Concept-Map-Based Multi-Document Summarization using Concept Coreference Resolution and Global Importance Optimization

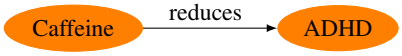
Tobias Falke, Christian M. Meyer, Iryna Gurevych
Research Training Group AIPHES and UKP Lab
Department of Computer Science, Technische Universität Darmstadt
https://www.aiphes.tu-darmstadt.de

Abstract

Concept-map-based multi-document summarization is a variant of traditional summarization that produces structured summaries in the form of concept maps. In this work, we propose a new model[1] for the task that addresses several issues in previous methods. It learns to identify and merge coreferent concepts to reduce redundancy, determines their importance with a strong supervised model and finds an optimal summary concept map via integer linear programming. It is also computationally more efficient than previous methods, allowing us to summarize larger document sets. We evaluate the model on two datasets, finding that it outperforms several approaches from previous work.

1 Introduction

Concept-map-based multi-document summarization (MDS) is a variant of traditional MDS that produces structured summaries in the form of a concept map instead of a coherent text (Falke and Gurevych, 2017a). A *concept map*, introduced by Novak and Gowin (1984), is a labeled graph showing *concepts* as nodes and *relations* between them as edges. As an example, consider a document collection discussing treatments for ADHD. A (very) small concept map would be



in which *Caffeine* and *ADHD* are concepts, while *reduces* is a relation, forming the *proposition "Caffeine – reduces – ADHD"*.

A summary in this form has interesting applications, as it provides a concise overview of a document collection, structures it across document boundaries and can be used as a table-of-contents to navigate in the collection. Several studies report successful applications of concept maps in this direction (Carvalho et al., 2001; Briggs et al., 2004; Richardson and Fox, 2005; Villalon, 2012; Valerio et al., 2012; Falke and Gurevych, 2017b).

The task we consider in this work is defined as follows: *Given a set of documents on a certain topic, extract a concept map that represents the most important content on that topic, satisfies a specified size limit and is connected.*

Although work dealing with the automatic extraction of concept maps from text exists (§2), current methods have several limitations. First, most approaches do not attempt to detect coreferences between extracted concepts. For instance, if both *ADHD symptoms* and *symptoms of ADHD* are found, they treat them as separate concepts. In a concept map, such duplicate concepts are immediately visible to a user, waste valuable space and make it harder to look for relations of that concept, as they are spread among the duplicates.

Second, previous work mostly focused on the extraction of concepts and relations, largely ignoring the subsequent selection step necessary to produce a summary of manageable size. Existing studies suggested only a few unsupervised metrics to determine important elements, leaving it unclear whether the task can benefit from more sophisticated supervised approaches. In addition, no method has been suggested to find an optimal summary concept map under the constraints of the size limit and connectedness.

Third, most approaches for concept map extraction and also traditional summarization are typically evaluated on small document sets where the computational complexity of methods is less relevant. We work on a corpus with sets of around 40 documents that should be summarized, which,

[1] Source code available at https://github.com/UKPLab/ijcnlp2017-cmaps

Proceedings of the The 8th International Joint Conference on Natural Language Processing, pages 801–811,
Taipei, Taiwan, November 27 – December 1, 2017

while being a realistic real-world application scenario, is 10 to 15 times larger than traditional DUC[2] and TAC[3] summarization corpora. This poses an additional challenge that requires the methods to scale to these sizes.

In this work, we propose a new model for concept-map-based MDS that overcomes the aforementioned issues. Building upon previous work in textual summarization, coreference resolution and semantic similarity, it learns to identify and merge coreferent concepts, scores them for importance and finds an optimal summary concept map via integer linear programming (ILP). We also present several optimizations that make it possible to apply our model to large document sets. Experiments on two datasets demonstrate the efficacy of the model, which outperforms several methods suggested in previous work.

2 Related Work

Previous approaches to construct concept maps from text, working with either single documents (Zubrinic et al., 2015; Villalon, 2012; Valerio and Leake, 2006; Kowata et al., 2010) or document clusters (Qasim et al., 2013; Zouaq and Nkambou, 2009; Rajaraman and Tan, 2002), all follow a similar pipeline: concept extraction, relation extraction, scoring and concept map construction.

During concept extraction, most approaches apply hand-written patterns to extract labels for concepts from syntactic representations, focusing on noun phrases-like structures. Similar approaches are used to extract relation labels for pairs of concepts. Alternatively, semantic representations have been suggested as a more easily accessible representation compared to syntax (Falke and Gurevych, 2017c; Olney et al., 2011).

Given these extractions, few attempts beyond string matching have been made to identify unique concepts. Valerio and Leake (2006) suggest to consider only certain part-of-speech during string matching, while the earlier approach of Rajaraman and Tan (2002) uses a clustering algorithm based on a vector space model. Our work proposes a more comprehensive approach, leveraging state-of-the-art semantic similarity measures and set partitioning to also detect coreferent concept labels that are paraphrases.

The selection of a summary-worthy subset of all extracted concepts and relations was largely ignored in previous work, as many studies did not have a focus on summarization. However, when dealing with larger document clusters, this step becomes inevitable. Zubrinic et al. (2015) suggest a tf-idf metric on the level of concept labels, Villalon (2012) uses Latent Semantic Analysis and Valerio and Leake (2006) suggest simple concept frequencies. Our model goes a step further and combines these with other features in a supervised model, which works well for textual summarization (Cao et al., 2016; Yang et al., 2017).

For building a summary concept map that is connected, does not exceed the target size and contains as many important concepts as possible, we are only aware of a heuristic approach suggested by Zubrinic et al. (2015). It iteratively removes low-scoring concepts from all extractions until a connected graph of the target size remains. However, it is not guaranteed that the optimal subset is found. Integer Linear Programming (ILP) has been successfully used to solve the knapsack problem that arises in sentence-level extractive summarization (McDonald, 2007). In our task, the knapsack problem is not present, as both the scoring and size restriction are defined on the level of concepts, but the connectedness requirement poses a similar constraint that restricts the subset selection. ILP formulations for such a problem have been proposed for graph-based abstractive summarization (Li et al., 2016; Liu et al., 2015). In our work, we transfer these ideas to concept maps and evaluate their efficacy. This is important, as the methods were originally proposed for different kinds of graphs (event networks and AMR graphs) and introduced to generate abstractive textual summaries, while we use our concept map graphs directly as the final summaries.

3 Model

Given a document set D, topic t and size limit L, our model applies the three stage approach that is illustrated in Figure 1 to create a summary concept map: (1) Concept and Relation Extraction, (2) Concept Graph Construction, (3) Graph Summarization. We describe these steps in the following sections in detail.

3.1 Concept and Relation Extraction

The goal of the first step is to identify spans in the documents that can be used as labels for concepts

[2] http://duc.nist.gov/
[3] https://tac.nist.gov/

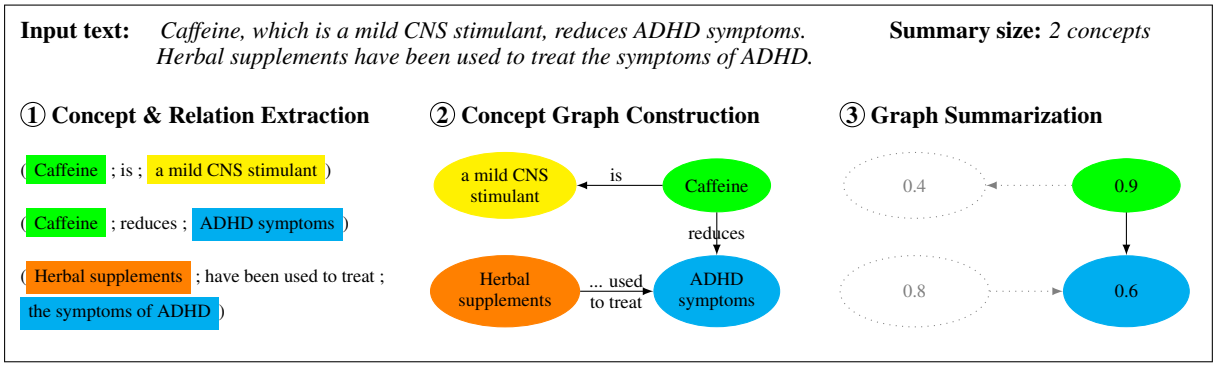


Figure 1: Conceptual illustration of the model: (1) Extracted propositions are (2) connected to a graph based on coreference and (3) the best subgraph, here of target size 2, is selected after scoring concepts.

and relations in the concept map.

Extraction For the extraction, we rely on Open Information Extraction (Banko et al., 2007), an approach that extracts binary *propositions* from text. Given a sentence such as

Caffeine, which is a mild CNS stimulant, reduces ADHD symptoms.

an Open IE system extracts the tuples:

(Caffeine ; is ; a mild CNS stimulant)
(Caffeine ; reduces ; ADHD symptoms)

This representation is particularly useful because it is very similar to propositions in a concept map, requiring only a few postprocessing steps. We use the extracted tuples (m_1, r, m_2), after applying the postprocessing steps discussed below, and use their arguments m_1, m_2 as *concept mentions* and predicates r as *relations*.

Filtering To ensure that the arguments of the extractions are meaningful concept mentions, we filter the candidate set with two simple constraints: First, an argument has to contain at least one noun token, and second, it cannot be longer than ten tokens. This removes overly long arguments that are clauses rather than suitable labels for concepts.

Post-Processing In addition, we apply three rule-based post-processing steps that refine the extractions in order to increase the recall of the candidate sets. First, using off-the-shelf coreference resolution, we try to resolve pronominal anaphora in arguments of the propositions.

Second, if an argument is a conjoining construction, as indicated by *conj*-edges in a dependency parse, we break it down into its conjuncts and introduce separate extractions for each of them:

(Caffeine ; works with ; young children and teens)

would be split into two extractions

(Caffeine ; works with ; young children)
(Caffeine ; works with ; teens)

And third, if the second argument starts with a verb, as in the following example,

(Herbal supplements ; have been used to ; treat the symptoms of ADHD)

we move that verb and subsequent prepositions to the predicate. In the example, the predicate is extended to *have been used to treat*, reducing the second argument to *the symptoms of ADHD*.

3.2 Concept Graph Construction

Given the concept mentions extracted in the previous step, several of these mentions may refer to the same concept. While this is obvious if the mentions are identical (e.g. *Caffeine* in the first two extractions of Figure 1), they could also differ slightly (e.g. *ADHD symptoms* and *the symptoms of ADHD*) or be synonyms or paraphrases without any lexical overlap. In this step, we connect all extracted propositions (m_1, r, m_2) to a concept graph by grouping coreferent mentions to a set of unique, non-redundant concepts (see Figure 1). As this special form of concept-specific and cross-document coreference goes beyond the capabilities of off-the-shelf coreference resolution systems, we propose a solution based on pairwise classification and set partitioning.

3.2.1 Pairwise Mention Classification

Given the set M of concept mentions, we want to determine whether a pair $(m_1, m_2) \in M^2$ refers to the same concept or not. We model this as a binary classification problem using a log-linear model

$$P(y = 1|m_1, m_2, \theta) = \sigma(\theta^T \phi(m_1, m_2))$$

where a positive classification, $y = 1$, means that the mentions are coreferent, $\phi(m_1, m_2)$ are features for a pair of mentions, σ is the sigmoid function and θ are the learned parameters.

As features we use different similarity measures that indicate if both terms have the same meaning. Lexical features are normalized Levenshtein distance and the overlap (Jaccard coefficient) between stemmed content words. To capture similarity on a semantic level, we use cosine similarity between concept embeddings[4] and two measures using word-level similarity based on Latent Semantic Analysis (Deerwester et al., 1990) and WordNet (Resnik, 1995) together with a word alignment method, both implemented in Semilar (Rus et al., 2013). The selection of these features is driven by practical reasons: Since the number of pairs is in $\mathcal{O}(|M|^2)$, the feature set has to be small and restricted to fast-to-compute metrics to make the approach computationally feasible.

3.2.2 Mention Partitioning

The task of grouping mentions to concepts can be seen as finding a partition of M based on the pairwise classifications. However, this is non-trivial, as single predictions might conflict: Both (a, b) and (b, c) could be classified as coreferent, but not (a, c). Formally, the relation of all coreferent pairs $S \subseteq M^2$ has to be an equivalence relation, i.e. reflexive, symmetric and transitive, to represent a consistent partitioning.

For a similar problem, Barzilay and Lapata (2006) propose to use ILP to find a valid partitioning that maximally agrees with the pairwise classifications. Let $x_p \in \{0, 1\}$ indicate the coreference of mentions $p = (m_1, m_2)$ and be $c(p) = P(y = 1|m_1, m_2)$. Then they optimize the assignments x_p to maximize

$$\sum_{p \in M^2} c(p)\ x_p + (1 - c(p))\ (1 - x_p) \qquad (1)$$

[4]Using the sum of vectors for all tokens; 300-dimensional word2vec Google News embeddings (Mikolov et al., 2013).

Algorithm 1 Greedy Local Partitioning Search

Input: pairwise predictions $c(p)$ for $p \in M^2$
Output: coreferent pairs $S \subseteq M^2$

```
1:  function SEARCH(x, y)
2:      S ← { p | c(p) ≥ 0.5 }
3:      b ← SCORE(S)
4:      S_m ← SHUFFLE(TRANSREDUCTION(S))
5:      for p ∈ S_m do
6:          S' ← S \ {p}
7:          if b < SCORE(S') then
8:              b ← SCORE(S'), S ← S'
9:      return TRANSCLOSURE(S)
10: function SCORE(S)
11:     S+ ← TRANSCLOSURE(S)
12:     return Compute Equation 1 for S+
```

under the transitivity constraints

$$x_{p_i} \geq x_{p_j} + x_{p_k} - 1 \qquad (2)$$

for all $p_i, p_j, p_k \in M^2$ where $i \neq j \neq k$. Unfortunately, this ILP needs $\mathcal{O}(|M|^2)$ variables and $\mathcal{O}(|M|^3)$ constraints, which makes it difficult to solve for our problem (where $|M|$ is up to 20k and we thus have up to 400 million variables and 8 trillion constraints). As an alternative approach, we use an approximate optimization algorithm.

Algorithm 1 shows our greedy local search algorithm. It creates the transitive closure over all positive classifications as the initial solution and computes the objective function (lines 2-3). This solution is a very aggressive grouping that joins as many mentions as possible, ignoring all negative classifications. The algorithm then tries to iteratively improve this solution by removing one positive classification at a time (line 6) if that improves the objective (lines 7-8). Removals are only tested for pairs in the transitive reduction of the initial solution (lines 4-5), as removing others would not change the partitioning. This approach still runs for several hours on large problem instances due to the expensive calculation of SCORE (lines 11-12), making more complete local searches, using best-first or beam search, impractical.

As a result, we obtain a relation S that partitions M into a set of sets $C = \{C_1, \ldots C_n\}$ where each C_i is a set of mentions representing a concept.

3.2.3 Graph Construction

Using the partitioning, we can now connect the extracted propositions to a graph $G = (C, R)$ in

which the nodes are concepts C and an edge with label r exists for every proposition (m_1, r, m_2) between the nodes of the concepts of m_1 and m_2. For each concept C_i, we select one mention $m_l \in C_i$ as its label. We experimentally found that using the most frequent mention, breaking ties by choosing the shortest, is a good heuristic to choose the most generic and representative label.

3.3 Graph Summarization

With the concept graph $G = (C, R)$ built from the documents, we can cast the selection of a summary concept map as a subgraph selection problem:

Given G, find a subgraph $G' = (C', R')$ with $C' \subseteq C$ and $R' \subseteq R$ that maximizes

$$\sum_{C_i \in C'} s(C_i) \tag{3}$$

such that the subgraph is connected and satisfies the size constraint $|C'| \leq L$. With $s(C_i)$, we denote the importance of concept C_i.

3.3.1 Subgraph Selection

The selection of a subgraph that maximizes Equation 3 can be formulated as an ILP. Let x_i be a binary decision variable that represents whether concept C_i is part of the subgraph. Then, the objective can be written as

$$\max \sum_{i=1}^{|C|} x_i \, s(C_i) \tag{4}$$

subject to[5]

$$x_i \in \{0, 1\} \;\; \forall \, i \in C \tag{5}$$

$$\sum_{i=1}^{|C|} x_i \;\; \leq \;\; L. \tag{6}$$

To ensure that the selected subgraph is connected, we introduce flow variables following previous work (Li et al., 2016; Liu et al., 2015). Let f_{ij} be a non-negative integer variable capturing the flow from concept C_i to C_j. We only introduce flow variables for concept pairs that have a relation in R. The constraints

$$f_{ij} \;\; \leq \;\; x_i \cdot |C| \;\; \forall \, (i, j) \in R \tag{7}$$

$$f_{ij} \;\; \leq \;\; x_j \cdot |C| \;\; \forall \, (i, j) \in R \tag{8}$$

$$\sum_i f_{ij} - \sum_k f_{jk} - x_j \;\; = \;\; 0 \;\; \forall \, j \in C \tag{9}$$

$$f_{ij} \in \mathbb{N} \;\; \forall \, (i, j) \in R \tag{10}$$

enforce that flow can only move between concepts that are selected (7,8) and a selected concept consumes one unit of flow (9). Further, let $i = 0$ be a virtual root node and e_{0i} a virtual edge from the root to each concept. The additional constraints

$$|C| \cdot e_{0i} - f_{0i} \;\; \geq \;\; 0 \;\; \forall \, i \in C \tag{11}$$

$$\sum_{i=1}^{|C|} e_{0i} \;\; = \;\; 1 \tag{12}$$

$$\sum_{i=1}^{|C|} f_{0i} \;\; - \;\; \sum_{i=1}^{|C|} x_i \;\; = \;\; 0 \tag{13}$$

$$e_{0i} \in \{0, 1\} \;\; \forall \, i \in C \tag{14}$$

$$f_{0i} \in \mathbb{N}_0 \;\; \forall \, i \in C \tag{15}$$

ensure that only one virtual edge can be active (12), that the virtual node can only send flow over this active edge (11) and that the total amount of flow sent from the root cannot exceed the size of the selected subgraph (13). As a consequence, if n concepts are selected, n units of flow are sent from the root over the edges of the graph and each selected concept consumes one of them. This is only possible if the subgraph is connected.

The above ILP formulation has the advantage that it only requires $\mathcal{O}(|C| + |R|)$ variables and constraints as opposed to $\mathcal{O}(|C|^2)$ with the flow constraints used by Li et al. (2016). For sparse graphs, where $|R| \ll |C|^2$, this leads to much smaller ILPs. We further leverage the fact that G is typically disconnected and solve separate ILPs for each connected component. Only with these measures, the ILP approach can be solved for the real-world problem sizes in our evaluation dataset.

3.3.2 Score Prediction

The subgraph selection introduced above relies on estimates $s(C_i)$ of a concept's importance. These scores are estimated with a linear model

$$s(C_i) = \vartheta^T \psi(C_i, t)$$

where $\psi(C_i, t)$ are features for a concept C_i in a document cluster on topic t. Parameters ϑ are learned with *SVM*^{rank} (Joachims, 2002). We use a rich set of features that are commonly used in summarization and keyphrase extraction and briefly describe them in the following section:

Frequency Concept frequency and document frequency based on the partitioned mentions. In addition, frequencies re-weighted with background inverse document frequencies from Google N-Grams (Klein and Nelson, 2009).

[5] To simplify the notation, we write $i \in C$ instead of $i \in \{1, \ldots, |C|\}$ and correspondingly for R.

Position First, average and last position of a concept and the distance between first and last.

Topic Relatedness Relatedness of the concept to the topic, measured as the semantic similarity between the concept label and the document cluster's topic description t. As similarities, we use the measures introduced in Section 3.2.1.

Length Length of shortest, average and longest mention measured in tokens and in characters.

Label Several features describing the concept label, including the number of stopwords, capitalization, part-of-speech and named entities.

Word Categories As suggested in recent work by Yang et al. (2017), dictionary-based features that capture general properties of words such as concreteness, familiarity or imagery, using the MRC Psycholinguistic Database (Coltheart, 1981), the LIWC dictionary and an additional list of concreteness values (Brysbaert et al., 2014).

In addition, we derive several features from the concept's position in the concept graph G:

Centrality Measures Measures such as degree, closeness and betweenness centrality as well as PageRank scores that indicate the centrality of the node C_i in the graph G.

Concept Map HARD and CRD scores suggested by Reichherzer and Leake (2006) and their underlying metrics. They are slight variations or extensions of common graph metrics that were specifically developed to describe concept maps.

Graph Degeneracy Following Tixier et al. (2016) who show that graph degeneracy is helpful to identify keyphrases, we use the graph core number and core rank suggested by them.

All numeric features are discretized into bins, such that the final feature set has only binary features.

3.3.3 Finalization

After predicting scores for every concept and selecting the highest scoring subgraph with the ILP, we use this subgraph as the summary concept map. However, this graph might contain multiple edges between certain concepts. Because this is rare and the number of available relations is low, we use a simple heuristic and select the relation that was extracted with the highest confidence in the first step. The resulting graph is the final summary.

	EDUC	WIKI
Topics	30	38
Documents	40.5	14.6
Tokens	97880	27066
Concepts	25.0	11.3
Relations	25.2	13.8
Compression	0.16%	0.33%

Table 1: Benchmark datasets used in experiments. Values are averages per topic. Compression = tokens in concept map / tokens in documents.

4 Experimental Setup

4.1 Data

We evaluate our approach using two benchmark datasets and compare the generated concept maps against reference maps. As the first dataset, we use a recently published corpus by Falke and Gurevych (2017a) that provides summary concept maps for document clusters on educational topics. They were manually created using crowdsourcing and expert annotators. As the second dataset, we use a corpus in which the introductions of featured Wikipedia articles are used as summaries for web documents (Zopf et al., 2016). This property allows us to make use of the links to other Wikipedia pages in the summaries as annotations of concepts. In combination with Open Information Extraction, we can therefore automatically derive concepts and relations from the Wikipedia summaries to obtain a second corpus of summary concept maps.

We refer to these datasets as **EDUC** and **WIKI**. Table 1 shows their characteristics. Note that in both datasets the summaries are much smaller than the document sets, posing a challenging summarization task. In addition, the document clusters of EDUC are very large, constituting a challenging but real-world evaluation setting regarding computational efficiency. We randomly split both datasets into equally sized training and test sets.

4.2 Evaluation Metrics

As input, our model receives the documents to summarize, the corresponding topic description and the number of concepts in the reference concept map as the size limit. To compare a system-generated concept map with a reference concept

map we represent both as sets of propositions P, i.e. a set in which each element is the concatenation of a relation label with its two concept labels. We then calculate the overlap between the set P_S for the system map and the set P_R for the reference map. As the number of relations and thus propositions of the generated map can differ, we report precision, recall and F1-scores.

Our first metric based on **METEOR** (Denkowski and Lavie, 2014) has the advantage that it takes synonyms and paraphrases into account and does not solely rely on lexical matches. For each pair of propositions $p_s \in P_S$ and $p_r \in P_R$ we calculate the match score $meteor(p_s, p_r) \in [0, 1]$. Then, precision and recall per map are computed as:

$$Pr = \frac{1}{|P_S|} \sum_{p \in P_S} max\{meteor(p, p_r) \,|\, p_r \in P_R\}$$

$$Re = \frac{1}{|P_R|} \sum_{p \in P_R} max\{meteor(p, p_s) \,|\, p_s \in P_S\}$$

The F1-score is the equally weighted harmonic mean of precision and recall. Scores per map are macro-averaged over all topics.

As a second metric, we compute **ROUGE** (Lin, 2004), the standard metric for textual sumarization. We concatenate all propositions of a map into a single string, s_S and s_R, and separate propositions with a dot to ensure that no bigrams span across propositions and the metric is therefore order-independent. We run ROUGE 1.5.5[6] with s_S as the peer summary and s_R as a single model summary to compute ROUGE-2.

4.3 Implementation and Training

All source documents are preprocessed with Stanford CoreNLP 3.7.0 (Manning et al., 2014) to obtain tokenization, sentence splitting, part-of-speech tags, named entities, dependency parses and coreference chains. For Open Information Extraction, we use OpenIE-4[7], a system developed at the University of Washington that is currently state-of-the-art according to a recent comparison (Stanovsky and Dagan, 2016). ILPs are solved with the IBM CPLEX optimizer.[8]

The concept coreference model is implemented using the logistic regression model of Weka (Hall et al., 2009). For EDUC, we trained it on 17,500 pairs of mentions, and for WIKI, on 4,500 pairs of mentions, which were in both cases derived from the reference concept maps of the training part of the respective dataset.

The SVM^{rank} model for importance scoring is trained with Dlib[9]. We use the set of all extracted concepts from all topics in the training set and assign binary labels if these concepts also occur in the reference concept maps. The SVM then learns weights for all features such that the positive instances per topic are ranked higher than the negative instances. We tuned the regularization parameter C of the SVM by testing values from 0.1 to 100 with leave-one-out cross-validation on the training topics. The final models are trained on the full training set with the best parameter. We did this separately for all ablations of our model that produce different training data, obtaining best parameters of $C = 10$ for *coref=lem* on EDUC and all models on WIKI as well as $C = 30$ for *coref=doc* and *our model* on EDUC (see Table 2).

5 Results and Analysis

5.1 Evaluation Results

We compare our model against several previously suggested methods. As unsupervised methods, we include concept selection based on frequency (Valerio and Leake, 2006), denoted as *Valerio 06*, selection with idf-corrected frequencies (Zubrinic et al., 2015), *Zubrinic 15*, and using the popular *PageRank* algorithm (Page et al., 1999). For a fair comparison, we run all methods on the same extracted concepts and relations and with our ILP-based subgraph selection. In addition, we include the baseline method *Falke 17* published along with the EDUC corpus (Falke and Gurevych, 2017a), which includes a supervised importance scoring model based on a binary classifier. To the best of our knowledge, this is all existing work for this task to which we can compare the proposed model.

Table 2 shows METEOR and ROUGE-2 scores for all methods on both datasets. Our model outperforms all three unsupervised approaches significantly on both datasets, demonstrating the superiority of the supervised scoring model. With regard to *Falke 17*, which is supervised to a similar extent, the results are twofold: While our model improves in ROUGE-2, it has a lower METEOR score. We looked into these results in de-

[6] Parameter: -n 2 -x -m -c 95 -r 1000 -f A -p 0.5 -t 0 -d -a

[7] https://github.com/knowitall/openie

[8] https://ibm.com/software/commerce/optimization/cplex-optimizer/

[9] http://dlib.net/ml.html

	EDUC						WIKI					
	METEOR			ROUGE-2			METEOR			ROUGE-2		
Approach	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
PageRank	11.78	16.21	†13.61	7.14	11.66	†8.66	13.27	14.13	†13.62	8.35	6.17	‡7.01
Valerio 06	11.89	16.12	†13.65	7.33	12.09	‡8.97	13.44	13.79	‡13.55	8.57	7.16	‡7.61
Zubrinic 15	12.48	16.44	†14.15	7.68	12.08	‡9.25	14.63	14.92	‡14.72	10.50	7.91	‡8.87
Falke 17	15.12	19.49	**17.00**	6.03	17.98	8.91	14.30	23.11	17.46	6.77	23.18	10.20
Our model	15.14	17.34	16.12	9.37	11.93	**10.38**	19.57	18.98	**19.18**	17.00	10.69	**12.91**
- coref=lem	13.93	15.42	†14.57	8.21	8.59	†8.25	18.32	17.24	17.59	13.99	9.53	11.07
- coref=doc	14.14	15.21	†14.54	7.99	6.78	†7.26	16.81	16.63	16.59	13.09	9.16	10.29
- w/o ILP	15.29	17.46	16.26	9.38	11.88	**10.38**	18.22	17.80	17.94	14.73	9.74	11.51
s, ILP*	23.32	27.52	25.16	26.09	23.93	24.74	29.04	26.76	27.73	29.08	18.79	22.54
s, w/o ILP*	18.28	25.15	†21.13	17.52	21.97	†19.34	24.45	24.46	†24.83	24.06	17.39	†19.57

Table 2: Results on test sections of both datasets for our model and previous work. (Improvements of our model are significant compared to approaches marked (for F1) with † ($p \leq 0.01$) or ‡ ($p \leq 0.05$)).[12]

tail and found that the high scores of *Falke 17* are due to heavy overgeneration during relation extraction, introducing many rather meaningless relations into the concept map.[10] Hence, the method only obtains higher scores by sacrificing the quality of the extracted propositions.

To verify this observation, we carried out an additional human evaluation between the two systems, capturing aspects beyond the content-oriented automatic metrics. For each topic, the concept maps generated by both approaches were shown to five crowdworkers on Mechanical Turk and they were asked for their preference with regard to different quality dimensions.[11] Table 3 shows that our concept maps tend to have more meaningful and topic-focused propositions and are especially more grammatical and less redundant.

Dimension	*Falke 2017*	*Our*
Meaning	44%	56%
Grammaticality	31%	69%
Focus	44%	56%
Non-Redundancy	21%	79%

Table 3: Human preference judgments between concept maps generated on EDUC ($n = 75$).

5.2 Analysis

Concept Coreference To analyze the contribution of our concept coreference detection and partitioning (§3.2.1,§3.2.2), we replaced it with two simpler baselines: merging concepts based on string matches after lemmatization (*coref=lem*), as done in previous work, and using per-document coreference chains detected by CoreNLP and merging them across documents by lemmatized string matching (*coref=doc*). Both alternatives cause a drop in both metrics on EDUC and WIKI, showing that our approach is important for the model's performance. The baselines merge much less mentions than necessary but also tend to lump too many too different mentions together. In contrast, our model can make many more merges based on semantic similarity and at the same time manages to avoid lumping effects by relying on the global partitioning approach.

Subgraph Selection To analyze the effectiveness of the subgraph selection (§3.3.1), we replaced the ILP approach with a greedy heuristic similar to Zubrinic et al. (2015): Given the graph of scored concepts, start with the most important one and select the best neighbor (by score, breaking ties by node degree) until the size limit is reached. While the ILP will always find the optimal solution and hence the best subgraph, this heuristic approach does not have such a guarantee. In fact, it found the optimal subgraph for only 35% of the topics, selecting a subgraph with an on average 0.63% (EDUC) and 1.30% (WIKI) lower objective function score in the other cases.

Row *w/o ILP* in Table 2 shows the effect on the summary concept map. While it is rather small for EDUC, the differences on WIKI are bigger – in line with the observation that the selected subgraphs are less optimal. A problem for this analysis are errors in the preceding scoring step: The optimal

[10] Note that METEOR scores can be improved by incorrect relations if they are between a correct pair of concepts, leading to a partial match of the proposition.

[11] To control for the influence of graph layouting quality, we showed the concept maps as simple lists of propositions.

[12] Approximate randomization test with $N = 10000$.

Method	Var.	Const.	Time (s)
(Li et al., 2016)	37M	75M	2670.61
by component	26M	52M	999.25
Our ILP	22k	31k	7.31
by component	18k	26k	5.61

Table 4: Comparison of average ILP size and runtime per topic for subgraph selection on EDUC.

	EDUC		WIKI	
Step	Count	Recall	Count	Recall
Mentions	8630	73.87	2549	88.93
Concepts	4022	60.27	1315	82.38
Subgraph	25	16.53	11	30.71

Table 5: Average number of concepts and recall per topic at different steps in our model.

subgraph according to the estimated scores might not be the best with regard to the gold standard, explaining the slightly higher METEOR scores on EDUC without the ILP. To control for this effect, we also tested the selection using gold scores $s^*(C_i)$ for all concepts C_i, demonstrating that the optimal subgraphs selected by the ILP are clearly superior (last two rows in Table 2).

Score Prediction The contribution of our supervised scoring model based on ranking SVMs (§3.3.2) can be seen in Table 2 when comparing it to the unsupervised approaches *PageRank*, *Valerio 06* and *Zubrinic 15*. Note that all models use the same concepts and relations as input and the same ILP-based subgraph selection. Our model clearly outperforms all of them. Looking into the learned weights for our set of features, we observed that the most helpful features are frequencies, in particular document frequency and idf-weighted concept frequency, and topic relatedness as well as page rank. To identify unimportant concepts (i.e. assigning low scores), the model makes use of concreteness values and the label's length.

Runtime As mentioned earlier, the size of the document sets in EDUC resembles an interesting real-world setting that required us to pay special attention to complexity. Table 4 compares our subgraph selection ILP with the ILP formulation by Li et al. (2016). For the extracted graphs, with on average 4022 nodes and 5613 edges between them, our formulation leads to ILPs that are orders of magnitude smaller and can be solved in a fraction of the time.[13] For both formulations, solving separate ILPs for each connected component in the graph further improves the runtime.

Error Analysis Table 5 shows the number of concepts and their recall at different steps in our model, which is a good indicator of bottlenecks. The recall of mentions shows that performance is already lost during extraction, suggesting that better approaches would be beneficial. We observed that the problem is mainly the identification of correct spans rather than missing some concepts completely. A custom extraction model instead of relying on Open IE could resolve this.

With regard to concept coreference, we found that even more coreferent mentions could be grouped together. However, while the current model only accidentally merged mentions of different gold concepts in two cases across all topics, a stronger grouping could introduce more of these errors. Please also note that the drop in recall in Table 5 is due to exact string matching of the recall metric used here, missing concepts for which the selected cluster label is not exactly the gold concept. As the METEOR and ROUGE evaluations show, this is not a problem for the final result.

Finally, Table 5 reveals that one of the main bottlenecks is to determine the important concepts. On both datasets, but especially on the bigger document sets of EDUC, a substantial amount of recall is lost during this challenging step.

6 Conclusion

We proposed a new model for concept-map-based MDS and showed that it outperforms several methods from previous work. All of our contributions, including concept coreference resolution, the supervised scoring model and the global optimization approach contribute to its efficacy. In addition, it is able to scale to large document sets, which makes it much faster than previous methods in realistic scenarios with such documents sets.

Acknowledgments

This work has been supported by the German Research Foundation as part of the Research Training Group "Adaptive Preparation of Information from Heterogeneous Sources" (AIPHES) under grant No. GRK 1994/1.

[13] Times for running CPLEX multi-threaded on 24 cores. Direct comparison with the same data on the same machine.

References

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open Information Extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, pages 2670–2676, Hyderabad, India.

Regina Barzilay and Mirella Lapata. 2006. Aggregation via set partitioning for natural language generation. In *Proceedings of the Human Language Technology Conference of the NAACL*, pages 359–366, New York, NY, USA.

Geoffrey Briggs, David A. Shamma, Alberto J. Cañas, Roger Carff, Jeffrey Scargle, and Joseph D. Novak. 2004. Concept Maps Applied to Mars Exploration Public Outreach. In *Concept Maps: Theory, Methodology, Technology. Proceedings of the First International Conference on Concept Mapping*, pages 109–116, Pamplona, Spain.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known English word lemmas. *Behavior Research Methods*, 46(3):904–911.

Ziqiang Cao, Furu Wei, Li Dong, Sujian Li, and Ming Zhou. 2016. Ranking with Recursive Neural Networks and Its Application to Multi-document Summarization. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 2153–2159, Phoenix, AZ, USA.

Marco Carvalho, Rattikorn Hewett, and Alberto J. Cañas. 2001. Enhancing Web Searches from Concept Map-based Knowledge Models. In *Proceedings of the 5th World Multi-Conference on Systemics, Cybernetics and Informatics*, pages 69–73, Orlando, FL, USA.

Max Coltheart. 1981. The MRC psycholinguistic database. *The Quarterly Journal of Experimental Psychology Section A*, 33(4):497–505.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Michael Denkowski and Alon Lavie. 2014. Meteor Universal: Language Specific Translation Evaluation for Any Target Language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, Baltimore, MD, USA.

Tobias Falke and Iryna Gurevych. 2017a. Bringing Structure into Summaries: Crowdsourcing a Benchmark Corpus of Concept Maps. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2969–2979, Copenhagen, Denmark.

Tobias Falke and Iryna Gurevych. 2017b. GraphDocExplore: A Framework for the Experimental Comparison of Graph-based Document Exploration Techniques. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 19–24, Copenhagen, Denmark.

Tobias Falke and Iryna Gurevych. 2017c. Utilizing Automatic Predicate-Argument Analysis for Concept Map Mining. In *Proceedings of the 12th International Conference on Computational Semantics (IWCS)*, Montpellier, France.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1):10–18.

Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 133–142, Edmonton, Canada.

Martin Klein and Michael L. Nelson. 2009. Correlation of Term Count and Document Frequency for Google N-Grams. In *Advances in Information Retrieval*, volume 5478 of *Lecture Notes in Computer Science*, pages 620–627. Springer, Berlin, Heidelberg.

Juliana H. Kowata, Davidson Cury, and Maria Claudia Silva Boeres. 2010. Concept Maps Core Elements Candidates Recognition from Text. In *Concept Maps: Making Learning Meaningful. Proceedings of the 4th International Conference on Concept Mapping*, pages 120–127, Vina del Mar, Chile.

Wei Li, Lei He, and Hai Zhuge. 2016. Abstractive News Summarization based on Event Semantic Link Network. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 236–246, Osaka, Japan.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain.

Fei Liu, Jeffrey Flanigan, Sam Thomson, Norman Sadeh, and Noah A. Smith. 2015. Toward Abstractive Summarization Using Semantic Representations. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1077–1086, Denver, Colorado.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 55–60, Baltimore, MD, USA.

Ryan McDonald. 2007. A Study of Global Inference Algorithms in Multi-Document Summarization. In *Proceedings of the 29th European conference on IR research*, pages 557–564, Rome, Italy.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119, Lake Tahoe, NV, USA.

Joseph D. Novak and D. Bob Gowin. 1984. *Learning How to Learn*. Cambridge University Press, Cambridge.

Andrew Olney, Whitney Cade, and Claire Williams. 2011. Generating Concept Map Exercises from Textbooks. In *Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 111–119, Portland, OR, USA.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The PageRank Citation Ranking: Bringing Order to the Web: Technical Report.

Iqbal Qasim, Jin-Woo Jeong, Jee-Uk Heu, and Dong-Ho Lee. 2013. Concept map construction from text documents using affinity propagation. *Journal of Information Science*, 39(6):719–736.

Kanagasabai Rajaraman and Ah-Hwee Tan. 2002. Knowledge discovery from texts: A Concept Frame Graph Approach. In *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, pages 669–671, McLean, VA, USA.

Thomas Reichherzer and David Leake. 2006. Understanding the Role of Structure in Concept Maps. In *Proceedings of the Twenty-Eighth Annual Conference of the Cognitive Science Society*, pages 2004–2009, Vancouver, Canada.

Philip Resnik. 1995. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 448–453, Montreal, Canada.

Ryan Richardson and Edward A. Fox. 2005. Using concept maps as a cross-language resource discovery tool for large documents in digital libraries. In *Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, page 415, Denver, CO, USA.

Vasile Rus, Mihai Lintean, Rajendra Banjade, Nobal Niraula, and Dan Stefanescu. 2013. SEMILAR: The Semantic Similarity Toolkit. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 163–168, Sofia, Bulgaria.

Gabriel Stanovsky and Ido Dagan. 2016. Creating a Large Benchmark for Open Information Extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2300–2305, Austin, TX, USA.

Antoine Tixier, Fragkiskos Malliaros, and Michaelis Vazirgiannis. 2016. A Graph Degeneracy-based Approach to Keyword Extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1860–1870, Austin, TX, USA.

Alejandro Valerio and David B. Leake. 2006. Jump-Starting Concept Map Construction with Knowledge Extracted from Documents. In *Proceedings of the 2nd International Conference on Concept Mapping*, pages 296–303, San José, Costa Rica.

Alejandro Valerio, David B. Leake, and Alberto J. Cañas. 2012. Using Automatically Generated Concept Maps for Document Understanding: A Human Subjects Experiment. In *Proceedings of the 5th International Conference on Concept Mapping*, pages 438–445, Valetta, Malta.

Jorge J. Villalon. 2012. *Automated Generation of Concept Maps to Support Writing*. PhD Thesis, University of Sydney, Australia.

Yinfei Yang, Forrest Bao, and Ani Nenkova. 2017. Detecting (Un)Important Content for Single-Document News Summarization. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 707–712, Valencia, Spain.

Markus Zopf, Maxime Peyrard, and Judith Eckle-Kohler. 2016. The Next Step for Multi-Document Summarization: A Heterogeneous Multi-Genre Corpus Built with a Novel Construction Approach. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 1535–1545, Osaka, Japan.

Amal Zouaq and Roger Nkambou. 2009. Evaluating the Generation of Domain Ontologies in the Knowledge Puzzle Project. *IEEE Transactions on Knowledge and Data Engineering*, 21(11):1559–1572.

Krunoslav Zubrinic, Ines Obradovic, and Tomo Sjekavica. 2015. Implementation of method for generating concept map from unstructured text in the Croatian language. In *23rd International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pages 220–223, Split, Croatia.