 character bigram similarity	$+0.3$	$+0.0$	-0.1
14	sw_2, sw_1 character bigram similarity	-0.8	$+0.2$	-0.0
15	sw_1, sw_2 character trigram similarity	$+0.3$	-0.0	$+0.4$
16	sw_2, sw_1 character trigram similarity	-0.0	-0.0	$+0.2$
17	sw_1, sw_2 LCS	$+0.3$	$+0.0$	$+0.0$
18	sw_1, sw_2 IPA levenshtein distance	$+1.9$	-0.0	$+0.2$
19	sw_1, *blend* IPA levenshtein distance	$+2.5$	$+0.0$	$+0.7$
20	sw_2, *blend* IPA levenshtein distance	$+2.2$	-0.2	$+0.2$
21	sw_1, sw_2 levenshtein distance	$+2.3$	-0.1	$+1.2$
22	sw_1, *blend* levenshtein distance	$+3.5$	-1.1	$+0.1$
23	sw_2, *blend* levenshtein distance	$+0.2$	-1.3	$+0.5$
24	sw_1 phonemes	-0.5	-0.8	-0.1
25	sw_2 phonemes	$+0.0$	-0.3	$+0.1$
26	sw_1 syllables	-0.1	-0.9	$+0.0$
27	sw_2 syllables	-1.9	$+0.8$	$+0.1$
28	sw_1 length	$+0.3$	-0.4	$+0.6$
29	sw_2 length	-0.2	-0.0	-0.0
30	sw_1 contribution	-0.4	$+0.2$	-0.0
31	sw_2 contribution	-0.3	$+0.1$	-0.0
32	sw_1, sw_2 removal	-0.1	-0.1	$+0.5$
33	source word splits	-1.2	$+0.1$	-0.7
34	sw_1 corpus frequency	± 0.0	-0.0	-0.0
35	sw_1 affix frequency	-0.4	-6.8	-2.4
36	sw_2 corpus frequency	-0.0	-0.0	± 0.0
37	sw_2 affix frequency	-5.7	-4.5	-2.7

6 Discussion

6.1 Swedish lexical blends

6.1.1 Statistical properties

The experiment performed in Section 5.1. aims to answer the first set of research questions. The tests reveal that for character, phoneme and syllable lengths, the first source word is significantly shorter than the second source word.

When measuring the contribution to the blend from the source words the t-test shows that the second source words contribute significantly more than the first source word. Exploring this further, it can be seen that the opposite relationship holds, namely that the longer source word contributes significantly more to the blend. This is the opposite of the findings in (Gries, 2004a). The cause of this can most likely be explained by the fact that many Swedish blends consist of compounds, which generally tends to be longer than non-compound words.

In summary, it can be shown that Swedish blends tend to show the same properties as English blends, except for the contribution to the blend from the source words. This would imply that the process generally is the same in the two languages.

6.1.2 Source word Part-of-Speech

The majority of the source words tend to be nouns or verbs. However, some blends consist of rather unexpected word classes. For example, the blend *henniska* is composed of the pronoun *hen*[17] ('he'/'she') and the noun *människa* ('human'). To the author's knowledge, no previous blends have been found which contain pronouns. Another interesting blend is *varsågodling* ('please-take plantation'). This blend is formed by combining *odling* ('plantation') with the interjection *varsågod* ('please take').

These two examples show that when searching for source word candidates, no word class should be excluded.

6.1.3 Frequent source words in lexical blends

In the list of all blends, most source words appear only once. There are exceptions, Table 19 shows the frequency distribution of the types for the source words.

Table 19: Frequency of source word types in the list of all blends.

FREQUENCY	WORDS
1	337
2	30
3	7
4	4
5	2

In (Östling, 2010) Swedish compounds are described in term of different constructions. For example, compounds using the word *kyrka* ('church') tend to either specify the material the church was built with, e.g. *sten-kyrka* ('stone church') or the area in which it is located, such as *stads-kyrka* ('city church').

A similar pattern can be seen in the blends which are constructed using the same source word. One of the most popular source word, with a frequency of 5 is the pronoun *hen* ('he'/'she'). The blends that are formed from *hen* ('he'/'she') are shown below:

Hen is used to gender-neutralize words. The word for human, e.g. *människa*, contain the substring *män* ('man'). In the blend, this substring has been replaced by *hen* ('he'/'she'). The same pattern can

[17] Gender neutral pronoun, *hen* is used as a replacement for *he* or *she*.

BLEND	CONSTRUCTION
Gudhen	gud[en] ('god') + hen ('he'/'she')
Gudhen	gud[innan] ('goddess') + hen ('he'/'she')
Henniska	he**n** ('he'/'she') + [mä]**n**niska ('human')
Henvän	hen ('he'/'she') + [flick/pojk]vän ('boy/girlfriend')
Henom	he**n** ('he'/'she') + [ho]**n**om ('him')

be observed for *guden* ('god'), where *hen* is appended to the reduced form *gud* ('god')[18], the same operation is performed on the feminine version, *gudinna* ('goddess'), where the ending substring `innan` is replaced with *hen* ('he'/'she').

Two different constructions seems to appear here, both with the same goal, but targeting different domains. First, one use of *hen* ('he'/'she') in blending appears to be the removal of non-neutral gender substrings. This is seen in *henniska* and *henvän*, where *män* ('man') and *pojk/flick* ('boy/girl') have been substituted for *hen* ('he'/'she'). For the blends *gudhen* and *henom*, *hen* ('he'/'she') is not used to replace a string denoting male or female. The usage of *hen* ('he'/'she') as a source word in these blends seems to be as a means to encourage a gender neutral interpretation.

The other word that occurs five times is *promenad* ('stroll'). The blends containing *promenad* ('stroll') is the following:

BLEND	CONSTRUCTION
Promelur	prome[nad] ('stroll') + [tupp]lur 'nap'
Promenut	prome**n**[ad] ('stroll') + [mi]**n**ut ('minute')
Pokenad	Pok**e**[mon] + [prom]**e**nad ('stroll')
Rullenad	rull[stol] ('wheelchair') + [prom]enad ('stroll')
Vovvenad	vovv**e** ('dog') + [prom]**e**nad ('stroll')

Four of these blends seem to indicate the type of promenad, a *pokenad* is a stroll while playing the game Pokemon Go, a *rullenad* is a stroll in a wheelchair, a *vovvenad* is taking a stroll with a dog and a *promelur* is a stroll with a sleeping child in a baby carriage. The other blend, *promenut*, specify the time a walk takes. For example, if it takes 15 minutes to walk from point A to B, it takes 15 *prominuter*. These examples show that there appear to be blend constructions based on semantics.

6.1.4 Blends and other word formation processes

There are similarities between blending and several other word formation processes. Most notably in the list of blends, between blending and compounding/affixation.

In one type of blending, one of the source words is a compound. For the word *äggbanjo* ('egg banjo') = *ägg[skivare]* ('egg slicer') + *banjo* ('banjo'), the first word is a compound. The second part of the compound has been replaced by the word *banjo* ('banjo') in the blend. *Äggbanjo* ('egg banjo') could be seen as a compound, as *äggbanjo* ('egg banjo') was described, it is derived from *äggskivare* ('egg slicer'), not from *ägg* ('egg'). Due to this, this has been interpreted as a blend.

Another similar, but more distinct example is *alfanummer* ('alphanumber'), which is a blend of *alfabet* ('alphabet') and *telefonnummer* ('telephone number'). In this case, the word denotes a telephone number which also contains alphabetical characters. In the blend, the first source word *alfabet* ('alphabet') have been reduced to *alfa*. The second source word is a compound of *telefon* ('telephone') + *nummer* ('number'), where the first part of the compound has been removed.

[18]Technically, *gudhen* contain an overlap at the end, gud**en** + h**en**. Thus, the word could also be interpreted as a blend created by an infix, i.e. *h* is inserted into *guden*.

In this case, it follows from the definition of blends ("two concatenated word in which at least one word is reduced") as *alfabet* has been reduced.

Some blends also share similarities with affixation. One such blend is *utroduktion* ('outroduction') = ***ut*** ('out') + *[in]**t**roduktion)* ('introduction'). It is formed by removing *in-* from *introduktion* ('introduction') and replacing it with *ut-*. The meaning of the morphemes *in-* is 'in' and *ut-* is 'out', the new blend acts as an antonym to *introduktion* ('introduction'). This is much like the examples shown in section 2.1. where different affixes changed the semantic meaning of a word.

These examples show that blends share close similarities with already existing word formation processes.

6.2 Method

6.2.1 Candidate selection

During the manual data collection, certain types of blends have been discarded. Among these blends are infix blends, where a word is blended inside another word, such as in *melodihomosexualen* = *melodi[festiv]alen* ('Swedish qualifier for Eurovision Song Contest') + *homosexu[ell]* ('homosexual'). Another type of blend which has been removed is non prefix-suffix blends, where for example two prefixes have been used as in *mokus* = *mo[r]* ('mother') + *kus[in]* ('cousin').

The problem with these blends for the current system is that allowing them to be included would result in that the candidate selection would have to be more inclusive. For every blend, the candidate search would have to take into account: prefix-suffix combinations, prefix-prefix combinations, suffix-suffix combinations, and infix combinations. This would yield an explosion of possible candidate pairs for each blend. To implement the splitting strategy automatically used, where overlapping and noverlapping blends are treated differently, a classifier which is able to separate overlapping and noverlapping blends would be required.

6.2.2 Quality of the gold standard

Some issues were found during the creation of the gold standard. The issues arise for candidate pairs that fit semantically and orthographically with the blend.

For example, the gold candidate pair for the blend *göteburgare* = *göte**b**[org]* ('Gothenburg') + *[ham]**b**urgare* ('hamburger') is *(göteborg, hamburgare)*. But similar and apparently correct word pairs were found, such as *(göteborg, burgare)* ('Gothenburg', 'burger') where *burgare* ('burger') is a clipping of *hamburgare*. Another candidate pair that could be considered correct is *(göteborgsk, hamburgare)* ('Gothenburgian', 'hamburger'), where *göteborgsk* ('Gothenburgian') is a derivation of *göteborg* ('Gothenburg') with the suffix *-sk*. The derivation causes *Göteborg* ('Gothenburg') to change part-of-speech from a proper noun to an adjective (meaning roughly 'from Göteborg'). Thus, semantically *(göteborgsk, hamburgare)* would have the same meaning as *(göteborg, hamburgare)*. Ideally, the pairs *(göteborg, burgare)* and *(göteborgsk, hamburgare)* should also be considered correct.

A related example comes from the blend *promelur* = *prome[nad]* ('stroll') + *[tupp]lur* ('light nap'). The second source word, *tupplur* ('light nap'), is a compound formed from *tupp* + *lur* ('light nap'). The meaning of *lur* is 'light sleep'[19], the same is true for *tupplur* ('light nap'). The description of the blend mention *tupplur* ('light nap') and not *lur* ('light nap'), thus only *tupplur* ('light nap') was used in the gold standard.

Blends occur rarely and are often described briefly: either by a description of its meaning or through the words used in its creation. Finding all the correct word pairs have not been done due to the scope of the thesis, but it should be noted that more than one candidate word pair could be considered as correct.

[19] https://svenska.se/saob/?id=L_1039-0372.3M4Q&pz=5 accessed 2018-07-25

6.2.3 Evaluation of embedding models

Both character and word embeddings are used in the model to represent words. What has not been done is to evaluate these models externally to see how well they actually capture the meanings of words. For English, several word analogy tasks and standardized datasets exist for evaluating embeddings. For Swedish on the other hand, there are none to the author's knowledge. As such, it cannot be determined how well the embedding models used actually capture the semantic relation between words.

It is shown in the feature ablation (Section 5.2.3.) that the similarity between the source words play an important role in the model, which may be considered as a downstream evaluation metric indicating that the embeddings do capture word similarities.

6.2.4 Machine learning model

Only logistic regression is tested which may be considered a weakness. Using a variety of machine learning algorithms would have been preferred, as this would allow an evaluation of how different learning approaches influence the source word prediction.

6.2.5 Left-out features

One common type of language feature has not been used in this thesis, namely word *n*-grams from corpora. It was found that quite a few of the candidate pairs actually has any *n*-grams associated with them in the news corpus. As such, this feature would be null in most cases and was discarded early in development.

6.3 Results

6.3.1 Ranking experiments

In the ranking experiments, the logistic regression model is compared to the random and feature ranking baseline. The evaluation is performed by measuring the accuracy at different ranking ranges.

The random baseline showed a poor performance. For the highest ranking word pair, a correct word pair was only found in 2-3% of the cases. The performance of the different datasets was generally the same. Observing larger ranges, it is found that the results tend to improve. The best performance was observed for the overlapping blends. When selecting 10 word pairs at random, a correct word pair is found in 9-15% of the cases.

The feature ranking baseline generally performs better than the random baseline. For the highest ranking word pair, there is only a small difference compared to the random baseline for the noverlapping blends and the combined dataset. For the overlapping blends, however, there is a large difference, where the feature ranking baseline is 15.9 percentage points more accurate. The accuracy in the larger ranges generally seems to increase more than for the random baseline, where at the top 10 ranking word pairs the correct word pair is found in 16-42% of the cases. As with the random baseline, the performance on the overlapping blends is much higher than on the noverlapping blends and the combined datasets.

The logistic regression model outperforms both baselines on all ranges. For the highest ranking word pairs, a correct word pair is found in 23-44% of the cases. For the largest range, the top 10 ranking word pairs a correct word pair is found in 54-74% of the cases. This is a vast improvement for all datasets. As before, the model performs best on the overlapping blends, followed by the combined dataset, the worst performance is again observed for the noverlapping blends.

In summary, it appears as if the overlapping dataset is easier to predict than the noverlapping and combined dataset. In part, this is likely due to the overlapping blends having fewer candidate pairs.

In comparison to the model by Cook and Stevenson (2010), the current results for the combined dataset are 7.8 percentage points lower. As the dataset used by Cook and Stevenson (2010) is roughly two times the size of the combined dataset used in this thesis, this is not completely unexpected. When comparing the feature ranking baseline, which is based on (Cook and Stevenson, 2010), against the model at the top ranking word pair, the model's performance is 25.3 percentage points higher.

It should be noted that the current features used is not the same as in (Cook and Stevenson, 2010). The model of Cook and Stevenson (2010) relies on features which give the correct word pairs a higher score than the incorrect word pairs. This has not been a constraint during the development of the current feature set.

6.3.2 Classification experiments

The classification experiment evaluates the performance of the top three and five retrieved word pairs. The result from the classification experiment is an estimate of the system's performance in an external application. It shows the fraction of correct word pairs compared to incorrect word pairs in the top n retrieved word pairs (precision) and the fraction of correct word pairs in the top *n* retrieved word pairs, compared to the number of correct word pairs not in the top *n* retrieved word pairs (recall).

In an implementation part of a pipeline, the system would ideally retrieve some number of candidate word pairs to choose from. The precision will thus tell us how often the system would be correct if a word pair would be chosen at random from the retrieved word pairs. It can be seen that the precision generally is rather low, but higher for the top three retrieved word pairs compared to the top five retrieved word pairs. This is mainly due to the inevitably retrieved incorrect word pairs (as described in Section 5.2.2.).

The system's performance is also estimated by the normalized score, which takes the upper bound into account. This score indicates the intrinsic performance of the system, which is much higher than the extrinsic performance.

6.3.3 Feature ablation

Feature categories: The feature ablation for categories of features show that the most important category for overlapping and noverlapping blends is SEMANTICS. For the combined dataset, the SEMANTIC features showed a comparable performance loss, and the SOURCE WORD features showed the highest loss. The SOURCE WORD category showed the lowest performance loss for the overlapping and noverlapping blends, which is surprising.

This would suggest that when both overlapping and noverlapping blends are considered, the relationship between the source words, ignoring the blend itself, plays the most important role.

Feature groups: The feature ablation for the groups of feature shows that the highest loss for the overlapping blends is the feature group *character similarity*. For noverlapping blends, the feature group *affix frequency* shows the highest loss. The *affix frequency* group also showed the highest loss for the combined dataset.

For both the noverlapping and complete dataset, *character similarity* also showed a rather high performance loss, indicating that the feature is important for all types of blends. It can also be observed that the *word similarity* group showed a high loss of accuracy. The *affix frequency* group did show a loss for overlapping blends, ranked third, also suggesting it is not a useless feature for these blends. However, the performance loss is much lower than for noverlapping blends and the complete dataset.

In summary, the *word/character similarity* and *affix frequency* feature groups appear to be the most important features.

Individual features: The feature ablation for individual features shows that for overlapping blends, the changes tend to be minor. The features with the highest negative performance change for the overlapping blends are summarized below:

From Table 20 it can be seen that the same features seem to produce the highest losses, namely 37 (sw_2 affix frequency), 35 (sw_1 affix frequency), 10 (sw_1, sw_2 word similarity) and 4 (sw_1, sw_2 character similarity).

Regarding the semantic similarity features, it appears as if the most informative features are the similarity between the two source words, and not between the source words and the blend. There is only one case in which the similarity between the source words and the blend seems to matter. For overlap-

Table 20: Individual features which showed the largest performance loss during the feature ablation experiment.

Overlapping		Noverlapping		Combined	
Id	Feature	Id	Feature	Id	Feature
37	−5.7	35	−6.8	4	−3.9
4	−3.6	10	−5.4	37	−2.7
12	−2.7	4	−5.1	35	−2.4
10	−2.2	37	−4.5	10	−2.2

ping blends, feature 12 ($sw_1, blend$ word similarity) showed a performance loss of 2.7 percentage points.

The affix frequency feature showed a high performance loss, especially for the noverlapping blends. This shows that the frequency of the source words is important, and primarily when compared with other possible word pairs since the corpus frequency feature barely has any impact. The affix frequency feature is the only feature which considers the relationship between the current word pair and the other candidate word pairs. Constructing additional features with the same principle would be beneficial given that the number of word pairs may be quite large.

Some features show a performance increase when removed. Most notably, all the Levenshtein distance features show an performance increase of 1.9 to 3.5 percentage points for the overlapping blends. For the noverlapping and combined datasets, the Levenshtein features only resulted in minor performance changes (generally negative for the noverlapping blends and positive for the complete dataset). The Levenshtein distance would encode several aspects of structural similarity, both the parts removed and the overlapping parts. One cause why this feature show performance gains for the overlapping blends may be because there are fewer unlikely word pairs. Since it is assumed there must be two or more ways of combining the source words in an overlapping blend, the Levenshtein distance between source words would be smaller. Without the constraint that the source words create the blend in two or more ways, more word pairs would be considered as candidates. This could allow the Levenshtein feature to exclude bad candidate pairs, but the pairs excluded would most likely be the ones that were removed by the constraint.

7 Conclusions

This section will present the conclusions in regards to the research questions, and propose some directions for future work and research.

Regarding research question (1.1) there is a relationship between the source words in terms of orthographic and phonetic length (Section 5.1.1), where the first source word is significantly shorter. Concerning research question (1.2) the first source word is significantly more frequent in the reference corpus (Section 5.1.3). For research question (1.3) the second source word contributes significantly more to the lexical blend compared to the first source word, and that the longer source word does contribute more to the lexical blend than the shorter source word (Section 5.1.2.).

To investigate the importance of different features a set of feature ablation experiments were performed. Regarding research question (2.1) the features relating to the semantics and frequency were the best predictors (Section 5.2.3). In regards to research question (2.2) it is easier to predict the source words of overlapping blends than noverlapping blends (Section 5.2.1 and 5.2.2.), For research question (2.3), the affix frequency features are more important or noverlapping blends than for overlapping blends. The Levenshtein distance feature is more informative for noverlapping blends than for overlapping blends (Section 5.2.3.)

The results are promising given the small dataset. It is encouraging that the only language-dependent features are the resources, which means that the feature set should be applicable to English (where a much larger dataset of lexical blends can be found). Improvements to the model will focus on re-thinking the orthographic and phonetic features used and incorporating more features that take into account the other candidate word pairs and finding the more prominent word pairs (e.g. features similar to the affix frequency feature, as discussed in Section 6.3.3.). The candidate selection will also be refined, by finding a more efficient and accurate method of selecting word pairs, for example by splitting the blend based on syllables and/or morphemes. The discussion focused on the quality of the gold standard, one improvement that will be done is including more correct word pairs in the gold standard.

References

Baayen, R. H., Piepenbrock, R., and van H, R. (1993). The CELEX lexical database on cd-rom.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Bolander, M. (2005). *Funktionell svensk grammatik*. Liber.

Borin, L., Forsberg, M., and Lönngren, L. (2008). Saldo 1.0 (svenskt associationslexikon version 2). *Språkbanken, University of Gothenburg*.

Brants, T. and Franz, A. (2006). Web 1T 5-gram version 1.

Cook, C. P. (2010). *Exploiting linguistic knowledge to infer properties of neologisms*. University of Toronto.

Cook, P. (2012). Using social media to find english lexical blends. In *Proc. of EURALEX*, pages 846–854.

Cook, P. and Stevenson, S. (2007). Automagically inferring the source words of lexical blends. In *Proceedings of the Tenth Conference of the Pacific Association for Computational Linguistics (PACLING-2007)*, pages 289–297.

Cook, P. and Stevenson, S. (2010). Automatically identifying the source words of lexical blends in English. *Computational Linguistics*, 36(1):129–149.

Firth, J. R. (1961). *Papers in Linguistics 1934-1951: Repr*. Oxford University Press.

Gangal, V., Jhamtani, H., Neubig, G., Hovy, E., and Nyberg, E. (2017). Charmanteau: Character embedding models for portmanteau creation. *arXiv preprint arXiv:1707.01176*.

Grave, E., Bojanowski, P., Gupta, P., Joulin, A., and Mikolov, T. (2018). Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Gries, S. T. (2004a). Isn't that fantabulous? How similarity motivates intentional morphological blends in english. *Language, culture, and mind*, pages 415–428.

Gries, S. T. (2004b). Shouldn't it be breakfunch? A quantitative analysis of blend structure in English. *Linguistics*, pages 639–668.

Gries, S. T. (2012). Quantitative corpus data on blend formation: Psycho-and cognitive-linguistic perspectives. *Cross-disciplinary perspectives on lexical blending*, 252:145.

Jurafsky, D. and Martin, J. H. (2009). *Speech and language processing*, volume 2. Pearson London.

Lehrer, A. (2007). Blendalicious. *Lexical creativity, texts and contexts*, 19:115–136.

Mattiello, E. (2013). *Extra-grammatical morphology in English: abbreviations, blends, reduplicatives, and related phenomena*, volume 82. Walter de Gruyter.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mortensen, D. R., Dalmia, S., and Littell, P. (2018). Epitran: Precision G2P for many languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA).

Östling, R. (2010). A construction grammar method for disambiguating swedish compounds. In *SLTC 2010 Workshop on Compounds and Multiword Expressions*.

Plag, I. (2003). *Word-Formation in English*. Cambridge Textbooks in Linguistics. Cambridge University Press.

Renner, V. (2015). Lexical blending as wordplay. *Wordplay and Metalinguistic/Metadiscursive Reflection: Authors, Contexts, Techniques, and Meta-Reflection*, 1:119.

Ronneberger-Sibold, E. (2012). Blending between grammar and universal cognitive principles: Evidence from German, Farsi, and Chinese. *Cross-disciplinary perspectives on lexical blending*, 252:115.

Schicchi, D. and Pilato, G. (2017). Wordy: A semi-automatic methodology aimed at the creation of neologisms based on a semantic network and blending devices. In *Conference on Complex, Intelligent, and Software Intensive Systems*, pages 236–248. Springer.

Schütze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press.

Sjöbergh, J. and Kann, V. (2004). Finding the correct interpretation of swedish compounds, a statistical approach. In *4th International Conference on Language Resources and Evaluation. Lissabon. S. 899–902*.

8 Appendix A: Lexical Blends

Table 21: The complete list of Swedish lexical blends. FREQ. indicates the relative frequency of the word in the corpus (described in Section 4.1.1.)

BLEND	SW_1	FREQ.	SW_2	FREQ.
aftonhora	aftonbladet	4.76e-05	hora	5.62e-06
äggbanjo	äggskivare	NA	banjo	9.62e-07
alfanummer	alfabet	1.55e-06	telefonnummer	9.66e-06
alkobom	alkohol	5.86e-05	bom	1.60e-05
alkolektuell	alkohol	5.86e-05	intellektuell	2.41e-05
alkoläsk	alkohol	5.86e-05	läsk	6.65e-06
automagisk	automatisk	2.91e-05	magisk	2.01e-05
badfota	bada	1.73e-05	barfota	2.54e-09
bådhänt	båda	0.00011	högerhänt	4.32e-07
bajskoma	bajs	1.53e-06	matkoma	NA
brunka	bajs	1.53e-06	runka	2.24e-07
bakfräsch	bakfull	7.66e-07	fräsch	2.00e-05
bakmätt	bakfull	7.66e-07	mätt	1.87e-05
baksnygg	bakfull	7.66e-07	snygg	6.58e-05
bankomat	bank	0.00025	automat	5.39e-06
bessergissa	besserwisser	6.44e-07	gissa	2.13e-05
bionik	bio	2.71e-05	elektronik	7.36e-06
blågon	blåbär	3.97e-06	lingon	3.67e-06
blingon	blåbär	3.97e-06	lingon	3.67e-06
blok	blogg	3.71e-05	bok	0.0004
bloggbävning	blogg	3.71e-05	jordbävning	1.97e-05
bloppis	blogg	3.71e-05	loppis	2.48e-06
blorange	blond	8.42e-06	orange	1.60e-05
bokig	bok	0.0004	tokig	7.78e-06
bombasm	bombast	NA	sarkasm	8.09e-07
bollywood	bombay	NA	hollywood	1.67e-05
brilljanne	brilliant	3.99e-07	janne	3.00e-05
brittsommar	brittisk	0.00018	sommar	0.00026
brony	bror	9.52e-05	pony	NA
bromantik	bror	9.52e-05	romantik	5.71e-06
brokabulär	bror	9.52e-05	vokabulär	1.37e-06
burkini	burka	2.87e-06	bikini	1.61e-06
chattityd	chatt	6.88e-06	attityd	3.84e-05
cinematek	cinema	NA	bibliotek	4.79e-05
cyborgmässoafton	cyborg	2.01e-07	valborgsmässoafton	1.82e-06
dansett	dans	5.20e-05	balett	7.14e-06

danseoke	dans	5.20e-05	karaoke	1.01e-06
datalektiker	data	1.92e-05	dyslektiker	8.09e-07
diagrammatisk	diagram	2.26e-06	grammatisk	8.65e-07
diamantmunk	diamantbröllop	7.38e-08	munk	8.51e-06
diskotek	disko	NA	bibliotek	4.79e-05
doktorskop	doktor	2.45e-05	stetoskop	2.19e-07
dokudrama	dokumentär	3.25e-05	drama	4.87e-05
dricknödig	dricka	7.56e-05	kissnödig	6.01e-07
drinkorexi	drink	8.70e-06	anorexi	1.61e-06
dumska	dumhet	4.52e-06	ondska	1.12e-05
duschnödig	duscha	4.46e-06	kissnödig	6.01e-07
efterföljarskap	efterföljare	5.52e-06	ledarskap	2.80e-05
ekodukt	ekologisk	3.16e-05	produkt	0.00011
epost	elektronisk	2.75e-05	post	8.63e-05
ensams	ensam	0.00014	sams	1.03e-06
eurabien	europa	0.00031	arabien	5.06e-07
eurasien	europa	0.00031	asien	4.50e-05
exbo	ex	5.46e-06	sambo	4.00e-05
facenuft	facebook	4.01e-05	förnuft	1.17e-05
fansin	fan	6.22e-05	magasin	1.39e-05
farlug	farlig	0.00011	slug	1.99e-06
fastlans	fast	0.00034	frilans	1.26e-06
fejkon	fejk	NA	bacon	6.11e-06
feltema	fel	0.00022	biltema	NA
feminazi	feminist	9.59e-06	nazi	NA
fjanig	fjant	2.54e-07	mesig	2.61e-06
fjollåtta	fjolla	3.31e-07	nollåtta	NA
fjunjacka	fjun	1.73e-07	dunjacka	NA
flerbo	flera	2.55e-06	sambo	4.00e-05
flextid	flexibel	1.57e-05	arbetstid	2.70e-05
flexuell	flexibel	1.57e-05	sexuell	8.35e-05
flexidaritet	flexibel	1.57e-05	solidaritet	1.55e-05
flexiterian	flexibel	1.57e-05	vegetarian	3.27e-06
folkjanne	folköl	2.01e-06	janne	3.00e-05
folknäsa	folköl	2.01e-06	näsa	2.05e-05
försovmorgon	försova	NA	sovmorgon	9.75e-07
fotjuice	fotsvett	1.22e-07	juice	4.45e-06
frångänlighet	från	0.00467	tillgänglighet	1.71e-05
fredagsfys	fredagsmys	1.10e-06	fys	NA
frimester	fri	0.00025	semester	4.91e-05
friggebod	friggebo	NA	bod	4.20e-06
frolf	frisbee	8.25e-07	golf	4.22e-05

frunch	frukost	1.81e-05	lunch	3.39e-05
funtionsuppsättning	funktionsnedsättning	4.59e-06	uppsättning	2.25e-05
fyllosof	fyllo	6.67e-07	filosof	1.22e-05
gåggning	gå	0.00292	joggning	3.89e-07
geoblockering	geografisk	1.59e-05	blockering	1.65e-06
ghettoborg	ghetto	1.12e-06	göteborg	0.00106
givomat	giv	3.26e-06	automat	5.39e-06
glädjemätare	glädje	5.70e-05	hastighetsmätare	9.52e-07
glassfluss	glass	1.51e-05	halsfluss	9.32e-07
glokal	global	0.00011	lokal	0.00028
göteburgare	göteborg	0.00106	hamburgare	6.93e-06
gnaska	gott	0.00011	snaska	1.22e-07
grannmor	granne	7.77e-05	mormor	2.01e-05
grunka	gråta	3.19e-05	runka	2.24e-07
grotal	grotesk	4.84e-06	brutal	3.20e-05
gudhen	guden	NA	hen	3.20e-06
gudhen	gudinna	1.73e-06	hen	3.20e-06
guldomat	guld	9.85e-05	automat	5.39e-06
guldmunk	guldbröllop	1.42e-07	munk	8.51e-06
hackaton	hacka	3.25e-05	maraton	5.58e-06
helgbo	helg	0.00019	sambo	4.00e-05
heliport	helikopter	4.65e-05	port	1.90e-05
hemester	hem	0.00041	semester	4.91e-05
henom	hen	3.20e-06	honom	NA
henniska	hen	3.20e-06	människa	0.00083
henvän	hen	3.20e-06	pojkvän	1.96e-05
höronpropp	höra	0.00042	öronpropp	1.20e-06
hotivera	hot	0.00015	motivera	3.79e-05
hydrokopter	hydro	NA	helikopter	4.65e-05
idiolekt	idiom	8.07e-07	dialekt	5.79e-06
igårse	igår	NA	morse	4.86e-07
inrymma	in	0.00199	utrymma	1.01e-05
informatik	information	0.00019	teknik	0.0001
iprensa	ipren	NA	influensa	1.07e-05
danskhjälm	dansk	0.00019	jethjälm	2.54e-09
kallingdrom	kallingar	0.0	syndrom	5.54e-06
kattikett	katt	2.55e-05	etikett	8.05e-06
keffe	keff	9.42e-08	kaffe	4.45e-05
killräcklig	kille	0.00012	tillräcklig	0.00015
klicktivism	klick	7.59e-07	aktivism	1.35e-06
knappra	knappa	NA	smattra	2.61e-06
knarkometer	knark	7.35e-06	termometer	3.21e-06

knaffel	kniv	4.04e-05	gaffel	3.44e-06
kollektomat	kollekt	6.11e-07	automat	5.39e-06
kontronym	kontra	1.71e-05	synonym	5.22e-06
köttrymden	kött	5.87e-05	cyberrymden	0.0
kräklig	kräkas	4.12e-06	äcklig	6.01e-06
krunka	kräkas	4.12e-06	runka	2.24e-07
kussas	kram	7.42e-06	pussas	7.61e-07
krashisera	krasch	1.57e-05	kritisera	0.0001
labradoodle	labrador	1.19e-06	poodle	NA
långtrevlig	långtråkig	1.63e-06	trevlig	5.53e-05
lökburgare	lökig	1.09e-07	hamburgare	6.93e-06
mationera	mat	0.00019	motionera	6.02e-06
medskam	medlidande	1.71e-06	skam	1.69e-05
minusmeny	minus	1.16e-05	plusmeny	NA
mizeria	misär	5.22e-06	pizzeria	5.94e-06
mombie	mobil	0.0001	zombie	2.32e-06
motell	motor	5.54e-05	hotell	8.98e-05
nunch	natt	0.00026	lunch	3.39e-05
oftsynt	ofta	0.00052	sällsynt	2.25e-05
obror	okamratlig	5.34e-08	bror	9.52e-05
omblera	om	0.0088	möblera	5.05e-06
opkonst	optisk	1.96e-06	konst	9.27e-05
otroligg	otrolig	8.35e-05	ligg	4.55e-07
paddarm	padda	2.42e-06	musarm	2.36e-07
pappografi	pappa	0.00018	mammografi	2.72e-06
parasömn	paradoxal	1.02e-05	sömn	1.37e-05
permafrost	permanent	2.89e-05	frost	6.55e-06
platinummunk	platinumbröllop	NA	munk	8.51e-06
plogga	plocka	9.62e-05	jogga	5.25e-06
pokenad	pokemon	NA	promenad	2.26e-05
popinion	populärvetenskaplig	1.20e-06	opinion	2.66e-05
prokott	pro	NA	bojkott	9.26e-06
promenut	promenad	2.26e-05	minut	0.00045
promelur	promenad	2.26e-05	tupplur	9.60e-07
pysselsättning	pyssel	1.32e-06	sysselsättning	4.05e-05
råkelse	råka	5.59e-05	händelse	0.00018
renovräkning	renovering	1.51e-05	räkning	3.17e-05
rullist	rullstol	8.18e-06	cyklist	2.71e-05
rullenad	rullstol	8.18e-06	promenad	2.26e-05
slimpa	semla	3.65e-06	limpa	1.90e-06
semoothie	semla	3.65e-06	smoothie	NA
sextremism	sex	0.00044	extremism	4.80e-06

skiffla	skaffa	8.19e-05	fiffla	1.62e-06
skärmarbronx	skärmarbrink	NA	bronx	NA
skaffel	sked	3.39e-06	gaffel	3.44e-06
skråta	skratta	6.91e-05	gråta	3.19e-05
skypebo	skype	4.84e-06	sambo	4.00e-05
slangopedia	slang	4.82e-06	wikipedia	3.45e-06
slappityd	slapp	4.85e-06	attityd	3.84e-05
slunkig	sliskig	6.77e-07	sunkig	1.98e-06
slucko	slusk	1.12e-07	pucko	4.35e-07
slynda	slyna	1.32e-07	hynda	5.85e-08
smulgubbe	smultron	9.42e-07	jordgubbe	7.93e-06
smög	smyg	3.48e-06	bög	4.67e-06
snällskap	snäll	3.58e-05	sällskap	5.17e-05
snimp	snus	1.05e-05	fimp	1.06e-06
snajs	snygg	6.58e-05	najs	NA
snyrra	snygg	6.58e-05	syrra	1.72e-06
sociolekt	social	0.00016	dialekt	5.79e-06
spoodle	spaniel	3.26e-07	poodle	NA
spyke	spöke	7.09e-06	psyke	5.00e-06
ståhjuling	stå	0.00107	enhjuling	2.26e-07
stagflation	stagnation	2.00e-06	inflation	4.74e-05
stjärtnapp	stjärtlapp	NA	napp	3.70e-06
stressenär	stress	2.74e-05	resenär	4.81e-05
stulgran	stulen	NA	julgran	4.21e-06
styggsurf	stygg	1.05e-06	porrsurf	NA
sugrörsseende	sugrör	7.48e-07	tunnelseende	6.72e-07
sväron	svärförälder	1.42e-06	päron	4.66e-06
svenhippa	svensexa	8.02e-07	möhippa	5.85e-07
svorska	svenska	8.62e-05	norska	6.89e-06
svemester	sverige	0.00163	semester	4.91e-05
svegan	sverige	0.00163	vegan	1.44e-06
sympartisk	sympatisk	9.22e-06	partisk	1.80e-06
teknostress	teknologi	9.37e-06	stress	2.74e-05
telematik	telekommunikation	1.86e-06	informatik	NA
tourätta	tourettsyndrom	NA	rätta	2.41e-05
trånglig	trång	2.70e-05	krånglig	1.44e-05
tummträ	tummstock	NA	trä	1.65e-05
tvodd	tv	0.00019	podd	NA
tvångest	tvång	1.17e-05	ångest	1.85e-05
tvocial	tvång	1.17e-05	social	0.00016
tvärdelös	tvär	1.16e-05	värdelös	7.64e-06
tvångla	tvinga	0.00034	hångla	1.62e-06

tvingla	tvinga	0.00034	mingla	2.71e-06
underraskning	under	0.00289	överraskning	3.94e-05
utroduktion	ut	0.00219	introduktion	8.56e-06
utvigning	ut	0.00219	invigning	1.65e-05
vobba	vabba	3.08e-07	jobba	0.00036
våldtäktsbladet	våldtäkt	6.02e-05	aftonbladet	4.76e-05
vanlis	vanlig	0.00039	kändis	1.67e-05
väntkorv	vänta	0.00062	varmkorv	NA
vårdotek	vård	0.00015	apotek	3.15e-05
vårdvisare	vårdgivare	7.05e-06	visare	6.49e-07
varsågodling	varsågod	NA	odling	1.15e-05
varsegodis	varsegod	NA	godis	1.33e-05
vlogg	video	2.85e-05	logg	1.00e-06
visukal	visuell	1.08e-05	musikal	1.69e-05
vovvenad	vovve	9.47e-07	promenad	2.26e-05
wikipetter	wikipedia	3.45e-06	petter	3.39e-05
xyllo	xylocaingel	NA	lyllo	NA
zudel	zucchini	1.79e-06	nudlar	0.0

Stockholms universitet/Stockholm University
SE-106 91 Stockholm
Telefon 08 - 16 20 00
www.su.se

Stockholms universitet