Evaluation of New Features for Extractive Summarization of Meeting Transcripts

Improvement of meeting summarization based on functional segmentation, introducing topic model, named entities and domain specific frequency measure

EMILIO MARINONE
Evaluation of New Features for Extractive Summarization of Meeting Transcripts

Improvement of meeting summarization based on functional segmentation, introducing topic model, named entities and domain specific frequency measure

EMILIO MARINONE

Master in Systems, Control and Robotics
Date: February 7, 2019
Email: marinone@kth.se
Supervisor: Johan Boye
Examiner: John Folkesson
Principal: Seavus Stockholm AB
Principal's Supervisor: Reijo Silander
Swedish title: Utvärdering av nya funktioner för extraktiv sammanfattning av mötesutskrifter
School of Electrical Engineering and Computer Science
Abstract

Automatic summarization of meeting transcripts has been widely studied in last two decades, achieving continuous improvements in terms of the standard summarization metric (ROUGE). A user study has shown that people noticeably prefer abstractive summarization rather than the extractive approach. However, a fluent and informative abstract depends heavily on the performance of the Information Extraction method(s) applied.

In this work, basic concepts useful for understanding meeting summarization methods like Parts-of-Speech (POS), Named Entity Recognition (NER), frequency and similarity measure and topic models are introduced together with a broad literature analysis. The proposed method takes inspiration from the current unsupervised extractive state of the art and introduces new features that improve the baseline. It is based on functional segmentation, meaning that it first aims to divide the preprocessed source transcript into monologues and dialogues. Then, two different approaches are used to extract the most important sentences from each segment, whose concatenation together with redundancy reduction creates the final summary.

Results show that a topic model trained on an extended corpus, some variations in the proposed parameters and the consideration of word tags improve the performance in terms of ROUGE Precision, Recall and F-measure. It outperforms the currently best performing unsupervised extractive summarization method in terms of ROUGE-1 Precision and F-measure.

A subjective evaluation of the generated summaries demonstrates that the current unsupervised framework is not yet accurate enough for commercial use, but the new introduced features can help supervised methods to achieve acceptable performance. A much larger, non-artificially constructed meeting dataset with reference summaries is also needed for training supervised methods as well as a more accurate algorithm evaluation.

The source code is available on GitHub: https://github.com/marinone94/ThesisMeetingSummarization
Sammanfattning

Automatgenererade textsammanfattningar av mötestranskript har varit ett allmänt studerat område de senaste två decennierna där resultatet varit ständiga förbättringar mätt mot standardsammanfattningsvärdet (ROUGE). En studie visar att människor märkbart föredrar abstraherade sammanfattningar gentemot omfattande sammanfattningar. En informativ och flytande textsammanfattning förlitar sig däremot mycket på informationsextraheringsmetoden som används.


Resultaten visar att en ämnesmodell, tränad på ett omfattande korpus med viss variation i de föreslagna parametrarna och med ordmärkning i åtanke, förbättrar prestandan mot ROUGE, precision, Recall och F-mätning. Den överträffar den nuvarande bästa Rouge-1 precision och F-mätning.

En subjektiv utvärdering av de genererade textsammanfattningarna visar att det nuvarande, övervakade ramverket inte är exakt nog för kommersiellt bruk än men att de nyintroducerade egenskaperna kan hjälpa övervakade metoder uppnå acceptabla resultat. En mycket större, icke artificiellt skapad, datamängd bestående utav textsammanfattningar av möten krävs för att träna de övervakade metoderna så väl som en mer noggrann utvärdering av de utvalda algoritmerna. Nya och existerande sammanfattningsmetoder kan appliceras på meningar extraherade ur den föreslagna metoden.
## Contents

1 Introduction ............................................. 1  
   1.1 Issues ............................................. 2  
   1.2 Extractive vs abstractive: two approaches to text summarization ............................... 3  
   1.3 A user study in the meeting domain ............................................. 4  
   1.4 Contribution ............................................. 4  
   1.5 Report structure ............................................. 5  

2 Background ............................................. 6  
   2.1 Basic concepts ............................................. 6  
      2.1.1 Similarity measure ............................................. 6  
      2.1.2 Frequency measure ............................................. 6  
      2.1.3 Divergence measure ............................................. 7  
      2.1.4 Tokenization, POS tag and NER ............................................. 8  
      2.1.5 Text preprocessing ............................................. 8  
      2.1.6 Topic modelling ............................................. 9  
      2.1.7 NLP open source libraries ............................................. 9  
      2.1.8 Precision, Recall and F-measure ............................................. 10  
   2.2 Extractive methods ............................................. 11  
      2.2.1 Binary classification ............................................. 11  
      2.2.2 Graph-based methods ............................................. 12  
      2.2.3 Cluster-based methods ............................................. 13  
      2.2.4 Ranking-based methods ............................................. 13  
      2.2.5 Unsupervised extractive state of the art ............................................. 14  
   2.3 Abstractive methods ............................................. 17  
      2.3.1 Graph-based methods ............................................. 17  
      2.3.2 Focused summarization ............................................. 17  
      2.3.3 Template-based methods ............................................. 18  
      2.3.4 Abstractive state of the art ............................................. 19
3 Proposed Method

3.1 Preprocessing ............................................. 23
3.2 Functional segmentation .............................. 24
  3.2.1 Candidate boundaries ................................. 25
  3.2.2 Optimal segmentation ................................. 27
  3.2.3 Segments labelling .................................... 28
3.3 Monologues summarization ............................. 31
  3.3.1 Keyword extraction .................................... 31
  3.3.2 ILP formulation ......................................... 32
3.4 Dialogues summarization ............................... 34
  3.4.1 Topical similarity ...................................... 34
  3.4.2 Graph construction ..................................... 35
  3.4.3 Utterance importances initialization ............... 36
  3.4.4 Utterance importances propagation ................. 37
  3.4.5 Redundancy reduction ................................. 37

4 Experimental Results

4.1 Project datasets and metrics .......................... 39
  4.1.1 Corpora .................................................. 40
  4.1.2 Metrics .................................................. 41
4.2 Experiments ............................................... 41
  4.2.1 Baseline ............................................... 42
  4.2.2 Test case 1 .............................................. 42
  4.2.3 Test case 2 .............................................. 43
  4.2.4 Test case 3 .............................................. 43
  4.2.5 Test case 4 .............................................. 43
  4.2.6 Test case 5 .............................................. 43
  4.2.7 Test case 6 .............................................. 44
  4.2.8 Test case 7 .............................................. 44
  4.2.9 Test case 8 .............................................. 44
  4.2.10 Test case 9 ............................................. 45
  4.2.11 Test case 10 .......................................... 45
  4.2.12 Test case 11 .......................................... 45
  4.2.13 Test case 12 .......................................... 45
  4.2.14 Test case 13 .......................................... 46
  4.2.15 Test case 14 .......................................... 46
  4.2.16 Test case 15 .......................................... 46
4.3 Analysis of the results .................................. 48
5 Conclusions ........................................ 51
  5.1 Basic concepts and literature .................. 51
  5.2 Proposed method and results .................. 51
  5.3 Contribution and future works ................. 52
  5.4 Applicability to commercial use ............... 52
  5.5 Social and ethical impact ...................... 52

6 Acknowledgments .................................. 54

Bibliography ........................................ 55

A SpaCy NER Types and Descriptions ............... 62

B Human Gold-Standard Functional Segmentation ..... 63

C Test Cases ........................................ 66
  C.1 Baseline ...................................... 66
  C.2 Test case 1 - Topical similarity .............. 67
  C.3 Test case 2 - Topical and lexical similarity 68
  C.4 Test case 3 - More importance to the frequency measure for keyword extraction 68
  C.5 Test case 4 - More importance to the entropy measure in the keyword extraction 69
  C.6 Test case 5 - Introduce the BBC news corpus for training the topic model ............. 70
  C.7 Test case 6 - Introduce also the basic corpus for training the topic model ............. 70
  C.8 Test case 7 - suidf for lexical Similarity .... 71
  C.9 Test case 8 - Merge topical and lexical similarity before the random walk ............. 72
  C.10 Test case 9 - Use topic keywords for keyword extraction ............................. 73
  C.11 Test case 10 - Use topic keywords for keyword extraction and initial utterance importances 73
  C.12 Test case 11 - Accumulate topic keyword weights in each sentence ..................... 74
  C.13 Test case 12 - Accumulate topic keyword weights with extended model corpus .......... 75
  C.14 Test case 13 - Use named entities for keyword extraction ............................. 76
  C.15 Test case 14 - Use named entities for keyword extraction and initial utterance importances 76
C.16 Test case 15 - 30% compression rate . . . . . . . . . . . . . . . . . . 77

D  Summarization Example 79
Chapter 1

Introduction

A broad analysis based on research and practice has shown that meetings dominate workers' and managers' time, being often costly, unproductive and dissatisfying [50]. The authors also state that the amount and time spent in meetings have constantly increased in the last decades due to several reasons. This time is not always spent well, since unproductive meeting time corresponded to a $37 billion annual waste worldwide in 1995 [52]. And it is quite likely even worse nowadays. In most meetings, lots of extra time is also spent due to someone having to summarize the most important aspects discussed. For this reason, in the last two decades, many researches have been focused on developing methods that could efficiently produce an informative automatic summary from a meeting transcript.

Most readers have a clear idea of what a summary is, but let us recall its definition. It is "a text that is produced from one or more texts, that conveys important information in the original text(s), and that is no longer than half of the original text(s) and usually significantly less than that" [47]. Hence, a summarizer aims to preserve all the important information while noticeably reducing the length of the original source. Ideally, the output should also be fluent as well as grammatically and semantically correct.

A good performance in automatic summarization of meeting transcripts would allow companies and organizations to save time and money, since employees avoid repetitive activities such as producing meeting protocols or collecting information about specific aspects like
problems and decisions. Those summaries are extremely useful for the employees that could not attend that meeting, but can also be used for obtaining a clear and structured overview of the project as well as an analysis of the internal knowledge, skills and shortcomings.

1.1 Issues

The Automatic Speech Recognition (ASR) methods proposed have continuously decreased the transcription error rate. The massive introduction of deep network architectures has created an impressive breakthrough [57] and tests on different tools have shown the speaker is almost always correctly identified even in agitated dialogues. Hence, the automatic speech-to-text transformation, where each sentence is associated to a speaker id, provides a source transcript in real-time, which can then be summarized. However, this problem is much more complex than summarizing any other type of text, due to it being a multi-speaker conversational speech. Utterances are truncated, sentences are often ungrammatical and some of them may refer to topics discussed much earlier. These issues are illustrated in figure 1.1.

![Dialogue utterances in red and truncated sentences in light blue and yellow, from a meeting transcript.](image)

Several methods are achieving great results when summarizing structured texts like news, documents and books, but the same algorithms’ performance degrades drastically when applied to the meeting domain. Hence, they need at least to be adapted and extended to be suitable for meeting summarization. Furthermore, the two annotated datasets publicly available (AMI Meeting Corpus [11] and ICSI Meeting Corpus [31]) overall include less than 200 hours of recorded meetings. They have been artificially set up for research purpose, meaning
that the attendants discuss only about a specific problem, resulting in poor generalization. Finally, an ideal metric has not been found yet, mainly because a perfect human gold standard summary basically does not exist. Various research projects have been conducted in the past years, resulting in algorithms and methods that continuously improved the quality of the automatic summaries, considering both informativeness and readability.

1.2 Extractive vs abstractive: two approaches to text summarization

In general, text summarization can be divided into two main branches: extractive and abstractive. The former can be described as highlighting and extracting the most important sentences in a book. Basically, it selects the most important utterances and concatenates them to generate a summary. It is only composed by units available in the source. The latter instead generates a novel text including and adapting the information extracted. It requires the capability of introducing utterances not embodied in the original text. In the next chapter, different approaches of both categories are presented.

Figure 1.2: Section of a meeting transcript and related summaries.
Left: meeting transcript. In green, extracted utterances are highlighted.
Right: abstractive summary. In green, the abstract generated from the selected sentences.
1.3 A user study in the meeting domain

The extractive approach has dominated the field of meeting summarization for many years, since it does not require text generation. However, extractive algorithms present several critical issues. In [38], the first abstractor specifically developed for meeting transcripts is presented. A comparison with gold-standard human abstracts and extracts has shown that automatic abstracts significantly outperform the gold-standard extractive summary but their fluency and informativeness is still far from a gold-standard abstract. This finding is really impressive and shows a clear direction in which researches have to point to. Users were required to evaluate understanding of the meeting, required effort, coherence, relevance, usefulness and missing info. They preferred the automatic abstracts to the human extracts in all these six dimensions. Nevertheless, good extractive algorithms are required to generate useful abstracts, which would otherwise miss important sections or include unnecessary parts.

1.4 Contribution

This project aims to create an effective fully unsupervised\(^1\) framework to generate an informative extractive summary, while being domain independent and without any prior data about the topics discussed. Here, different summarization techniques are presented, and the proposed method extends the extractive summarizer based on functional segmentation. Briefly, it divides the meeting transcript into *monologues* and *dialogues*, then it summarizes independently each section with two different methods to exploit the difference between soliloquies and discussions [7]. This work introduces and evaluates the impact of new features in different subsections. The source code is available on GitHub\(^2\).

---
\(^1\)Understanding the differences between the two main types of machine learning methods - Supervised and Unsupervised learning: https://towardsdatascience.com/supervised-vs-unsupervised-learning-1f68e32ea8d/
\(^2\)https://github.com/marinone94/ThesisMeetingSummarization
1.5 Report structure

In the next chapter, the topic background is presented. After a brief introduction to the basic concepts used in several natural language processing tasks, the most effective extractive and abstractive approaches for meeting summarization are presented. In chapter 3, the proposed architecture is described in detail, as well as all the algorithms it is built on. In chapter 4, the evaluation metric and corpus used in this project are described before presenting the results. The final chapter summarizes the conclusions drawn and presents some ideas for future work. After the references and the acknowledgments, the reader will find a full meeting transcript, its automatically generated summary and the human gold-standard reference, together with other appendices.
Chapter 2

Background

Before describing the related works in this specific sub-field, a brief summary of basic concepts often used for this task is presented, aiming to produce a report useful and understandable also for readers without a specific background in natural language processing and machine learning.

2.1 Basic concepts

2.1.1 Similarity measure

Many techniques applied to text summarization compute utterance similarity, and the most common similitude measure in NLP is the cosine similarity, which is a measure of orientation between two non-zero vectors\(^1\). Hence, it does not consider their magnitudes. The cosine similarity between \(x\) and \(y\) is derived from the Euclidean dot product as:

\[
\text{CosSim}(x, y) = \frac{x \cdot y}{||x|| \cdot ||y||}, \quad e^{-1, 1} \tag{2.1}
\]

The cosine distance is just:

\[
\text{CosDist}(x, y) = 1 - \text{CosSim}(x, y) \tag{2.2}
\]

2.1.2 Frequency measure

Another key concept is the frequency measure. Given a document \(d\) from a set of documents \(D\), the most common weighting factor of a

\(^{1}\) Representing many different features
given word \( w \) in information retrieval is the term frequency - inverse
document frequency (tfidf):

\[
tfidf(w, d) = tf(w, d) \cdot idf(w) = \frac{N(w, d)}{N(d)} \cdot \left(1 + \ln \left(\frac{|D|}{|D^*|}\right)\right) \quad (2.3)
\]

where:

- \( N(w, d) \): number of times \( w \) occurs in \( d \)
- \( N(d) \): number of words in \( d \)
- \( |D| \): number of documents in the dataset
- \( |D^*| \): number of documents where \( w \) occurs at least once

This measure gives more weight to terms occurring often in the given
document but not in many other documents in the set, and it is the
key concept that PageRank [9]\(^2\) is based on. It also normalizes the term
frequency with respect to the document length.

### 2.1.3 Divergence measure

Let us introduce also the most common divergence measure, that is a
representation of the "distance" between two different statistical distri-
butions. Divergence, in statistics, is a weaker notion than distance \(^3\),
since it does not require symmetry nor to satisfy the triangle inequality.
Jeffrey divergence [46] is the most used divergence measure in NLP:
it is a numerically stable and symmetric form of the Kullback-Leiber
metric [32] and, for two frequency vectors \( x \) and \( y \), is defined as:

\[
div_{je}(x, y) = \sum_i x(i) \log \frac{x(i)}{\frac{x(i) + y(i)}{2}} + y(i) \log \frac{y(i)}{\frac{x(i) + y(i)}{2}} \quad (2.4)
\]

\(^2\)Described in next section, it is has been the first algorithm used by Google Search
to rank websites in their search engine results. It takes its name from Larry Page,
Google co-founder and Alphabet Inc. CEO, which developed it in his PhD
\(^3\)https://ipfs.io/ipfs/QmXoyPizjW3WknFjKnKlwHCnL72vedxjQkDPP1mXWo6uco/wiki/Divergence_(statistics).html
2.1.4 Tokenization, POS tag and NER

In natural language processing, tokens, Parts-of-Speech (POS) and named entities are often a basis for solving more complex tasks. Tokenization is a process aiming to identify and segment text sections such as words and sentences. Depending on the context, different delimiters should be considered. For example, the dot is usually a sentence delimiter, but it is not the case when it is part of an acronym. A POS is a category of words with the same syntactic function (e.g., nouns, pronouns, verbs)\(^4\), which can be further divided into subcategories. The same word can have different meanings, hence the context is again crucial to perform a correct classification. Named Entity Recognition (NER) is instead a process aiming to classify named entities given a set of pre-determined categories (like name, location, organization)\(^5\). The state-of-the-art models for such tasks are nowadays achieving performance close to the human ones, and are implemented in several libraries which are introduced later (chapter 2.1.7).

2.1.5 Text preprocessing

A descriptive summary is mainly depending on the quality of the information extraction procedure, which tries to identity the most important utterances to include in the final summary. For this reason, a common preprocessing step consists of removing the stopwords (e.g., a, the, and) from the source text, since they are usually not informative, and then applying the Porter stemming [45]. Stemming is a process that removes the more common morphological and inflectional endings from words: for example, family and families will both become famili. It becomes extremely helpful when dealing with term frequency and utterance similarity. In this work, I will refer to this procedure as "usual preprocessing" or "standard preprocessing", since it has been widely used in literature. In the proposed method, stemming is replaced by lemmatization\(^6\), which is described later in chapter 3.1.

\(^4\)Most common parts of speech: noun, verb, adjective, adverb, pronoun, preposition, conjunction, interjection, article

\(^5\)See appendix A for the whole list of types and correspondent descriptions used in this project

\(^6\)https://spacy.io/usage/linguistic-features#section-pos-tagging
2.1.6 Topic modelling

Topic modelling is a technique to extract abstract "topics" from a collection of documents\(^7\). A topic model is a kind of probabilistic generative model that has been used widely in the field of computer science. Commonly, it is used to determine what topic a document is most likely about. Alternatively, it can generate word-topics distributions, that is the probability of a word to belong to each topic in the model.

An intuitive overview of topic modeling is presented in [34], which the reader can refer to for an introduction of the most common topic models like Latent Semantic Indexing (LSI) [15], Probabilistic Latent Semantic Analysis (PLSA) [29] and Latent Dirichlet Allocation (LDA) [5]. They have been effectively used in text mining and information retrieval in several recent works (Online LDA [2]\(^6\), PLDA [48]\(^9\)). Various works have then extended these models, adapting them to specific fields (social media [44]\(^10\), image annotation [17]\(^11\)).

There are two ways to build and use a topic model. A corpus is always needed for training the model. Training corpora used in the experiments are introduced and described in chapter 4.1. Then, depending on the task the model is used for, a new document can be analyzed, but can also be used for re-training the model itself. In this work, the topic model will be used to compute the topical similarity between utterances.

2.1.7 NLP open source libraries

Many packages are available for approaching NLP in Python, but the most used ones for learning and building up new solutions are Natural Language Toolkit (NLTK) [4], Stanford's CoreNLP[35], Gensim [49] and SpaCy [30]. They are all free and open source projects.

\(^7\)http://text2vec.org/topic_modeling.html#example
\(^6\)On-line LDA: Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking
\(^9\)Partially labeled topic models for interpretable text mining
\(^10\)Investigating topic models for social media user recommendation
\(^11\)Topic models for image annotation and text illustration
The former one is described, on its website, as "an amazing library to play with natural language". However, it has not been developed for production, but more for teaching and learning purposes. The Stanford CoreNLP is a Java library that can be quite easily integrated in a Python script, and supports the six most spoken languages. Gensim is an open source Python toolkit for vector space and topic modelling, and it is the easiest and most efficient library to deal with topic models. Hence, it will be used for that specific task in this work.

Finally, SpaCy is a new Python library implementing the current state of the art of most NLP algorithms. It is the fastest, most efficient and best performing library currently available, together with an excellent user guide, and it is really easy to start working with. Several large companies have recently built their solutions using SpaCy for the reasons just mentioned above. Therefore, this work has largely used SpaCy, e.g. for POS\textsuperscript{12} and NER\textsuperscript{13} tagging as well as for tokenization, and this will be assumed in next chapters. Else, the libraries used will be explicitly stated.

### 2.1.8 Precision, Recall and F-measure

In Computer Science, Precision (or positive predictive value) - \( P \) - is the fraction of relevant instances among the retrieved instances, while Recall (or sensitivity) - \( R \) - is the fraction of relevant instances that have been retrieved over the total amount of relevant instances\textsuperscript{14}. In other words, precision says how many selected items are relevant, while recall says how many relevant items are selected. F-measure (also \( F_1 \) score or F-score) - \( F \) - is the harmonic average of the precision and recall\textsuperscript{15}:

\[
F = \frac{2 \cdot P \cdot R}{P + R}
\]  

\textsuperscript{12}https://spacy.io/usage/linguistic-features#section-pos-tagging
\textsuperscript{13}https://spacy.io/usage/linguistic-features#section-named-entities
\textsuperscript{14}https://en.wikipedia.org/wiki/Precision_and_recall
\textsuperscript{15}https://en.wikipedia.org/wiki/F1_score
2.2 Extractive methods

Extractive summaries have been dominating the research world for many years, mainly due to the complexity of generating novel text from important utterances. Like in most fields, supervised methods are outperforming the unsupervised ones, despite the small amount of annotated data available. However, most supervised methods have only been tested on the corpus used for training them, creating some doubts about their effectiveness when applied to other meeting domains. Let us now introduce the most common unsupervised methods for text summarization, and their correspondent adapted versions for dealing with the meeting structure.

2.2.1 Binary classification

The extractive summarization task can be approached as a binary classification problem, where each sentence is marked as important or non-important and consequently included or not included in the final summary. The most efficient supervised algorithms proposed for generic summarization are based on linear-chain Conditional Random Fields (CRF)\textsuperscript{16} [51], where the problem is seen as a sequence labeling problem, and on Supported Vector Machine\textsuperscript{17} (SVM) [27], where both linguistic and syntactic features are extracted from the training dataset. As explained in the introduction, performance degrades drastically when they are applied to meeting transcripts compared to structured texts. Hence, a speech-feature-based summarizer has been proposed [40]. The authors used basic prosodic features like energy, duration and frequency of each utterance together with the simple \textit{tfidf}. A Gaussian mixture model is then used to identify the important sentences, showing noticeably better performance compared to the "standard" methods.

\textsuperscript{16}Introduction to CRF: http://blog.echen.me/2012/01/03/introduction-to-conditional-random-fields/

\textsuperscript{17}Introduction to SVM: https://towardsdatascience.com/support-vector-machines-a-brief-overview-37e018ae310f/
2.2.2 Graph-based methods

The idea of transforming a text into a graph for representing lexical relations between utterances has been widely used in NLP. The most used graph-based summarizer is TextRank [37], in which nodes represent sentences and edge weights are proportional to the number of common words between the two nodes\(^{18}\). Each node is then ranked through PageRank [9], the first algorithm used by Google Search to rank websites. However, it is not suited for highly redundant texts [42], despite its good performance when applied to properly structured ones.

Therefore, researchers from International Computer Science Institute (ICSI) and Ecole Polytechnique Federale de Lausanne (EPFL) have proposed ClusterRank [23]. It extends TextRank to overcome the spontaneous nature of meetings by clustering utterances together and using such clusters, which are nothing but text segments, as nodes of the algorithm. Clustering is performed by using a simple approach similar to TextTiling [26], which merges utterances addressing the same argument. It merges clusters according to a given threshold (tuned on a small development set). Their similarity is determined using the cosine distance (eq. 2.2) of the words they contain, where each word weight is proportional to its tfidf (eq. 2.3). Sentences are just ranked summing up words weights, and the top ranked one in each cluster is added to the final summary.

Pretty similar to TextRank is LexRank [16], which has been developed in parallel by other researchers, where the only difference compared to the previous approach is that the edge weights are proportional to the semantic similarity instead of the lexical similarity. It can be adapted to meeting transcripts using the same clustering approach.

Considering the capability of the graph approach to generate summaries, researchers of Carnegie Mellon University have proposed a multi-layer graph for improving meeting summarization [13], i.e. a three-layer graph composed of utterance, hidden and speaker layers. After the usual preprocessing, the authors face the utterance selection problem by computing the importance of each utterance. Each ut-

\(^{18}\)Lexical similarity
utterance is represented by an utterance node, edges are weighted according to topical similarity and each speaker is a node in the speaker layer. The multi-layer graph is constructed such that the utterance importance is determined propagating weights with a random walk through the graph. It is mathematically proven that the weights will converge. The smartest innovation is the introduction of an intermediate layer representing terms or latent topics. The key point is that the propagated scores additionally consider speaker information, which is automatically modeled via hidden parameters in the graph. Experiments enhance that this architecture can model the importance of utterances and speakers through hidden parameters in the multi-layer graph, showing consistent relative improvement compared to some baselines and to the same random-walk applied to a two-layer graph, which is only composed of utterance and speakers nodes. Since this work has only been tested on their own corpus, it has not been compared with other methods.

2.2.3 Cluster-based methods

An interesting clustering method was proposed in 2006 [1]. It clusters sentences by maximizing intra-cluster similarity while minimizing inter-cluster similarity. Then, it selects a representative sentence for each cluster depending on their proximity to the cluster center, composing the final summary. The authors state that it works with structured text, but it is not sophisticated enough to be effective on meeting transcripts too. For this reason, to the best of my knowledge, purely cluster-based methods have never been applied in the meeting domain.

2.2.4 Ranking-based methods

Ranking-based methods aim to rank utterances according to sentence level and word level features. The most common ranking-based approach for text summarization is the maximum marginal relevance (MMR) [10]. It assigns a score to each text unit depending on its similarity with a query and with the sentences already in the summary. Then, the top ranked sentence is added and the procedure iteratively
repeated, constrained in terms of summary length. As usual, it needs some changes to be efficient when dealing with the unique features of meetings, and a new similarity measurement has been introduced in [55]. It basically counts the number of times a word appears close to each word in the corpus and then integrates this word-level information to score each sentence.

A modern approach using discourse information is proposed to find meeting keywords [8]. It provides the base for extracting sentences in the current state of the art, which will be presented in the next subsection. This method first segments the meeting transcript by applying functional segmentation, aiming to divide it into subsections and partitioning into monologues and discussions, and to determine the important participants involved in each section. For more on functional segmentation, the reader can refer to [6]. The main idea is that there is a significant change in the speakers involved when changing segment, and the method tries to find these points. Furthermore, only specific POS are allowed to be keywords. After segmenting the text, it computes idf on the whole transcript and a segment entropy score (later introduced in eq. 3.14). It then determines a probability distribution for each word by summing the global and local distributions. If the probability distribution has some peaks, it is likely an informative word and is highly ranked. The most highly ranked words are the summary keywords. Their experiments show that graph-based summarization methods are not so accurate, and this method outperforms the other existing extractive ones.

2.2.5 Unsupervised extractive state of the art

As prompted in the future works in [8], the authors extended the previous method by proposing an algorithm that generates extractive summaries of meeting transcripts based on functional segmentation [7]. After the usual preprocessing, it segments the text into functionally coherent parts using the algorithm proposed in [6], dividing it into two main categories: monologues and dialogues. Then, they are summarized with two different approaches. Figure 2.1 shows the algorithm steps at high level.
In the segmentation step, it first finds candidate boundaries, i.e. all the words that might correspond to the end of a segment. Then, it determines the global best segmentation, which is the combination of a subset of candidate boundaries, generating the desired number of segments. Those segments are as close as possible to what is defined as an ideal distribution\textsuperscript{19}. This leads to a segmentation really similar to the manually segmented reference. The last segmentation step requires to label each segment finding the respective active speakers\textsuperscript{20}. Before summarizing, all the sentences by participants not marked as active speakers are removed, since they should probably not be part of the summary. Whereupon, the proposed method summarizes monologues and dialogues with two different approaches.

Monologues are summarized by extracting local keywords [8] according to document and segment level scores. Their weights are the input of an Integer Linear Programming\textsuperscript{21} (ILP) algorithm. It aims to maximize the weight of the keywords in the final summary under the length constraint.

\textsuperscript{19}In each segment, all the active speakers (defined in the next footnote) are equally involved.

\textsuperscript{20}A person is considered as an active speaker in a segment if its sentences are needed for understanding the segment meaning.

\textsuperscript{21}An Integer Programming problem is a mathematical optimization or feasibility program in which some or all of the variables are restricted to be integers. In many settings the term refers to integer linear programming, in which the objective function and the constraints (other than the integer constraints) are linear.

https://en.wikipedia.org/wiki/Integer_programming
Dialogue summaries are instead obtained by applying the reinforced random walk method [13] on a three-layer directed graph. Edge weights between utterance, topic and speaker layers are proportional to the two neighbor nodes' topical or lexical similarity. The random walk propagates the weights until the utterance scores have converged, and the top ranked ones are extracted. The final summary is prepared by first generating a longer summary for each segment (since the same compression ratio might not be the optimal solution for all segments), and finally they are taken as input for generating summaries of desired length and finally redundancy reduction [10] is applied to the output.

Experiments, evaluated with various metrics, show that this method outperforms previous state-of-the-art extractive summarizers and it is, to the best of my knowledge, the best unsupervised extractive algorithm currently available. Due to it being fully unsupervised, it does not require any annotated data except for evaluation.
2.3 Abstractive methods

In the last decade, several abstractive methods have been proposed to overcome the problems presented in the previous section, following the way suggested in the user study in [38]. The performance of such summarizers depends both on the selection of the important utterances and on the fluency and grammatical correctness of the generated text. A well written summary is useless if is generated based on non-important utterances. Furthermore, if the fluency of the novel text is worse than the extracted summary, there is no reason for applying an abstractive algorithm.

2.3.1 Graph-based methods

As in the extractive approach, good results have been obtained with graph-based methods. In [36], the authors propose a supervised word graph approach, extending the multi-sentence compression method presented in [18]. They first cluster sentences together by identifying pairs of sentences that can be summarized with a common sentence, and then it identifies which sentences can be clustered together. For each group, a multi-directional entailment graph is built, with sentence pairs as nodes and entailment relations\textsuperscript{22} as edges. Like in many other approaches, a multi-sentence fusion technique is applied to each cluster to generate the abstract. A word graph is constructed over sentences selected in the previous step, and then the valid paths\textsuperscript{23} are ranked by maximizing fluency and coverage. The top one becomes the abstract sentence summary. More details will be presented when describing the current state of the art, which is again based on a sentence fusion technique.

2.3.2 Focused summarization

Sometimes, users are not interested in a whole summary, but instead they want to summarize only particular aspects of a meeting like decisions, actions and ideas. Eventually, by merging those parts, it is possible to obtain an abstract covering most of the relevant points discussed in a meeting. For this reason, an unsupervised framework for

\textsuperscript{22}Semantic relations
\textsuperscript{23}Valid paths are the ones containing at least a verb
focused meeting summarization has been developed in [54], and is based on the hypothesis that existing methods for domain-specific relation extraction can be modified to identify salient phrases. In their work, the authors focus on the task of decisions summarization and adapt an unsupervised relation-learning approach [12] to identify decision cues and decision content, by defining a new set of task-specific constraints and features. They demonstrate that their method outperforms two extractive baselines, gives similar results compared to another relation-based technique and that its performance was close to two supervised learning alternatives, which require labeled data to retrain for each new corpus.

2.3.3 Template-based methods

Considering the intrinsic difficulties of generating a novel text from important utterances, different groups of researchers ([53], [43]) proposed to first create an abstract template, and then to fill it with utterances extracted from the source. Based on the concept of focused summarization, together with developing a novel extraction algorithm called Multiple-Sequence Alignment (MSA) [53], the authors generate templates that can be used for different domains. This method first performs a content selection, by training a classifier to identify summary-worthy phrases. Then, it generates and ranks candidate sentences for the abstract, and after a redundancy reduction the full meeting abstract comprises the focused summary for each meeting element. The authors prove its domain independence by yielding comparable results when the system is trained and tested on the same corpus as well as when it is trained and tested on two different ones.

In [43], the algorithm first acquires templates from human-authored summaries, then it segments the transcript based on topics by extending a method for discourse segmentation of multi-party conversation [22]. After extracting important phrases for each topic, it selects the most suitable templates by referring to the relationship between human authored summaries and their sources. Finally, it fills them with the phrases to create summaries. This method demonstrates the adaptability of a word graph algorithm to generate templates and presents a novel template selection algorithm that effectively leverages the rela-
tionship between human-authored summary sentences and their source transcripts. Evaluation shows that this system successfully creates informative and readable summaries.

### 2.3.4 Abstractive state of the art

The most effective abstractive summarizer, proposed in [3], is based on utterance fusion and is divided into three main sections: text segmentation, classification of important utterances and utterance fusion for summary generation. An intuitive scheme is shown in figure 2.2.

![Diagram of abstractive summarization method](image)

Figure 2.2: Abstractive summarization method presented in [3]

The transcript is first segmented by topics with LCSeq [22], while the classification of important utterances for each segment is based on basic features\(^{24}\), content features\(^{25}\) (both adopted from previous works [21], [55]) and two new segment-based features introduced by the authors: most important speaker\(^{26}\) in each segment and cosine similarity between the dialogue and the entire segment. Content words include nouns, adjectives, verbs and adverbs. The authors chose a Random Forest\(^{27}\) constructed on the training set, with resampling to oversample the minority data. This limits the unbalanced data problem caused

\(^{24}\)Length of a dialogue, number of content words, portion of content words and number of new nouns introduced

\(^{25}\)Cosine similarity with entire meeting transcript, presence of proper nouns, most important speaker in meeting and content words in previous dialogue act

\(^{26}\)The speaker uttering the maximum number of words

\(^{27}\)Introduction to Random Forest:
https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd/
by the intrinsic meeting structure, in which some topics are discussed for much more time than other important ones.

Finally, it has to combine information from multiple utterances extracted in each segment to generate a sentence per topic. They adapt a sentence fusion technique [19] to the meeting domain. The dependency parse trees of each utterance are merged together, and the best sub-tree satisfying the constraints and maximizing the information propagation is selected using an ILP formulation. Pronoun resolution\textsuperscript{28} is shown to produce more understandable summaries. The ambiguity resolver helps to correctly merge dependency trees where both have the same word with a different meaning. Finally, the merged tree is linearized to produce a single sentence, as introduced in [20].

It is shown that it overcomes the previous state of the art on abstractive summarization of meeting transcripts and it is the best method currently available, to the best of my knowledge. The experiments on content selection and readability indicate that their method can generate relevant abstractive summaries from meeting transcripts without any templates. However, not all generated summaries are usable due to the lack coherence among several entities discussed within the same summary sentence. It also requires supervised learning for extracting important utterances, therefore it would be interesting to test its effectiveness on other meeting transcripts, since all of the training corpus is composed of meetings discussing the same problem.

\textsuperscript{28}Replace pronouns with the nouns they refer to, since the utterance with the explicit noun might not be included in the final summary
Chapter 3

Proposed Method

The proposed method is composed of three main steps and takes inspiration from the current unsupervised extractive state of the art [7]. It introduces new features to improve the extractive procedure. As already stated, the proposed framework is fully unsupervised and domain independent, in order to create a general basis for further improvements given more information about the meeting agenda, the company or some specific aspects a user may want to summarize. Figure 3.1 shows the main sections of the proposed method, which is briefly introduced here and described in detail in the upcoming sections.

Figure 3.1: Proposed method: high-level representation
After the preprocessing step, consisting of removing stopwords and applying the lemmatization implemented in SpaCy\(^1\), the functional segmentation divides the transcript into monologues and dialogues, combining local and global information\(^6\). It also determines the active speakers for each segment.

Then, based on the idea suggested in\(^7\), monologues are summarized by applying an ILP formulation on each segment. It maximizes the sum of the keyword scores from all sentences included in the final summary of that segment. Keywords are extracted according to the method explained in\(^8\), considering the whole meeting transcript and not only monologues. Since text units labelled (with SpaCy NER tagger\(^2\)) as organizations, locations, products and events are usually uttered in relevant parts of the meeting, the weights of such words are increased. As well, it considers the most likely topic for the document, by analyzing it with the topic model. The weights of words with the highest probability to belong to such topic are again increased, introducing new features in the extractive framework. Depending on the amount of keywords extracted and the desired summary length, the algorithm will generate a more compressed or informative resume.

The selection of relevant sentences in dialogues is approached with a reinforced random walk through a two-layer (utterance and speaker) graph, as in\(^{14}\), in which feature similarities are computed to be assigned to the edge weights. Here, named entity tags affect the initial utterance importances. In chapter 3.4, equations regarding graph construction and utterance importances propagation are presented.

---

\(^1\)https://spacy.io/usage/linguistic-features#section-pos-tagging
\(^2\)https://spacy.io/usage/linguistic-features#section-named-entities
3.1 Preprocessing

In the preprocessing step, the stopwords are removed. The list of stopwords is imported from the NLTK Corpus\(^3\), expanded with the addition of utterances typical of the spoken language like *uhm*, *mmh* and so on\(^4\). Then, lemmatization is applied to reduce inflectional forms and transform related forms of a word to a common base. This approach is here preferred over the stemming approach. The former achieves better results, since it uses a vocabulary and performs morphological analysis of words, while the stemming just chops off the end of the words in the hope of achieving its goal correctly most of the time\(^5\).

---

\(^3\)https://github.com/nltk/nltk/tree/develop/nltk/corpus

\(^4\)Stop words specific for meeting domain introduced in this project are: "um", "uhm", "mm", "mmm", "yeah", "ehm", "mmh", "uhmm", "ah", "m", "re", "s", "ve", "hmm", "mm-hmm", "uh", "blah", "bah", "okay", "ok"

3.2 Functional segmentation

The functional segmentation step aims to divide the transcript into monologues and dialogues and identify the active speakers. The intrinsic differences of those segments have advised the researchers in [7] to use two different approaches for generating their summaries. See appendix B for an example of manual functional segmentation of a meeting transcript.

Briefly, the method proposed in [6] first determines a set of candidate boundaries\(^6\), filtering out only the ones which will not belong to the final boundary set with high confidence. Then, it finds the best combination of candidate boundaries that divides the source transcript in the desired number of segments \(N_s\), where the speaker distribution\(^7\) in all segments is as similar as possible to the ideal one. The ideal distribution of speakers is the one where all the active speakers are equally involved, meaning that they utter the same amount of non-stopwords in the segment. It has been proven in [6] that minimizing the distance between the real and the ideal distribution leads to a segmentation really close to the human one. Figure 3.2 shows the steps used for segmenting the meeting transcript.

---

\(^6\)Each sentence is initially a possible boundary. The set of candidate boundaries (CB) is a vector of indices of all the sentences that might be on the border of a segment. Sentences are indexed by their position in the transcript

\(^7\)Vector of size \(\text{num\_speak}\) where each element is the number of words uttered by \(j^{th}\) speaker divided by the total number of words in the segment
3.2.1 Candidate boundaries

Considering each sentence as a candidate boundary, a score value per candidate is computed, extracting features from two windows of length \( L_{\text{win}} \) placed before and after the sentence. The contribution of a speaker \( j \) in the window \( \text{win} \) is computed as:

\[
WC_j(\text{win}) = \frac{\text{num\_word}(j, \text{win})}{\sum_{k \in K} \text{num\_word}(k, \text{win})}
\]  

(3.1)

where \( \text{num\_word}(j, \text{win}) \) is the number of words uttered in the window by speaker \( j \), and \( K \) is the set of all the attendants. The contribution of all speakers constitutes the vector \( WC(\text{win}) \), of length \( \text{num\_speak} = N_j \).

The importance of words uttered by a speaker \( j \) is instead measured according to the \( \text{suidf} \) metric, introduced in [39]. It is preferred to a more conventional metric like \( \text{tfidf} \), since it also considers the word’s usage among the participants. It is calculated by first computing the surprise of the participant \( j \) uttering the word \( w \) as:

\[
surp(j, w) = -\log \left( \frac{\sum_{j' \neq j} tf(w, j')}{\sum_{j' \neq j} N(j')} \right)
\]  

(3.2)

where \( N(s) \) is the number of words uttered by speaker \( s \), and \( tf(w, s) \) is the number of times that \( s \) utters \( w \). If \( w \) is commonly used, then its score with respect to any speaker will be low. The surprise of a word is then given by averaging its scores over all the speakers in the meeting:

\[
surp(w) = \frac{1}{|K|} \sum_{j \in K} surp(j, w)
\]  

(3.3)

Finally, the \( \text{suidf} \) of a word is computed as:

\[
\text{suidf}(w) = surp(w) \cdot \sqrt{idf(w)} \cdot \frac{s(w)}{|K|}
\]  

(3.4)

where \( idf(w) \) is the inverse document frequency described in chapter 2.1.2, and \( s(w) \) is the number of participants that uttered \( w \) at least once. The reasons for using this metric are described more in detail in [39].

---

\(^{9}\)In this approach, a window is just a group of sentences
in which experiments demonstrate the effectiveness of such empirical choices. Now, it is possible to compute the words' importance in the window:

\[ WI_j(\text{win}) = \frac{\sum_{w \in \text{Words}(j, \text{win})} \text{stidf}(w)}{\sum_{k \in K} \sum_{w \in \text{Words}(k, \text{win})} \text{stidf}(w)} \]  

(3.5)

where \( \text{Words}(j, \text{win}) \) is the set of words uttered by \( j \) in the window. \( WI(\text{win}) \) is, again, a vector of length \( \text{num}\_\text{speaks} \).

Finally, the score of each candidate boundary is defined as the Jefrey divergence (eq. 2.4) between the features extracted from the left (\( \text{win}_i, \text{left} \)) and right (\( \text{win}_i, \text{right} \)) windows:

\[ \text{score}[i] = \text{div}_J[\text{WC}(\text{win}_i, \text{left}), \text{WC}(\text{win}_i, \text{right})] + \text{div}_J[\text{WI}(\text{win}_i, \text{left}), \text{WI}(\text{win}_i, \text{right})] \]

(3.6)

![Figure 3.3: Score plot of the first 100 sentences from the meeting ES2004a](image)

The score plot's local maxima form the candidate boundary set:

\[ CB = \{i \mid \text{score}(i) > \text{score}(i - 1) \quad \& \& \quad \text{score}(i) > \text{score}(i + 1)\} \]  

(3.7)

From this set, and given the estimation of the final number of segments, a dynamic programming approach can be used to determine the best combination of the CB generating the desired number of segments. The number of segments is estimated by first smoothing the score function:

\[ \text{score}(i) = \frac{1}{s + 1} \sum_{j=i-s/2}^{i+s/2} \text{score}(j) \]

(3.8)

to get rid of the small variations in the score plot which do not represent a significant change in the active speakers, where \( s \) is a smoothing
integer parameter.

Then, it calculates the peak score function to measure the relative strength of the change at \( cp \) with respect to the contiguous scores:

\[
peak\_score(cp) = 2 \cdot score(cp) - score(pw) - score(nw)
\]  
\[(3.9)\]

In eq. 3.9, \( cp \) means candidate point and \( pw \) and \( nw \) represent the adjacent local minima in the smoothed score plot. The number of locations where \( peak\_score(i) \) is higher than a multiple \( (k_{peak} = 1, 2, ...) \) of the difference between the average \( (s) \) and the standard deviation \( (\sigma) \) of all \( peak\_scores \), gives the estimated number of segments \( N_s \):

\[
N_s = |\{|peak\_score(i) > k_{peak} \cdot (s - \sigma)\}| + 1
\]  
\[(3.10)\]

The dynamic programming has a high computational cost \( (O(L \cdot L_{CB}^2)) \), where \( L \) is the total number of sentences. Therefore, the points \( cp \) where eq. 3.10 is fulfilled can be considered as boundaries, if the user does not aim for achieving the optimal performance but rather prefers a faster solution. In this work, this lighter solution is used, since it does not introduce any feature on top of the existing method. However, let us present the method proposed for obtaining the optimal segmentation.

### 3.2.2 Optimal segmentation

Here, candidate boundaries, number of speakers and the preprocessed sentences are used as input of the dynamic programming, to find the best combination of CBs for segmenting the transcript. Let \( dstr(s) \) be the distribution of speakers in segment \( s \), i.e. a vector expressing the
fraction of words uttered by each speaker, $dstr_{id}(N_s, cat)$ be the ideal distribution for the category $cat$, where a category is e.g. $\{1,0,0,1\}$ if speakers 1 and 4 are the only active ones, and $|K|$ be the number of speakers in the meeting. The segment $s$ is partitioned into subsegments of length $M = 2 \cdot |K|$, and the segment score is computed as:

$$score_{seg}(s) = \frac{1}{L - M} \min_{cat \in C} \sum_{i=1}^{L-M} \{div_{jeff}(dstr(s[i : i + M]), dstr_{id}(|K|, cat))\}$$

(3.11)

where $L$ is the number of sentences in the segment, and $C$ is the set of all possible active speaker categories. This equation computes the score segment using the minimum distance between the real speaker distribution and its closest ideal one. The pseudo-code in 3.6 shows the dynamic programming approach to determine the best global segmentation extracted from the candidate boundary set.

### 3.2.3 Segments labelling

Finally, each segment $s$ is labelled with the category for which the Jeffrey divergence (eq. 2.4) between its ideal distribution and the segment distribution is minimized:

$$cat(s) = \min_{cat \in C} \left( div_{jeff}(dstr(s), dstr_{id}(cat, N_s)) \right)$$

(3.12)
**Input:** seq (Input sequence with length \( L \)), CB (Candidate boundary set with size \( L_{eq} \)), Num\(_{seg}\) (Number of boundaries).

**Initialization:**
for \( i = 1 \) to \( L_{eq} \) do
    \( sb(i, 1) = \text{score}_{eq}(1 : CB_i) \);
    \( bp^*(i, 1) = 0 \);
end for

**Recursion:**
for \( i = 2 \) to \( Num_{seg} \) do
    for \( k = 2 \) to \( L_{eq} \) do
        \( sb(i, k) = \min_{j < i} \{ sb(j, k - 1) + \text{score}_{eq}(CB_j + 1 : CB_k) \} \);
        \( bp^*(i, k) = \arg \min_{j < i} \{ sb(j, k - 1) + \text{score}_{eq}(CB_j + 1 : CB_k) \} \);
    end for
end for

**Backtracking:**
\( n = Num_{seg} \);
boundaries ← \{ \};
current ← \( bp^*(L_{eq}, n) \);
while current > 0 do
    Add \( CB_{current} \) to boundaries;
    \( n ← n - 1 \);
    current ← \( bp^*(current, n) \);
end while

**Output:** boundaries

Figure 3.6: Algorithm for finding the best boundary combination [6]

Here, the segment is not divided anymore into subsegments of length \( M \). The category states who are the active speakers per segment, and all the utterances from the other speakers are filtered out, assuming they are most of the times not so important to be included in the final summary. For an extended analysis of this method, together with parameter settings and performance evaluation, the reader can refer to [6]. However, in the next chapter the parameters setting are recalled and their effects are described and explained.

The next images show a segmentation example taken from a section of meeting E52004a. The example takes a limited portion of the transcript, but the reader should be aware of that the results depend on a global analysis. Figure 3.7 shows a portion of text where candidate boundaries are marked as “CANDIDATE BOUNDARY”. Figure 3.8 shows the local result of the global optimal segmentation, by marking the selected boundaries in dark gray and in bright colours sentences from non-active speakers.

The global segmentation has found a monologue of speaker 3 and a dialogue between speakers 2 and 3. The underlined candidate boundary in figure 3.8 shows a candidate boundary that would have probably been selected with a local analysis, dividing the dialogue between
speakers 2 and 3 into two independent monologues. However, since this is the result of a global optimization with predefined number of segments, this is the combination that minimizes the global distance between the computed segmentation and the ideal one\(^9\). Once the transcript has been divided into monologues and dialogues, two different extractive approaches are applied, to exploit the different characteristics of the utterances in those categories.

---

\(^9\)Recall: ideal segments are the ones where active speakers are equally involved
3.3 Monologues summarization

In monologues, there is no change in the active speaker involved, hence it is more similar to a sermon or written news. The effectiveness of optimization approaches for meeting summarization has already been demonstrated in different works ([24], [56]). An ILP formulation is proposed, and it aims to maximizing the weights of the contained important concepts constrained by the maximum length allowed.

3.3.1 Keyword extraction

Meeting keywords are extracted with a technique extending the one presented in [8], which sums a document level and a segment level score to rank each candidate word, i.e. all the non-stopwords that are nouns, verbs and adjectives\(^{10}\). Note that the keywords extraction is performed over all the meeting and not only on the monologues. The document level score proposed is the basic \(idf\), since in [8] the authors show that it outperforms \(tfidf\) (chapter 2.1.2).

The segment level score is the word’s entropy among the functional segmentation previously determined. It first calculates the probability of being in a segment \(s_j\), given the word \(w\):

\[
p(s_j|w) = \frac{n(w,s_j)}{N_w}
\]

(3.13)

where \(n(w,s_j)\) is the number of times \(w\) is uttered in \(s_j\). Then, the negated entropy of a word is:

\[
-H(w,S) = \sum_{s_j \cap w \neq \emptyset} p(s_j|w) \cdot \left(\log p(s_j|w)\right)
\]

(3.14)

where \(S\) is the set of segments. This score gives more importance to the words that are used only in a few segments, since words used evenly in all segments are probably not so important for the specific segment. This work introduces a weighted sum, for which frequency \((k_{MF})\) and entropy \((k_{ME})\) coefficients will be tested:

\[
\text{score}(w) = k_{MF} \cdot idf(w) + k_{ME} \left( -H(w,S) \right)
\]

(3.15)

Furthermore, by applying the pre-trained SpaCy NER tagger on the original transcript, the weight of words labelled as *money, event, organizations, person* and so on are multiplied by a coefficient $K_{NER}$. In appendix C, all test cases with their parameters and results are reported in detail.

![Figure 3.9: Example of entities tagged with SpaCy](image)

Finally, the most likely topic $T_{max}$ for this meeting is determined through the LDA topic model introduced in chapter 2.1.6. It is trained on the meeting corpus together with the BBC news collection (from now, let us call the training dataset $D_{LDA}$). The probabilities $prob_{w,T_{max}}$ of words to belong to $T_{max}$ are normalized by dividing them by the probability $maxprob_{T_{max}}$ of the most probable word with respect to $T_{max}$:

$$P_{w,T_{max}} = \frac{P_{w,T_{max}}}{maxprob_{T_{max}}}, \forall w \in D_{LDA}$$  \hspace{1cm} (3.16)

All the words with probability higher than a threshold $T_i$ are considered *topic keywords*. Note that *meeting keywords* and *topic keywords* are not the same thing. The weights of *topic keywords* are multiplied by a coefficient:

$$k_{w,T_{max}} = i_T + k_T \cdot P_{w,T_{max}}$$  \hspace{1cm} (3.17)

The final word’s score is given by multiplying the word’s score (eq. 3.15) with its named entity coefficient and its topic keyword’s weight (eq. 3.16):

$$final\_score(w) = score(w) \ast k_{NER}(w) \ast k_{w,T_{max}}$$  \hspace{1cm} (3.18)

The top - $T_m$ - ranked words are the meeting keywords.

### 3.3.2 ILP formulation

An optimization problem is set up to determine the most important sentences, with the goal of maximizing the weights of all the keywords included in the monologue summary. The objective function is constrained by the compression ratio, calculated considering only
non-stopwords. Mathematically, the solution is computed according to the following ILP formulation:

\[
\text{obj: } \quad \max \sum_i w_i c_i \\
\text{s.t. } \quad \sum_j s_j l_j \leq L \\
\quad \quad \quad \quad s_j a_{ij} \leq c_i \quad \forall i, j \\
\quad \quad \quad \quad \sum_j s_j a_{ij} \geq c_i \quad \forall i \\
\quad \quad \quad \quad s_j, a_{ij}, c_i \in \{0, 1\}
\]

(3.19)

where \(s_j\) is a binary indicator variable denoting if the \(j^{th}\) utterance is included in the summary and \(l_j\) is the length of the \(j^{th}\) sentence. \(c_i\) is \(1\) when the \(i^{th}\) keyword, with weight \(w_i\) determined in the previous step, is part of the summary, while \(a_{ij}\) is \(1\) if the keyword \(w_i\) appears in the utterance \(j\) and \(L = \text{num. words} \cdot L_m\%\) is the maximum length allowed. The optimal values for \(s_j\) state which utterances from the source transcript\(^{11}\) must be included in the final summary. Image 3.10 shows an example of a summarized monologue from TS3005a, where pink sentences mark the correctly extracted sentences, while the missing sentences are marked in blue.

\(^{11}\) If the reader wants to implement this method, he should remember to keep track of the index of the original sentences, since some short ones may be composed only by stopwords, therefore totally filtered out in the preprocessing step.
3.4 Dialogues summarization

The dialogue summarization is approached by extending the two-layer graph method presented in [14]. It represents the meeting transcript in a two-layer graph - utterance and speaker layers -, then it applies a random walk to propagate the importance of each utterance through the graph. Edge weights in the utterance layer are proportional to the sum of topical and lexical similarity of the connected utterances, and the propagation of the importance of the utterances is affected by those parameters as well as the speaker information. The edge weight computation is described in the next subsections, along with the topic model used, as well as the utterance importances initialization and their propagation through the graph.

![Two-layer graph simplified example with three speakers and seven sentences](image)

Figure 3.11: Two-layer graph simplified example with three speakers and seven sentences [14]

3.4.1 Topical similarity

LDA is the topic model used for computing various probabilities, which then will be used for determining the topical similarity of two utterances:

1. \( P(t_i|d_j) \): probability of a topic given a document
2. \( P(t_i|w_j) \): probability of a topic given a word
3. Probability that topic $T_k$ is addressed by an utterance $U_i$:

$$P(T_k|U_i) = \frac{\sum_{w \in U_i} n(w, U_i) P(T_k|w)}{\sum_{w \in U_i} n(w, U_i)}$$  \hspace{1cm} (3.20)

4. Latent Topic Significance (LTS) of $w_i$ according to a topic $T_k$:

$$LTS_{w_i}(t_k) = \frac{\sum_{d_j \in D} n(w_i, d_j) P(t_k|d_j)}{\sum_{d_j \in D} n(w_i, d_j)(1 - P(t_k|d_j))}$$  \hspace{1cm} (3.21)

where $n(w_i, d_j)$ represents the number of times the $i^{th}$ word $w_i$ appears in the $j^{th}$ document of the topic model dataset introduced in chapter 2.1.6. It is directly proportional to the significance of the term $w_i$ with respect to the topic $T_k$.

The Gensim implementation\(^\text{12}\) uses a fast implementation of LDA parameter estimation based on online learning [28]. The topic model is trained on $D_{LDA}$ (chapter 3.3.1).

### 3.4.2 Graph construction

A directed graph $G$, with nodes $N = \{U : utterances, S : speakers\}$ and edges $E = \{e_{uu}, e_{us}, e_{ss}\}$, having weights $W = \{w_{uu}, w_{us}, w_{ss}\}$, is constructed. In [14], the utterance similarity between $U_i$ and $U_j$ is either calculated as topical similarity ($\text{TopSim}(U_i, U_j)$), via parameters computed from LDA:

$$\text{TopSim}(U_i, U_j) = \sum_{w \in U_i} \sum_{k=1}^{K} LTS_{w}(T_k) P(T_k|U_i)$$  \hspace{1cm} (3.22)

where $K$ is the number of topics, or as lexical similarity ($\text{LexSim}(U_i, U_j)$) via word overlap, computing the cosine similarity (eq. 2.1) between the frequency vectors of $U_i$ and $U_j$. The authors recommend to use topic similarity, since the word overlap may be sparse due to recognition errors. The results showed better performance of $\text{TopSim}$ when applied to the ASR transcript, while $\text{LexSim}$ with $\text{tfidf}$ is better on the manual transcript. Considering the recent improvements

\(^{12}\text{https://radimrehurek.com/gensim/tut2.html#id7}\)
in the ASR methods, the edge weights are computed as the sum of topical and lexical similarities. The frequency vectors used to compute the lexical similarity in the proposed method are *suid*, since it also considers the importance of different speakers, which is a key feature in the dialogue summarization. Hence, the matrix representing the edge weights in the utterance layer is $L_{uu} = [w_{U_i, U_j}]_{i|u|j|U}$, where $[w_{U_i, U_j}] = TopSim(U_i, U_j) + LexSim(U_i, U_j)$, where $|U|$ is the number of sentences in the summarized segment.

Speaker-speaker and utterance-speaker weights are computed with cosine similarity between the same frequency vectors that were used for computing the lexical similarity in the utterance layer. More in detail, $L_{ss} = [w_{S_i, S_j}]_{i|s|j|S}$, where $w_{S_i, S_j}$ stands for the similarity between the frequency vectors containing all utterances from speakers $S_i$ and $S_j$. Similarly, $L_{us} = [w_{U_i, S_j}]_{i|u|j|S}$ and $L_{su} = [w_{U_j, S_i}]_{i|s|j|S}$, where $w_{U_i, S_j}$ is the cosine similarity between the frequency vectors of utterances and speakers. Row-normalization is performed on all matrices, like suggested in [14]. The proposed random walk integrates the initial and propagated scores as well as the speaker information, to model the speakers' relation automatically. The next section covers the equations describing how utterance importances is propagated in the random walk. Additionally, it shows to which values they converge.

### 3.4.3 Utterance importances initialization

The random walk integrates the initial and propagated scores as well as the speaker information, to model the speakers' relation automatically. The utterance importances $S(s) = 1/|s|$ are uniformly initialized, where $s$ is a sentence of the dialogue. Then, each initial utterance importance is adapted according to the amount of named entities contained (eq. 3.23):

$$S(s) = S(s) \cdot K_{NER}^{(N_{NER})}, \forall s \in U$$

(3.23)

where $K_{NER}$ is the named entity coefficient, $N_{NER}$ is the amount of tagged word in the sentence and $U$ is the set of sentences in the segment. Then, the initial utterance importances are normalized:

$$S(s) = \frac{S(s)}{\sum_{s \in U}}, \forall s \in U$$

(3.24)

Speakers' importances are uniformly initialized.
3.4.4 Utterance importances propagation

Let $F_U^t$ and $F_S^t$ be the importance scores of the utterance and speaker sets - $U$ and $V$ - at the $t_{th}$ iteration. They are computed interpolating the initial importances - $F_U^0 = [S(0),...,S(|s|)]$ and $F_S^0$ (eq. 3.24) - and the score propagated from the other layer, with $\alpha$ propagation weight:

\[ F_U^{t+1} = (1 - \alpha)F_U^{(0)} + \alpha \cdot L_{UU}^T L_{US} F_S^{(t)} \]
\[ F_S^{t+1} = (1 - \alpha)F_S^{(0)} + \alpha \cdot L_{SS}^T L_{SU} F_U^{(t)} \]  \hspace{1cm} (3.25)

The algorithm converges to:

\[ F_U^* = (1 - \alpha)F_U^{(0)} + \alpha \cdot L_{UU}^T L_{US} F_S^* \]
\[ F_S^* = (1 - \alpha)F_S^{(0)} + \alpha \cdot L_{SS}^T L_{SU} F_U^* \]  \hspace{1cm} (3.26)

which can be re-written in form of:

\[ F_U^* = (1 - \alpha)F_U^{(0)} + \alpha \cdot L_{UU}^T L_{US} \left( (1 - \alpha)F_S^{(0)} + \alpha \cdot L_{SS}^T L_{SU} F_U^* \right) \]
\[ = (1 - \alpha)F_U^{(0)} + \alpha (1 - \alpha) L_{UU}^T U_S F_S^{(0)} + \alpha^2 L_{UU}^T L_{US} L_{SS}^T L_{SU} F_U^* \]
\[ = (1 - \alpha)F_U^{(0)} e^T + \alpha (1 - \alpha) L_{UU}^T L_{US} F_S^{(0)} e^T + \alpha^2 L_{UU}^T L_{US} L_{SS}^T L_{SU} \right) F_U^* \]
\[ = M \cdot F_U^* \]  \hspace{1cm} (3.27)

where $e = [1, 1, ..., 1]^T$, with $|e| = |U|$. The closed-form solution of $F_U^*$ is the dominant eigenvector of $M$, and it represents the utterance importance scores. The top ranked sentences are extracted to generate the dialogue summary, until the desired length is reached. The first utterance with which the summary segment partially exceeds the maximum number of words is included.

3.4.5 Redundancy reduction

A basic redundancy reduction is performed in each segment. If a sentence with the same non-stop, lemmatized words has already been included in that segment summary, it will not be added again. Image 3.12 shows an example of a summarized dialogue from TS3005a, where pink sentences mark the correctly extracted sentences, while the missing sentences are marked in blue.
Figure 3.12: TS300Sa transcript with correctly extracted sentences in pink, important missing sentences in blue

DEALogue
my function is user interface design,
I'm I'm the Industrial Designer
and I hope to look forward to a very pleasing end of this project.
I'm a Marketing expert.
My job is in the company to promote company or promote products to the customers.
So I also hope we have a pleasant working with with each other.
Well we have some expertise from different places of the of the company.
Well I said we're working on a project
the aim for the project is to to create a to design a new remote control which has to be original
user friendly.
And I hope we have the expertise to create such a project such a product.
The way we hope to achieve that is the following methods.
It consists of three phases.
namely the functional design.
Chapter 4

Experimental Results

The summarization results depend on the capability of identifying all and only the important sentences. This project aims to extend the current unsupervised extractive state of the art and evaluate the impact of:

- using a weighted sum of document level \( idf \) and segment level scores (eq. 3.14) instead of a basic sum, when computing the keyword weight (eq. 3.15)

- considering named entities in the keyword extraction (eq. 3.18) and in the utterance importances initialization (eq. 3.23)

- considering the most likely words for the most probable topic in both the keyword extraction (chapter 3.3.1) and the initialization of the utterance importance in the random walk (chapter 3.4)

- considering a weighted sum of topical and lexical similarities for computing the utterance layer edge weights (chapter 3.4.2)

- different frequency measures for computing lexical similarity (chapter 3.4.2)

- training the topic model on an extended corpus (chapter 4.1)

4.1 Project datasets and metrics

The AMI Meeting Corpus [11] is a multi-modal dataset consisting of 100 hours of meeting recordings, collected from 138 meetings. It was
created to produce a browser for summarizing meeting transcripts, but is also designed to be useful for a wide range of research areas. It has been largely used for evaluating algorithms analyzing meeting transcripts, including summarization methods, since it is provided with gold-standard human extractive and abstractive summaries. Annotations also provide a link between the most informative utterances used for abstracting and the corresponding sentence in the summary.

4.1.1 Corpora

For evaluating the proposed method, the same meetings used in [7] constitute the test dataset - specifically, the ones from the series ES2004, ES2014, IS1009, TS3003 and TS3009, where each series is formed by four meetings, with an average of 531 sentences (standard deviation: 228 sentences) and 560 (119) distinct words per meeting. In our experiments, due to the improvements in speech and speaker recognition over the last decade, the text source will be the manually annotated transcript instead of the automatic transcription output from the ASR. The AMI Corpus has some specific features, like relatively small set of documents and artificial meetings always discussing the same problem, which have suggested to test topic models trained on different corpora. Overall, it is composed of 138 meetings, with an average of 494 (228) sentences and 543 (146) distinct words per meeting. The complete set is used for computing the frequency measures and training the topic model. A news collection from the BBC news website, corresponding to stories in five topical areas\(^1\) from 2004-2005, is publicly available [25], and is used together with the AMI for training the topic model. It consists of 2,225 documents, with an average of 225 non-stopwords per document. The document order in the training phase does not affect the generated model. Various unlabeled texts are available on GitHub\(^2\) and will be referred to as basic dataset or basic corpus. Finally, the basic corpus for training the topic model is composed of 2,507 documents, with an average of 8 (3) non-stopwords per document. Sentence tokenization is not used for computing the topic model.

\(^1\)Business, entertainment, politics, sport and technology

4.1.2 Metrics

The most used method for analytic summarization evaluation is ROUGE [33], which evaluates the algorithm performance by computing the recall between the generated summary and the reference one based on overlapping words and sequences. Usually the metric applied is ROUGE – N, which is an n-gram recall and is computed as:

\[
ROUGE - N = \frac{\sum_{S \in \text{RefSumm}} \sum_{\text{gram}_n} \text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{S \in \text{RefSumm}} \sum_{\text{gram}_n} \text{Count}(\text{gram}_n)}
\]

(4.1)

where \( S \) is the set of sentences belonging to the reference summary RefSumm, \( \text{Count}_{\text{match}}(\text{gram}_n) \) is the maximum number of n-grams occurring in both texts and \( \text{gram}_n \) is an n-gram, i.e. a contiguous sequence of n words. Another commonly used ROUGE is ROUGE – SU4 which measures the overlapping of skip-bigrams\(^3\). ROUGE-L stays for the longest common subsequence-based statistics.

Therefore, this evaluation score depends on the human summary used as a reference, and it has been shown that different humans may produce summaries really different from each other. For this reason, other methods for evaluating summaries like the Pyramid method [41] have been proposed. However, they require many human summarizers to get different gold-standard human targets and, in general, a considerable effort. Due to these problems, together with the lack of publications evaluated with other metrics, the proposed method is evaluated with the most common ROUGE metrics. In conclusion, an easy, broad metric for performance evaluation is still an open challenge in the summarization domain. ROUGE, expressed in terms of Precision (P), Recall (R) and F-score (F) (see chapter 2.1.8 for a brief introduction of those concepts), is computed at the utterance level. In this project, the evaluation metrics used are ROUGE-1, ROUGE-2 and ROUGE-L.

4.2 Experiments

Several parameters can be tuned in the proposed method, as well as different measures that lead to different results. Here, all the constant parameters are summarized. In the test cases, different values of the

\(^3\)All sets of two words with less than four words between each other
following parameters have been tested, and only the optimal values are reported in table 4.1, together with the reference section or equation.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>$L_{\text{win}}$</th>
<th>$s$</th>
<th>$k_{\text{peak}}$</th>
<th>$T_m$</th>
<th>$N_T$</th>
<th>$\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Values</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
<tr>
<td>References</td>
<td>3.7 eq. 3.8</td>
<td>eq. 3.10</td>
<td>3.3.1</td>
<td>3.4.1</td>
<td>eq. 3.4.4</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.1: Constant parameters

The following test cases have been compared to investigate the impact of the features introduced in this work. All of the cases are associated to a reference ID, to avoid large text strings in table 4.3, which summarizes all the relevant results. If not explicitly stated, the tests used a compression ratio of 50%, which is the maximum compression allowed to lie into the definition of summary [47]. In each subsection, the features used in the test case are presented, together with the analysis of the results. In table 4.3, the results presented are the average and the standard deviation (in parenthesis) of the F-measure of ROUGE-1, ROUGE-2 and ROUGE-L metrics (F-1, F-2 and F-L), computed on the summaries of the test dataset. In table 4.2, the variable parameters are reported. Multiple tests have shown that, even though the algorithms use probabilistic concepts, results are consistent. Therefore, a single test for each parameter setting is conducted, and the standard deviation just represents the variation of the results on the test set.

4.2.1 Baseline

The baseline is a simplified version of the proposed method, similar to the current state of the art, on top of which the new features are evaluated. It uses only the lexical similarity - tfidf - to measure the utterance similarity for weighting edges in the random walk. The topic model is trained only on the AMI Meeting Corpus. Frequency and entropy have the same importance in keyword extraction.

4.2.2 Test case 1

The utterance similarity is based on topical similarity instead of lexical similarity, to show that different similarity measures have an impact on the importance propagation through the graph. The topic model is
trained on the AMI Corpus, showing a clear improvement in all of the metrics.

### 4.2.3 Test case 2

Now, both topical and lexical similarity are used for computing the utterance similarity. Two different graphs are used, and the converged utterance importances are summed to determine which sentences will be included in the final summary. The results show a little degradation with respect to test case 1, but are still way better than test case 0.

### 4.2.4 Test case 3

In this test case, the keyword extraction is analyzed. Here, the frequency measure impacts more on the words’ scores (eq. 3.15). Parameters related to functional segmentation are set to default values. An increase in Recall and F-measure is revealed, meaning that the word frequency over the corpus is more relevant than the word entropy, which reflects the rarity of the word in the meeting. However, in the paper where the keyword extraction in the meeting domain was introduced [8], the authors demonstrated a clear improvement by introducing the entropy. In conclusion, a good balance is obtained by assigning more importance to the frequency measure.

### 4.2.5 Test case 4

Now, keywords are extracted by assigning more importance to the entropy measure rather than to the frequency. Recall measures drop down with a small improvement on the Precision, leading to worse F-measures as well. Therefore, this test case confirms the conclusion drawn in test case 3.

### 4.2.6 Test case 5

Up to now, the topic model used for computing the topical similarity between utterances was trained on the AMI Corpus. However, its lim-
ited size is a known problem. To overcome it, the topic model is now trained on the AMI Corpus together with the BBC news collection (AB in table 4.2). The average results confirm the hypothesis that a more extended dataset leads to a better topic model, which helps in computing a more accurate topical similarity and a more accurate summary. All metrics are improved.

4.2.7 Test case 6

Is an even bigger corpus necessarily better for training a topic model? To answer this question, the basic corpus is used together with the two previous ones to train our model (ABC in table 4.2). Experiments enhance small variations compared to the previous test case, and generally do not justify the increased computational time, affecting both training and inference phases.

4.2.8 Test case 7

The $suidf$ frequency measure has been introduced in the functional segmentation to integrate $idf$, term frequency and speaker information, a specific feature of meetings. Here, the $suidf$ frequency measure is also used to compute the utterance lexical similarity used for summarizing dialogues, i.e. sections where more speakers are involved. The effectiveness of also considering speaker information is confirmed by the experiments, with a growth in terms of Recall and F-measure.

4.2.9 Test case 8

Until now, two different graphs have been used to propagate utterance importances using weights proportional to topical and lexical similarity, respectively. In this case, the edge weights are summed before the random walk, in order to evaluate which solution leads to better results according to the ROUGE metric. The results are almost not affected by this change, meaning that the sum of the similarity measures leads to the same converged utterance importances obtained by summing the importance obtained with random walks through two
different graphs. Computationally, this solution is more efficient.

4.2.10 Test case 9

Evaluates the effect of using topic keywords only for the meeting keywords extraction, while utterance importances are initialized uniformly. It does not show any improvement, when the model is trained on AMI Corpus and BBC news.

4.2.11 Test case 10

Now topic keywords are also used for adjusting the initial utterance importances in the random walk. Importances are initially uniform, then, in each sentence, the importances are multiplied by the topic keyword score, for each topic keyword in the sentence. Again, no significant effect is generated.

4.2.12 Test case 11

Another way of considering topic keywords for initializing utterance importances is to multiply the initial importances by the accumulated keywords' weights for each sentence. See (eq. C.2) for a mathematical representation. This solution shows a small improvement in Recall and F-measure, achieving the best results together with Test case 8.

4.2.13 Test case 12

Accumulated weights depending on topic keywords to initialize utterance importances is also tested when the topic model is trained on AMI, BBC and Basic Corpora. The recall metric is lower than in Test case 12, confirming that overextending the topic model training set does not guarantee better performance.
4.2.14 Test case 13

The last tests aim to determine the impact of considering named entities to extract meeting keywords. Each word weight is multiplied by a factor depending on its label. This configuration outperforms all the other cases in terms of Recall on ROUGE-2, which considers bigram overlaps. The other results are similar to the best ones just obtained.

4.2.15 Test case 14

Like the topic keywords, named entities can also be used to initialize the utterance importances that will then be propagated in the multilayer graph. This approach outperforms all the proposed approaches in terms of ROUGE-2 and ROUGE-L, even if it does not improve the results in terms of ROUGE-1. Therefore, these are the features that lead to the best result in terms of ROUGE metrics.

4.2.16 Test case 15

All the previous cases have been performed with a compression rate of 50% of the non stop words. Now, the optimal parameters determined in the previous test cases are used to perform experiments at 30% of compression ratio. As expected, the fraction of important sentences extracted compared to the total number of important sentences from the reference summary (Recall) decreases, while the fraction of relevant sentences compared to the number of irrelevant sentences (Precision) increases. The F-measure decreases of a few percent points, but it outperforms the current unsupervised extractive state of the art in terms of ROUGE-1 metric.
### Table 4.2: Experiment parameters

<table>
<thead>
<tr>
<th>#</th>
<th>$T_t$</th>
<th>$i_T$</th>
<th>$k_T$</th>
<th>$k_{MF}$</th>
<th>$k_{ME}$</th>
<th>$K_{NER}$</th>
<th>freq_{t Tex}</th>
<th>$D_{LDA}$</th>
<th>$k_{top}$</th>
<th>$k_{lex}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>1</td>
<td>N/A</td>
<td>tfidf</td>
<td>N/A</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>1</td>
<td>N/A</td>
<td>N/A</td>
<td>AMI</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>1</td>
<td>N/A</td>
<td>tfidf</td>
<td>AMI</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>tfidf</td>
<td>AMI</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>4</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>2</td>
<td>N/A</td>
<td>tfidf</td>
<td>AMI</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>5</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>tfidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>6</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>tfidf</td>
<td>ABC</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>7</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>suidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>8</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>suidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>9</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>suidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>10</td>
<td>0.2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>suidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>11</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>suidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>12</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>N/A</td>
<td>suidf</td>
<td>ABC</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>13</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>suidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>14</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>suidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>15</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td>suidf</td>
<td>AB</td>
<td>0.5</td>
<td>0.5</td>
</tr>
</tbody>
</table>

### Table 4.3: Results of the test cases

<table>
<thead>
<tr>
<th>#</th>
<th>F-1</th>
<th>F-2</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.684 (0.102)</td>
<td>0.478 (0.077)</td>
<td>0.632 (0.095)</td>
</tr>
<tr>
<td>1</td>
<td>0.700 (0.063)</td>
<td>0.504 (0.079)</td>
<td>0.654 (0.090)</td>
</tr>
<tr>
<td>2</td>
<td>0.698 (0.064)</td>
<td>0.500 (0.078)</td>
<td>0.651 (0.092)</td>
</tr>
<tr>
<td>3</td>
<td>0.702 (0.063)</td>
<td>0.504 (0.076)</td>
<td>0.652 (0.090)</td>
</tr>
<tr>
<td>4</td>
<td>0.701 (0.062)</td>
<td>0.501 (0.077)</td>
<td>0.656 (0.088)</td>
</tr>
<tr>
<td>5</td>
<td>0.704 (0.064)</td>
<td>0.505 (0.078)</td>
<td>0.652 (0.094)</td>
</tr>
<tr>
<td>6</td>
<td>0.704 (0.066)</td>
<td>0.507 (0.082)</td>
<td>0.653 (0.095)</td>
</tr>
<tr>
<td>7</td>
<td>0.705 (0.065)</td>
<td>0.507 (0.078)</td>
<td>0.653 (0.095)</td>
</tr>
<tr>
<td>8</td>
<td>0.705 (0.065)</td>
<td>0.507 (0.078)</td>
<td>0.653 (0.094)</td>
</tr>
<tr>
<td>9</td>
<td>0.705 (0.064)</td>
<td>0.507 (0.077)</td>
<td>0.654 (0.093)</td>
</tr>
<tr>
<td>10</td>
<td>0.703 (0.066)</td>
<td>0.506 (0.079)</td>
<td>0.652 (0.095)</td>
</tr>
<tr>
<td>11</td>
<td>0.705 (0.065)</td>
<td>0.507 (0.078)</td>
<td>0.653 (0.095)</td>
</tr>
<tr>
<td>12</td>
<td>0.704 (0.067)</td>
<td>0.507 (0.083)</td>
<td>0.653 (0.096)</td>
</tr>
<tr>
<td>13</td>
<td>0.705 (0.064)</td>
<td>0.507 (0.075)</td>
<td>0.653 (0.094)</td>
</tr>
<tr>
<td>14</td>
<td>0.705 (0.064)</td>
<td>0.509 (0.075)</td>
<td>0.655 (0.094)</td>
</tr>
<tr>
<td>15</td>
<td>0.665 (0.047)</td>
<td>0.451 (0.053)</td>
<td>0.629 (0.054)</td>
</tr>
</tbody>
</table>
## 4.3 Analysis of the results

The test cases just introduced in the previous chapter show that methods including all the introduced features outperform the baseline. Table 4.3 reveals that the standard deviation is constant in all the test cases, therefore even small variations in the evaluation metric represent a relevant impact of the introduced features. These results will then be most likely confirmed on other test sets.

The topical similarity is better than the lexical similarity to compute the edge weights of the graph used for the random walk. When the standard frequency measure (\textit{tfidf}) is used for computing the lexical similarity, the topical similarity achieves better results even compared to the method that sums the converged utterance importances obtained through two different random walks (\textit{Test cases 1-2}).

In \textit{Test cases 3-4}, it has been shown that the frequency of words is more important than their position in the meeting for the keyword extraction, recalling however that using only frequency information leads to a less accurate summary [8].

Another key result is that the topic model is more effective when it is trained on an extended Corpus, specifically including the AMI Corpus and the BBC news collection. However, when the basic Gensim Corpus was included, the performance degraded. This demonstrates that these kinds of corpora, composed of single-sentence documents, are not always generating a better model.

Let us back now to the random walk graph and specifically to its construction. The new topic model has improved the summary accuracy, and the introduction of \textit{suidf} improves it even more. Since this method is used to summarize multi-speaker conversations, the \textit{suidf} metric outperforms the \textit{tfidf} because it also considers the speaker information.

Furthermore, there is basically no difference between summing the converged utterance importances propagated in two different graphs and summing the edge weights before the random walk. The last approach is computationally more efficient.
The evaluation of the impact of topic keywords has shown a negligible effect. The best way is to use them for both initializing utterance importances and for extracting and assigning weights to keywords. Again, performance degrades when the topic model is trained also on the basic Corpus.

Like topic keywords, named entities have been investigated for both extracting keywords and initializing utterance importances. When applied only for keyword extraction, it leads to a better Recall on ROUGE-2. The most extended method includes entity tags also for initializing utterance importance, and achieves best performance on almost all metrics, outperforming the other settings in terms of ROUGE-2 and ROUGE-L.

Tests have been performed with different segmentation parameters, and optimal results are achieved with the configuration of Test case 14. In test case 14, word frequency has more importance than its position in the meeting to determine keyword weights, the topic model has to be trained on the AMI Corpus and BBC news together, $suidf$ is the metric used to compute lexical similarity for the graph construction while topic keywords have basically no impact on the results. The optimal parameters are summarized in table 4.4.

<table>
<thead>
<tr>
<th>$L_{win}$</th>
<th>$s$</th>
<th>$k_{peak}$</th>
<th>$T_I$</th>
<th>$i_T$</th>
<th>$k_T$</th>
<th>$k_{MF}$</th>
<th>$k_{ME}$</th>
<th>$T_m%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>$K_{NER}$</td>
<td>freqex</td>
<td>$D_{LDA}$</td>
<td>$k_{top}$</td>
<td>$k_{lex}$</td>
<td>$L_m%$</td>
<td>$L_d%$</td>
<td>$N_T$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>3</td>
<td>suidf</td>
<td>AMI+BBC</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Table 4.4: Optimal parameters

Figures 4.1 and 4.2 show two transcript segments, where correctly extracted sentences are marked in pink and missing sentences are marked in blue. The full meeting with corresponding extracted summary is attached in appendix D. There, the reader can observe that several important sentences are still not detected, demonstrating that this method is not effective for a reliable commercial solution. A larger and more complete dataset, together with supervised methods would be needed for this purpose. Tests performed with a compression rate of 30% show
that Precision and F-measure of the proposed method outperforms the current state of the art according to the ROUGE-1 metric.

Show next segment
It 's also to gets to know each other because I 'm asking three things
to do it on a blank sheet
with different colours
and I just showed you how to pick a colour
and also with different pen widths which I also showed you .
A favourite characteristic can be just one word.
I 'm not very good at drawing
but I will go first
and try to draw
or maybe you should guess what I 'm drawing
Dinosaur .
could be everything .
Maybe when I put on
it could be a turtle
well the small does n't have legs .
and I hope our project group will not be slow
but we will work to a good result
time for another animal .
would you like to go next ?
It was four months /
right .
To make it a little bit easier .
make that cute .
recognise as a giraffe .
the favourite chair characteristic is that the long neck
it can reach everything .
and I hope I can also reach a lot with this project .
so that 's my favourite animal .
Could you write the words ...

Anything else you need to know ?
Bunny rabbit .
A bunny rabbit .
Wrong one .
well you can guess what it is
and well it 's quick
Little rabbits
that 's my favourite animal .
and our final drawing .
Bob Ross .
I 've drawn a dolphin because of its intelligence .
one of the most intelligent
animals in our world .
You can try out the eraser now .
will not perfect
well thank you very much .
I can see we have some drawing talent in this group
nice animals
nice words .
sounds good .
back to business
back to the money part .
From the finance department I have learned that we are aiming for a selling price of twenty five euros .
and we 're hoping for a sale of fifty million euros
we are hoping to achieve that by aiming for an international market .
and the production cost will be twelve Euro fifty max .
well it 's time for some discussion .

Figure 4.1: TS3005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue

Figure 4.2: TS3005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue
Chapter 5

Conclusions

This project aimed to present a fully unsupervised extractive framework evaluating the impact of several features.

5.1 Basic concepts and literature

This report first introduces basic concepts useful in natural language processing like frequency and distance measures, NLP libraries, pre-processing and word tagging. Then, this report presents an overview of the scientific literature about Extractive and Abstractive Meeting Summarization. Here, many cluster and graph based approaches for both extractive and abstractive summarization are described, analyzing their pros and cons.

5.2 Proposed method and results

The presented method extends the current unsupervised extractive state of the art [7], introducing and evaluating the impact of new features like a new dataset for training topic models, the usage of a new library - SpaCy, different parameter settings, the concept of topic keywords and the application of named entities for adapting importance weights as well as the idea of combining results from two random walks on different graphs. Even though the variations in the results are quite small, consistent standard deviations indicate that the features impact will most likely be confirmed on other test sets. The presented method, which uses a smaller graph to perform the random walk com-
pared to the unsupervised extractive state of the art, demonstrates the effectiveness of those new features and outperforms the state of the art in terms of ROUGE-1 Precision and F-measure.

5.3 Contribution and future works

Hopefully, this work will help future researches on this field. First, a complete and extendable source code will be soon available for non-commercial use, such that there will be no more need to implement existing solutions from scratch. Second, different solutions have already been explored and tested, such that future works can either focus on other features and extend the proposed ones. Some of the introduced features will quite likely improve the effectiveness of supervised approaches. Finally, abstractive methods usually depends on the extracted sentences, and the sentences extracted with this method can be used as baseline to compare and analyze several abstractive approaches.

5.4 Applicability to commercial use

In terms of applicability to commercial solutions, currently available methods are not accurate enough, demanding for a much larger, realistic and generalized annotated dataset. This would allow researchers to develop more advanced supervised frameworks for general and focused\(^1\) summarization techniques. In appendix D the reader finds a full meeting transcription with relative automatic and reference summaries.

5.5 Social and ethical impact

As just stated, this work does not produce a feasible commercial solution. Therefore, it will not have a direct social nor ethical impact. However, it is a step forward towards a potential large-scale application.

\(^{1}\)Algorithms developed for detecting specific sentences, for example the ones where decisions are taken or where a specific topic is discussed
The social impact of a large-scale implementation of a tool capable to produce meeting minutes can be seen under two points of view. On one side, companies will save money, get immediate feedback and collect structured data which could then be further analyzed. Moreover, secretaries or other employees will not be required to perform such a boring and repetitive task. On the other side, it will most likely reduce the amount of job offers. Due to this risk, companies should think about creating training paths to teach their secretaries new knowledge, in order to guarantee an employment to all the people replaced by this Artificial Intelligence.

Finally, this software will be, in most of the cases, a cloud-base solution, such that the new data collected can be used for producing more accurate summaries. This will require additional security, to avoid an easy access to sensitive data by the hackers.
Chapter 6

Acknowledgments

Many people have assisted me in this project, and I will for sure forget to thank someone, to whom I apologize in advance.

Thanks to Seavus for the opportunity to work on this challenging project, for the continuous follow up with the Project Manager and Head of R&D, Mr. Reijo Silander, as well as for the technical and moral support all the other employees have secured me.

I would love to mark the extremely helpful assistance received by the Supervisor, Mr. Johan Boye, associate professor in the Speech Technology group at the School of Electrical Engineering and Computer Science at KTH. Correct advice and no urgency, together with a warm work environment, have been extremely important for completing and enjoying this project.

Thanks to all the researchers that have answered my emails and clarified my doubts about previous works. Thanks to all the people releasing open source code and sharing their results, boosting the technology evolution.

Last but most important, a huge thanksgiving to my family for the economical and moral support, as to my old and new friends.
Bibliography


Appendix A

SpaCy NER Types and Descriptions

The built-in entity types in the models trained on the OntoNotes 5\(^1\) corpus support the following entity types:

<table>
<thead>
<tr>
<th>TYPE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERSON</td>
<td>People, including fictional.</td>
</tr>
<tr>
<td>NER</td>
<td>Nationalities or religious or political groups.</td>
</tr>
<tr>
<td>FAC</td>
<td>Buildings, airports, highways, bridges, etc.</td>
</tr>
<tr>
<td>ORG</td>
<td>Companies, agencies, institutions, etc.</td>
</tr>
<tr>
<td>GPE</td>
<td>Countries, cities, states.</td>
</tr>
<tr>
<td>LOC</td>
<td>Non-GPE locators, mountain ranges, bodies of water</td>
</tr>
<tr>
<td>EVENT</td>
<td>Named hurricanes, battles, wars, sports events, etc.</td>
</tr>
<tr>
<td>WORK_OF_ART</td>
<td>Titles of books, songs, etc.</td>
</tr>
<tr>
<td>LAW</td>
<td>Named documents made into law.</td>
</tr>
<tr>
<td>LANGUAGE</td>
<td>Any named language.</td>
</tr>
<tr>
<td>DATE</td>
<td>Absolute or relative dates or periods.</td>
</tr>
<tr>
<td>TIME</td>
<td>Times shorter than a day.</td>
</tr>
<tr>
<td>PERCENT</td>
<td>Percentage, including ‘%’.</td>
</tr>
<tr>
<td>MONEY</td>
<td>Monetary values, including unit.</td>
</tr>
<tr>
<td>QUANTITY</td>
<td>Measurements, as of weight or distance.</td>
</tr>
<tr>
<td>ORGANIZATIONAL</td>
<td>“First”, “second”, etc.</td>
</tr>
<tr>
<td>CARDINAL</td>
<td>Numerals that do not fall under another type.</td>
</tr>
</tbody>
</table>

Figure A.1: List of supported entity types and descriptions in SpaCy

\(^{1}\)https://catalog.ldc.upenn.edu/ldc2013t19
Appendix B

Human Gold-Standard Functional Segmentation

In the next pages, the reader finds an example of how a human segments a meeting transcript, dividing monologues and dialogues and determining the active speakers. More in detail, figures B.1 and B.2 shows a section of meeting E52008a segmented in one monologue and two dialogues. Speaker A is marked in red, speaker B in green, speaker C in blue, speaker D in black and non-active speakers in each segment are marked in grey.
[...] 
B, NON-ACTIVE SPEAKERS 
B: yeah um . 
B: i agree with having too many remotes around 
B: my dad has a whole drawer at home of remotes for various 
things 
B: and i do n't know how to work half of them um . 
B: what 's important for me , i guess , is that it 's easy to 
use 
B: and that there 's not too many buttons 
B: they are not too small 
B: you know you know you need to n to know what you 're 
doing 
B: and one thing i particularly like is if you are not um sort 
of moving it around to get it to work with the infra-red . 
B: yeah 
B: um , i think there is a way around that , 
C: yeah 
B: but i know in my residence right now the the television you 
sort of have to walk all around the room to get it to turn 
on 
A: -lcb- vocalsound -rcb- um-hmm 
B: so i it 's just simpler just to just turn around the tv 
itself , 
B: and i think that 's -lcb- disfmarker -rcb- if we 're gon na 
mak emake a remote control , it should actually work for what it 's 
doing 
B: so -lcb- disfmarker -rcb- 
A, D, NON-ACTIVE SPEAKERS 
D: what about like batteries and things like that , 
D: like are there some remotes that don do n't require like 
batteries 
D: or do all remotes require batteries ? 
A: -lcb- vocalsound -rcb- um i would imagine all of them , 
B: i know 
A: but we could -lcb- disfmarker -rcb- 
A: but it 's possible we could use like a lithium battery 
A: um that would last a lot longer than like double as . 
B: yeah , 
B: something that does n't -lcb- disfmarker -rcb- 
D: mm-hmm 
A: um like tho those are the batteries that are used in a lot 
of um mp three players now and that kind of thing , um . 
D: mm-hmm 
A: mm . 
A, B, D, NON-ACTIVE SPEAKERS 
A: um . okay , 
A: it seems we have a little bit of a conflict over um to uh 
combining all the remotes cont together versus having f five 
different remotes 
A: so um 
A: like you said you do n't like having all the buttons on one 
on one remote , 
A: and yet you do n't wan na have five remotes . 
A: so how do we work with that ? 

Figure B.1: Gold-standard human segmentation: section of meeting ES2008a, part 1
B: -1cb- vocalsound -rcb- yeah.
A: mm.
B: could we get something that
B: just has -1cb- dismarker -rcb-
B: no
B: does n't have all the buttons that you need to program the
video recorder
B: or program s other things that i 'm not very coherent
about
B: but that just has your major buttons for -1cb- dismarker -
rcb- that work for everything, you know volume control, on, 
off,
A: mm-mm.
B: channel changing.
D: and maybe that spatially divides it.
D: so it 's like if you 're looki if you 're trying to get the
tv on that 's, you know, like the top thing on the remote,
i dunno if d be vertical or horizontal in terms of how we 're
gon na make it, but if it 's like all the tv stuff was here,
B: yeah.
D: then all the vcr stuff was here.
D: all the -1cb- dismarker -rcb- whatever else we have
programmed into it it 's all just in its separate place and
not like all the on buttons together.
A: mm.
B: that way -1cb- dismarker -rcb-
B: yeah.
D: 'cause then you like, i do n't even know what i 'm turning
on.
A: mm.
B: -1cb- vocalsound -rcb- yeah.
B: and um if you 'd save the more complicated functions
maybe for separate remotes that you would n't need to use
every day.
D: -1cb- vocalsound -rcb-
D: mm-mm.
A: okay,
A: so maybe have like one remote that has the main functions
on, off, channel changing, volume, and another rote remote
with all the special things.
B: um.
A: because that is one thing that um remotes tend to have
buttons that the tvs no longer have as well.
B: yeah.
A: so like you have to have them somewhere,
B: mm.
C: yeah.
A: 'cause you 're gon na m need those special functions
occasionally.
A: um but not necessarily on the m the normal remote.
B: right

[...]

Figure B.2: Gold-standard human segmentation: section of meeting ES2008a, part 2
Appendix C

Test Cases

All test cases and corresponding parameters and results are presented here. The meetings used for evaluating the features introduced are the ones from the series ES2004, ES2014, IS1009, TS3003 and TS3009 of the AMI Meeting Corpus, where each series is formed by four meetings. The evaluation metrics computed are P-1, R-1, F-1 (Precision, Recall and F-measure of the ROUGE-1), P-2, R-2, F-2 (Precision, Recall and F-measure of the ROUGE-2) and P-L, R-L, F-L (Precision, Recall and F-measure of the ROUGE-L).

C.1 Baseline

The baseline is a simplified version of the proposed method, similar to the current state of the art, on top of which the new features are evaluated. It uses only the lexical similarity to measure the utterance similarity for weighting edges in the random walk. The topic model is trained only on the AMI Meeting Corpus. Frequency and entropy have the same importance in keyword extraction. The parameters used are presented in Table C.1:

<table>
<thead>
<tr>
<th>$L_{win}$</th>
<th>$s$</th>
<th>$T_{i}$</th>
<th>$i_{T}$</th>
<th>$k_{T}$</th>
<th>$k_{MF}$</th>
<th>$k_{ME}$</th>
<th>$T_{m}$</th>
<th>$\alpha$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>1</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>$K_{NER}$</td>
<td>freqlex</td>
<td>$D_{LDA}$</td>
<td>$k_{top}$</td>
<td>$k_{lex}$</td>
<td>$L_{m}$%</td>
<td>$L_{d}$%</td>
<td>$N_{T}$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>N/A</td>
<td>tfidf</td>
<td>N/A</td>
<td>0</td>
<td>1</td>
<td>0.5</td>
<td>0.5</td>
<td>N/A</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Table C.1: Parameters of the baseline
This leads to the following results, presented in table C.2:

They are now used as comparison to evaluate the impact of considering different features.

## C.2 Test case 1 - Topical similarity

Here, the utterance similarity is based on topical similarity instead of lexical similarity, to show that different similarity measures have an impact on the importance propagation through the graph. The topic model is trained only on the AMI Corpus. The parameters used are presented in table C.3:

<table>
<thead>
<tr>
<th>( L_{\text{win}} )</th>
<th>s</th>
<th>( k_{\text{peak}} )</th>
<th>( T_I )</th>
<th>( i_T )</th>
<th>( k_T )</th>
<th>( k_{M^I} )</th>
<th>( k_{M^E} )</th>
<th>( T_m % )</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>( K_{\text{NER}} )</td>
<td>( \text{freq}_{\text{ex}} )</td>
<td>( D_{\text{LDA}} )</td>
<td>( k_{\text{top}} )</td>
<td>( k_{\text{ex}} )</td>
<td>( L_m % )</td>
<td>( L_q % )</td>
<td>( N_T )</td>
<td>( \alpha )</td>
</tr>
<tr>
<td>N/A</td>
<td>N/A</td>
<td>AMI</td>
<td>1</td>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
</tbody>
</table>

This leads to the following results, presented in table C.4:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.627</td>
<td>0.809</td>
<td>0.700</td>
<td>0.439</td>
<td>0.624</td>
<td>0.504</td>
<td>0.612</td>
<td>0.790</td>
<td>0.654</td>
</tr>
<tr>
<td>Dev</td>
<td>0.106</td>
<td>0.036</td>
<td>0.063</td>
<td>0.115</td>
<td>0.042</td>
<td>0.079</td>
<td>0.104</td>
<td>0.039</td>
<td>0.090</td>
</tr>
</tbody>
</table>

This test shows a clear improvement in all of the metrics.
C.3 Test case 2 - Topical and lexical similarity

Now, both topical and lexical similarity are used for computing the utterance similarity. Two different graphs are used, and the converged utterance importances are summed to determine which sentences will be included in the final summary. The parameters used are presented in table C.5:

<table>
<thead>
<tr>
<th>L&lt;sub&gt;win&lt;/sub&gt;</th>
<th>s</th>
<th>k&lt;sub&gt;peak&lt;/sub&gt;</th>
<th>T&lt;sub&gt;t&lt;/sub&gt;</th>
<th>i&lt;sub&gt;T&lt;/sub&gt;</th>
<th>k&lt;sub&gt;T&lt;/sub&gt;</th>
<th>k&lt;sub&gt;MF&lt;/sub&gt;</th>
<th>k&lt;sub&gt;ME&lt;/sub&gt;</th>
<th>T&lt;sub&gt;m&lt;/sub&gt; %</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>K&lt;sub&gt;NER&lt;/sub&gt;</td>
<td>freq&lt;sub&gt;lex&lt;/sub&gt;</td>
<td>D&lt;sub&gt;LDA&lt;/sub&gt;</td>
<td>k&lt;sub&gt;top&lt;/sub&gt;</td>
<td>k&lt;sub&gt;lex&lt;/sub&gt;</td>
<td>L&lt;sub&gt;m&lt;/sub&gt;%</td>
<td>L&lt;sub&gt;d&lt;/sub&gt;%</td>
<td>N&lt;sub&gt;T&lt;/sub&gt;</td>
<td>α</td>
</tr>
<tr>
<td>N/A</td>
<td>tfidf AMI</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
<td></td>
</tr>
</tbody>
</table>

Table C.5: Parameters of Test 2

This leads to the following results, presented in table C.6:

<table>
<thead>
<tr>
<th></th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.624</td>
<td>0.808</td>
<td>0.698</td>
<td>0.433</td>
<td>0.620</td>
<td>0.500</td>
<td>0.608</td>
<td>0.788</td>
<td>0.651</td>
</tr>
<tr>
<td>Dev</td>
<td>0.105</td>
<td>0.038</td>
<td>0.064</td>
<td>0.116</td>
<td>0.040</td>
<td>0.078</td>
<td>0.104</td>
<td>0.042</td>
<td>0.092</td>
</tr>
</tbody>
</table>

Table C.6: Results of test 2

The results show a little degradation with respect to test case 1, but are still better than the baseline.

C.4 Test case 3 - More importance to the frequency measure for keyword extraction

In this test case, the keyword extraction is analyzed. Here, the frequency measure impacts more on the words’ scores. Parameters related to functional segmentation are set to default values. The parameters used are presented in table C.7:

This leads to the following results, presented in table C.8:

An increase in Recall and F-measure is revealed, meaning that the word frequency over the corpus is more relevant that the word entropy, which reflects the rarity of the word in the meeting. However,
<table>
<thead>
<tr>
<th>L\textsubscript{win}</th>
<th>s</th>
<th>k\textsubscript{peak}</th>
<th>T\textsubscript{t}</th>
<th>i\textsubscript{T}</th>
<th>k\textsubscript{T}</th>
<th>k\textsubscript{MF}</th>
<th>k\textsubscript{ME}</th>
<th>T\textsubscript{m}</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>K\textsubscript{NER}</td>
<td>freq\textsubscript{lex}</td>
<td>D\textsubscript{LDA}</td>
<td>k\textsubscript{top}</td>
<td>k\textsubscript{lex}</td>
<td>L\textsubscript{m}</td>
<td>%</td>
<td>L\textsubscript{d}</td>
<td>%</td>
<td>N\textsubscript{T}</td>
</tr>
<tr>
<td>N/A</td>
<td>tfidf</td>
<td>AMI</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
</tr>
</tbody>
</table>

Table C.7: Parameters of Test 3

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.624</td>
<td>0.821</td>
<td>0.702</td>
<td>0.439</td>
<td>0.623</td>
<td>0.504</td>
<td>0.607</td>
<td>0.799</td>
<td>0.652</td>
</tr>
<tr>
<td>Dev</td>
<td>0.103</td>
<td>0.039</td>
<td>0.063</td>
<td>0.112</td>
<td>0.044</td>
<td>0.076</td>
<td>0.101</td>
<td>0.042</td>
<td>0.090</td>
</tr>
</tbody>
</table>

Table C.8: Results of test 3

In the paper where the keyword extraction in the meeting domain was introduced [8], the authors demonstrated a clear improvement by introducing the entropy. In conclusion, a good balance is obtained by assigning more importance to the frequency measure.

### C.5 Test case 4 - More importance to the entropy measure in the keyword extraction

Now, keywords are extracted by assigning more importance to the entropy measure rather than to the frequency. The parameters used are presented in table C.9:

<table>
<thead>
<tr>
<th>L\textsubscript{win}</th>
<th>s</th>
<th>k\textsubscript{peak}</th>
<th>T\textsubscript{t}</th>
<th>i\textsubscript{T}</th>
<th>k\textsubscript{T}</th>
<th>k\textsubscript{MF}</th>
<th>k\textsubscript{ME}</th>
<th>T\textsubscript{m}</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>1</td>
<td>2</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>K\textsubscript{NER}</td>
<td>freq\textsubscript{lex}</td>
<td>D\textsubscript{LDA}</td>
<td>k\textsubscript{top}</td>
<td>k\textsubscript{lex}</td>
<td>L\textsubscript{m}</td>
<td>%</td>
<td>L\textsubscript{d}</td>
<td>%</td>
<td>N\textsubscript{T}</td>
</tr>
<tr>
<td>N/A</td>
<td>tfidf</td>
<td>AMI</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Table C.9: Parameters of Test 4

This leads to the following results, presented in table C.10:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.626</td>
<td>0.809</td>
<td>0.701</td>
<td>0.434</td>
<td>0.623</td>
<td>0.501</td>
<td>0.613</td>
<td>0.790</td>
<td>0.656</td>
</tr>
<tr>
<td>Dev</td>
<td>0.104</td>
<td>0.037</td>
<td>0.062</td>
<td>0.112</td>
<td>0.044</td>
<td>0.077</td>
<td>0.102</td>
<td>0.039</td>
<td>0.088</td>
</tr>
</tbody>
</table>

Table C.10: Results of test 4
Recall measures drop down with a small improvement on the Precision, leading to worse F-measures as well. Therefore, this test case confirms the conclusion drawn in test case 3.

### C.6 Test case 5 - Introduce the BBC news corpus for training the topic model

Up to now, the topic model used for computing the topical similarity between utterances was trained on the AMI Corpus. However, its limited size is a known problem. To overcome it, the topic model is now trained on the AMI Corpus together with the BBC news collection. The parameters used are presented in table C.11:

<table>
<thead>
<tr>
<th>$L_{win}$</th>
<th>s</th>
<th>$k_{peak}$</th>
<th>$T_{i}$</th>
<th>$i_{T}$</th>
<th>$k_{TF}$</th>
<th>$k_{MF}$</th>
<th>$k_{ME}$</th>
<th>$T_{m}/%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
<td></td>
</tr>
</tbody>
</table>

| $K_{NER}$ | freqex | $D_{LDA}$ | $k_{top}$ | $k_{ex}$ | $L_{m}/\%$ | $L_{d}/\%$ | $N_{T}$ | $\alpha$ |
|-----------|--------|-----------|-----------|----------|-------------|-------------|---------|
| N/A       | tfidf  | AMI+BBC   | 0.5       | 0.5      | 0.5         | 0.5         | 16      | 0.9       |

**Table C.11: Parameters of Test 5**

This leads to the following results, presented in table C.12:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.625</td>
<td>0.825</td>
<td>0.704</td>
<td>0.439</td>
<td>0.625</td>
<td>0.505</td>
<td>0.606</td>
<td>0.800</td>
<td>0.652</td>
</tr>
<tr>
<td>Dev</td>
<td>0.105</td>
<td>0.038</td>
<td>0.064</td>
<td>0.115</td>
<td>0.040</td>
<td>0.078</td>
<td>0.104</td>
<td>0.043</td>
<td>0.094</td>
</tr>
</tbody>
</table>

**Table C.12: Results of test 5**

The average results confirm the hypothesis that a more extended Dataset leads to a better topic model, which helps in computing a more accurate topical similarity and a more accurate summary. All metrics are improved.

### C.7 Test case 6 - Introduce also the basic corpus for training the topic model

Is a bigger corpus necessarily better for training a topic model? To answer this question, the basic corpus is used together with the two
previous ones to train our model. The parameters used are presented in table C.13:

<table>
<thead>
<tr>
<th>L_win</th>
<th>s</th>
<th>k_peak</th>
<th>T_t</th>
<th>i_T</th>
<th>k_T</th>
<th>k_ME</th>
<th>k_ME %</th>
<th>T_m %</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table C.13: Parameters of Test 6

This leads to the following results, presented in table C.14:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.627</td>
<td>0.821</td>
<td>0.704</td>
<td>0.441</td>
<td>0.626</td>
<td>0.507</td>
<td>0.609</td>
<td>0.796</td>
<td>0.653</td>
</tr>
<tr>
<td>Dev</td>
<td>0.107</td>
<td>0.040</td>
<td>0.066</td>
<td>0.112</td>
<td>0.041</td>
<td>0.082</td>
<td>0.107</td>
<td>0.043</td>
<td>0.095</td>
</tr>
</tbody>
</table>

Table C.14: Results of test 6

Experiments enhance small variations compared to the previous test case, and generally do not justify the increased computational time, affecting both training and inference phases.

### C.8 Test case 7 - suidf for lexical Similarity

The *suidf* frequency measure has been introduced in the functional segmentation to integrate *idf*, term frequency and speaker information, a specific feature of meetings. Here, the *suidf* frequency measure is also used to compute the utterance lexical similarity used for summarizing dialogues, i.e. sections where more speakers are involved. The parameters used are presented in table C.15:

<table>
<thead>
<tr>
<th>L_win</th>
<th>s</th>
<th>k_peak</th>
<th>T_t</th>
<th>i_T</th>
<th>k_T</th>
<th>k_ME</th>
<th>k_ME %</th>
<th>T_m %</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table C.15: Parameters of Test 7

This leads to the following results, presented in table C.16:
### Test case 8 - Merge topical and lexical similarity before the random walk

Until now, two different graphs have been used to propagate utterance importances using weights proportional to topical and lexical similarity, respectively. In this case, the edge weights are summed before the random walk, in order to evaluate which solution leads to better results based on ROUGE metric. The parameters used are presented in table C.17:

<table>
<thead>
<tr>
<th>$L_{win}$</th>
<th>$s$</th>
<th>$k_{peak}$</th>
<th>$T_t$</th>
<th>$i_T$</th>
<th>$k_T$</th>
<th>$k_{MF}$</th>
<th>$k_{ME}$</th>
<th>$T_m%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>$K_{NER}$</td>
<td>freq$</td>
<td>D_{LDA}$</td>
<td>$k_{top}$</td>
<td>$k_{lex}$</td>
<td>$L_{m%}$</td>
<td>$L_{d%}$</td>
<td>$N_T$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>N/A</td>
<td>suidf</td>
<td>AMI+BBC</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
</tbody>
</table>

This leads to the following results, presented in table C.18:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.626</td>
<td>0.825</td>
<td>0.705</td>
<td>0.441</td>
<td>0.626</td>
<td>0.507</td>
<td>0.608</td>
<td>0.801</td>
<td>0.653</td>
</tr>
<tr>
<td>Dev</td>
<td>0.106</td>
<td>0.037</td>
<td>0.065</td>
<td>0.116</td>
<td>0.038</td>
<td>0.078</td>
<td>0.105</td>
<td>0.042</td>
<td>0.094</td>
</tr>
</tbody>
</table>

The results are almost not affected by this change, meaning that the sum of the similarity measures leads to the same converged utterance importances obtained by summing the importance obtained with random walks through two different graphs. Computationally, this solution is more efficient.
C.10 Test case 9 - Use topic keywords for keyword extraction

Evaluates the effect of using topic keywords\(^1\) only for keyword extraction, while utterance importances are initialized uniformly. The parameters used are presented in table C.19:

<table>
<thead>
<tr>
<th>L(_{\text{win}})</th>
<th>s</th>
<th>(k_{\text{peak}})</th>
<th>(T_t)</th>
<th>(i_T)</th>
<th>(k_T)</th>
<th>(k_{MF})</th>
<th>(k_{ME})</th>
<th>(T_n) %</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(K_{NER})</th>
<th>freq(_{lex})</th>
<th>(D_{LDA})</th>
<th>(k_{top})</th>
<th>(k_{lex})</th>
<th>(L_m) %</th>
<th>(L_d) %</th>
<th>(N_T)</th>
<th>(\alpha)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N/A</td>
<td>suidf</td>
<td>AMI+BBC</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
<td></td>
</tr>
</tbody>
</table>

Table C.19: Parameters of Test 9

This leads to the following results, presented in table C.20:

<table>
<thead>
<tr>
<th></th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.627</td>
<td>0.823</td>
<td>0.705</td>
<td>0.442</td>
<td>0.625</td>
<td>0.507</td>
<td>0.609</td>
<td>0.800</td>
<td>0.654</td>
</tr>
<tr>
<td>Dev</td>
<td>0.105</td>
<td>0.040</td>
<td>0.064</td>
<td>0.114</td>
<td>0.041</td>
<td>0.077</td>
<td>0.104</td>
<td>0.045</td>
<td>0.093</td>
</tr>
</tbody>
</table>

Table C.20: Results of test 9

It does not show any improvement, when the model is trained on AMI Corpus and BBC news.

C.11 Test case 10 - Use topic keywords for keyword extraction and initial utterance importances

Let now introduce the concept of topic keywords, for keyword extraction and initial utterance importances. Importance are initially uniform, then in each sentence, the importance is multiplied by the topic keyword score, for each topic keyword in the sentence:

\[
score(s) = score(s) \cdot weight(w), \forall w \in K_w \cap s
\]

\(^1\)Topic keywords are defined as words with probability higher than a threshold to belong to the most likely topic of that meeting.
where $K_w$ is the set of topic keywords. The parameters used are presented in table C.21:

<table>
<thead>
<tr>
<th>$L_{win}$</th>
<th>$s$</th>
<th>$k_{peak}$</th>
<th>$T_t$</th>
<th>$i_T$</th>
<th>$k_T$</th>
<th>$k_{MF}$</th>
<th>$k_{ME}$</th>
<th>$T_m%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0.2</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>$K_{NER}$</td>
<td>freq_lex</td>
<td>$D_{LDA}$</td>
<td>$k_{top}$</td>
<td>$k_{lex}$</td>
<td>$L_m %$</td>
<td>$L_d %$</td>
<td>$N_T$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>N/A</td>
<td>suidf</td>
<td>AMI+BBC</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Table C.21: Parameters of Test 10

This leads to the following results, presented in table C.22:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.625</td>
<td>0.823</td>
<td>0.703</td>
<td>0.441</td>
<td>0.624</td>
<td>0.506</td>
<td>0.607</td>
<td>0.800</td>
<td>0.652</td>
</tr>
<tr>
<td>Dev</td>
<td>0.107</td>
<td>0.038</td>
<td>0.066</td>
<td>0.116</td>
<td>0.042</td>
<td>0.079</td>
<td>0.106</td>
<td>0.043</td>
<td>0.095</td>
</tr>
</tbody>
</table>

Table C.22: Results of test 10

Again, no significant effect is generated.

### C.12 Test case 11 - Accumulate topic keyword weights in each sentence

Another way of considering topic keywords for initializing utterance importances is to multiply the initial importance by the accumulated keywords weight for each sentence.

$$
    acc\_score(s) = \sum_{w} weight(w), \forall w \in K_w \cap s
$$

$$
    score(s) = score(s) \cdot acc\_score(s)
$$

(C.2)

The parameters used are presented in table C.23:

This leads to the following results, presented in table C.24:

This solution shows a small improvement in Recall and F-measure, achieving the best results together with Test case 7.
### C.13 Test case 12 - Accumulate topic keyword weights with extended model corpus

Accumulated weights depending on topic keywords to initialize utterance importances is also tested when the topic model is trained on AMI, BBC and Basic Corpora. The parameters used are presented in table C.25:

<table>
<thead>
<tr>
<th>$L_{win}$</th>
<th>s</th>
<th>$k_{peak}$</th>
<th>$T_i$</th>
<th>$i_T$</th>
<th>$k_T$</th>
<th>$k_{MF}$</th>
<th>$k_{ME}$</th>
<th>$T_m$%</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>$K_{NER}$</td>
<td>freq lex</td>
<td>$D_{LDA}$</td>
<td>$k_{top}$</td>
<td>$k_{lex}$</td>
<td>$L_m$%</td>
<td>$L_d$%</td>
<td>$N_T$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>N/A</td>
<td>suidf</td>
<td>ALL</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.626</td>
<td>0.827</td>
<td>0.705</td>
<td>0.441</td>
<td>0.627</td>
<td>0.507</td>
<td>0.608</td>
<td>0.802</td>
<td>0.653</td>
</tr>
<tr>
<td>Dev</td>
<td>0.106</td>
<td>0.038</td>
<td>0.065</td>
<td>0.115</td>
<td>0.039</td>
<td>0.078</td>
<td>0.105</td>
<td>0.043</td>
<td>0.095</td>
</tr>
</tbody>
</table>

This leads to the following results, presented in table C.26:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.627</td>
<td>0.820</td>
<td>0.704</td>
<td>0.442</td>
<td>0.626</td>
<td>0.507</td>
<td>0.609</td>
<td>0.796</td>
<td>0.653</td>
</tr>
<tr>
<td>Dev</td>
<td>0.107</td>
<td>0.038</td>
<td>0.067</td>
<td>0.120</td>
<td>0.040</td>
<td>0.083</td>
<td>0.108</td>
<td>0.040</td>
<td>0.096</td>
</tr>
</tbody>
</table>

The recall metric is lower than in Test case 11, confirming that overextending the topic model training set does not guarantee better performance.
C.14  Test case 13 - Use named entities for keyword extraction

The last tests aim to determine the impact of considering named entities to extract meeting keywords. Each word weight is multiplied by $K_{NER}$. The parameters used are presented in table C.27:

<table>
<thead>
<tr>
<th>$L_{win}$</th>
<th>$s$</th>
<th>$k_{peak}$</th>
<th>$T_t$</th>
<th>$i_T$</th>
<th>$k_T$</th>
<th>$k_{MF}$</th>
<th>$k_{ME}$</th>
<th>$T_m%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>$K_{NER}$</td>
<td>freqlex</td>
<td>D_LDA</td>
<td>$k_{top}$</td>
<td>$k_{lex}$</td>
<td>$L_m%$</td>
<td>$L_q%$</td>
<td>$N_T$</td>
<td>$\alpha$</td>
</tr>
<tr>
<td>3</td>
<td>suidf</td>
<td>AMI+BBC</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Table C.27: Parameters of Test 13

This leads to the following results, presented in table C.28:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>F-L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Avg</td>
<td>0.625</td>
<td>0.825</td>
<td>0.705</td>
<td>0.440</td>
<td>0.630</td>
<td>0.507</td>
<td>0.608</td>
<td>0.802</td>
</tr>
<tr>
<td>0.653</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dev</td>
<td>0.107</td>
<td>0.039</td>
<td>0.064</td>
<td>0.113</td>
<td>0.042</td>
<td>0.075</td>
<td>0.105</td>
<td>0.045</td>
</tr>
<tr>
<td>0.094</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table C.28: Results of test 13

This configuration outperforms all the other cases in terms of Recall on ROUGE-2, which considers bigram overlaps. The other results are similar to the best ones just obtained.

C.15  Test case 14 - Use named entities for keyword extraction and initial utterance importances

Like the topic keywords, named entities can also be used to initialize the utterance importances that will then be propagated in the multi-layer graph. The parameters used are presented in table C.29:

This leads to the following results, presented in table C.30:
This approach outperforms all the proposed approaches in terms of ROUGE-2 and ROUGE-L, even if it does not improve the results in terms of ROUGE-1. Therefore, these are the features that lead to the best result in terms of ROUGE metrics.

### C.16 Test case 15 - 30% compression rate

All the previous cases have been performed with a compression rate of 50% of the non stop words. Now, the optimal parameters determined in the previous test cases set are used to perform test at 30% of compression ratio. The parameters used are presented in table C.31:

<table>
<thead>
<tr>
<th>L_wi</th>
<th>s</th>
<th>k_peak</th>
<th>T_l</th>
<th>i_T</th>
<th>k_T</th>
<th>k_MF</th>
<th>k_ME</th>
<th>T_m%</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>K_{NER} freq_{lex}</td>
<td>D_{LDA}</td>
<td>k_top</td>
<td>k_{lex}</td>
<td>L_m%</td>
<td>L_d%</td>
<td>N_T</td>
<td>α</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>suidf</td>
<td>AMI+BBC</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>16</td>
<td>0.9</td>
</tr>
</tbody>
</table>

Table C.31: Parameters of Test 15

This leads to the following results, presented in table C.32:

<table>
<thead>
<tr>
<th>-</th>
<th>P-1</th>
<th>R-1</th>
<th>F-1</th>
<th>P-2</th>
<th>R-2</th>
<th>F-2</th>
<th>P-L</th>
<th>R-L</th>
<th>F-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>0.668</td>
<td>0.675</td>
<td>0.665</td>
<td>0.470</td>
<td>0.453</td>
<td>0.451</td>
<td>0.645</td>
<td>0.651</td>
<td>0.629</td>
</tr>
<tr>
<td>Dev</td>
<td>0.100</td>
<td>0.055</td>
<td>0.047</td>
<td>0.110</td>
<td>0.056</td>
<td>0.053</td>
<td>0.100</td>
<td>0.055</td>
<td>0.054</td>
</tr>
</tbody>
</table>

Table C.32: Results of test 15
As expected, the fraction of important sentences extracted compared to the total number of important sentences from the reference summary (Recall) decreases, while the fraction of relevant sentences compared to the number of irrelevant sentences (Precision) increases. The F-measure decreases of a few percent points, but it outperforms the current unsupervised extractive state of the art in terms of ROUGE-1 metric.
Appendix D

Summarization Example

A meeting transcript and its corresponding reference and extracted summaries are presented together, to get a better overview of the current performance. The example meeting is TS3005a. Pink sentences are correctly detected. Blue sentences are important sentences not detected by our method, but included in the reference summary.
Good morning.
busy job.
good morning.
Good morning.
I’d like to welcome you to our first meeting.
I’ve prepared a little presentation,
and I hope you will introduce yourself in a few minutes
I’m the Project Manager of this project.
Well I will tell you on what actually is the project.
This is the agenda for our first meeting.
then we will get I will hope we will get acquainted to each other.
we’ll do a little cool training with these two things.
we’ll take a look at the project plan.

Actually we have to discuss because we have to create a product.
And then we will close this session.
I’d like to introduce you to this room.
as you probably have noticed there are little black fields on the table.
you have to put your laptop exactly in that field so the little cameras can see your face.
everywhere around the room especially here for your face.
and this is n’t a pie
it is a set of microphones.
there are microphones here also.
But please do n’t be afraid of them.
They won’t hurt you.
well I said I’m the Project Manager.
and I’m hoping for a good project.
and I’d like to hear who you are.
what your functions are on this project.
let’s start with the ladies.

Figure D.1: TS3005s transcript in black, correctly extracted sentences in pink and important missing sentences in blue, part 1
my function is User Interface Design,
I'm the Industrial Designer
and I hope to look forward to a very pleasing end of this project.
I'm Marketing Expert.
My job is in the company to promote company or promote products to the customers.
So I also hope we have a pleasant working with each other.
well we have some expertise from different pieces of the of the company.
well I said I'm working on a project
the aim for the project is to to create a to design a new remote control which has to be original
user friendly.
And I hope we have the expertise to create such a project such a product.
the way we hope to achieve that is the following methods.
It consists of three phases.
namely the functional design

conceptual design
detailed design
all of these phases consists of two parts
namely individual work part
a meeting where we will discuss our work so far.
But first I will tell you something about the tools we have here.
I already talked about the cameras
but they are not of much use to us.
we will have to take advantage of these two things.
They are smart boards.
you can give a presentation on them.
And this one here is a white board.
I will instruct you about that soon.
as you also noticed this presentation document is in our project folder.
every document you put in this folder is it is possible to show that here in our meeting room.
there are available on both smart boards.
but I think we will mainly use this one for the documents in the shared folder.
this is the same tool bar as it located here.
the most functions we will use will be to add a new page
to go back
forward between pages
and of course to save it every now.
this is the pen with which you can draw on the board.

Show next segment
but I first have to put it on the pen
you see I'm now to it too.
then you can write things like text or whatever you want.
As you can see you have to move it a little bit slow.
it's not such a fast board
it's a smart board but also a slow board.
but you can write things.
of course you can also
erase things.
so we have got left.
and you can also delete an entire page.
Just simply create a new one.
start all over because we want to save all the results.
does everyone understand this?

nice application?
Well you can erase it with the eraser
but you shouldn’t delete an entire page
but just create a new blank one.
I will delete this one now because we do not use it yet.
But you can of course erase when you make a mistake
but do not delete entire pages.
And you can also let's see
change the colour of your pen.
for instance take a blue one.
change the line width like to five.
that's what you will need for our first exercise.
because I'm going to ask you to draw your favourite animal.

Figure D.2: TS3005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue, part 2

Figure D.3: TS3005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue, part 3
Show next segment
It's also to get to know each other because I'm asking three things
to do it on a blank sheet
with different colours.
and I just showed you how to pick a colour.
and also with different pen widths which I also showed you.
a favourite characteristic can be just one word.
Well I'm not very good at drawing
but I will go first
and try to draw
or maybe you should guess what I'm drawing.
Simon's dinosaur.
could be everything.
Maybe when I put on
It could be a turtle.
well the shell doesn't have legs.
and I hope our project group will not be slow.
but we will work to a good result.
time for another animal.
would you like to go next?
it was four months I right.
To make it a little bit easier.
with that cute.
recognise as a giraffe.
the favourite animal characteristic is that the long neck.
it can reach everything.
and I hope I can also reach a lot with this project.
so that's my favourite animal.
could you write the words.

Anything else you need to know?
Bunny rabbit.
Bunny rabbit.
wrong one.
well you can guess what it is.
and well it's quick.
little rabbits.
that's my favourite animal.
and our final drawing.
Bob Ross.
I've drawn a dolphin because of its intelligence.
one of the most intelligent animals in our world.
you can try out the eraser now.
well not perfect.
well thank you very much.
I can see we have some drawing talent in this group.
nice animals.
nice words.
sounds good.
back to business.
back to the money part.
from the finance department I have learned that we are aiming for a selling price of twenty five euros.
and we're hoping for a sum of fifty million euros.
we are hoping to achieve that by aiming for an international market.
and the production cost will be twelve euro fifty max.
well it's time for some discussion.

Figure D.4: TS3005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue, part 4

Figure D.5: TS3005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue, part 5
Show next segment

I've written down some examples here of what we can speak about.
what's your experience with remote controls
what kind of ideas do you have to design a new remote control
maybe for which market segments should we aim
or should we aim for all segments.
well actually I'd like to hand the word back to you.
what's your experience with remote controls?
A lot of buttons.
and you always lose them.
A lot of buttons which you do not use or who use do not use complex.
I always lose them.
not user friendly.
search for the buttons
it's not fun to use a remote.
well maybe we should try to make it fun.
the channel number you have to use.
perhaps you have a lot of road remotes
and remote controls.
perhaps you can integrate them or something.
you have different remote controls for different devices.
different remote controls for the use of different devices.
perhaps that's an idea.
you still have a lot of buttons
and which do you not use.
but you could I thin
put those buttons behind some kind of protection so that if you only get to see them when you need 'em.
but it'll get very big then the remote control.
just for example you got the same size remote control you use everyday
but the usual buttons such as zapping as you call it in English.
the volume control is only the only possible buttons to use directly.
but not the buttons used to search on the channel on your television.
you only use those the first time.
so maybe a minimalist design
the least possible amount of buttons.
but you should make sure that you have every button they need on it.
because things for televisions.
so you do not want to bother people with loads of buttons.
but on the other hand they need many buttons so they do not have to get out of their seat.
because I think a market will be all kind of people.
elDERLY people.
young people.

Figure D.6: TS3005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue, part 6

Show next segment

but if it's if it's international you should look in Britain他们 have different things they can do with the TV
so that you can choose what you want to see.
the idea if you should take that in consideration.
or that you just should aim for the normal TV's that
I think that's better one.
I think if you're going to target a lot of people
in the UK.
only Britain then I think that the cost will rise higher than the twenty fifty.
I don't know if they have that anywhere else.
I think the size is better to use the whole world.
when I think of it I think the main idea of this remote remote control is to
make it user friendly.
so I think when we say that the customers will buy this remote control.
they already have the remote control which comes with the the standard remote control which comes with the television.
so it only has to have the most used buttons.
you do not have to integrate the buttons to search the channels on your television.
standard already.

but but it is impossible to to accommodate accommodate all the buttons on the on the difference different televisions sets on one remote control.
because for example one television has the opportunity in a to make it possible for to see on one side of the screen television
and on the other side just a regular television.
I think s a a a standard nowadays do this.
but they do not use the same signal.
on remote control.
because you can't see a Panasonic remote control on a on a Philips television.
well not everywhere.
so I think generally.
but then you have to choose the this always with r universal remotes you have to choose the code.
you can use which which type of television you have.
but I think like the two pages on the same screen.
not a normal television.
that's that's nowadays standard.
you can choose the code.

Figure D.7: TS3005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue, part 7
Now next request

I think that most people '\text{1}' will buy the remote control because the first they lost the one they lost first one or the first one is broken

so perhaps they have a get a an older television

so that option is not optimal for those people.

but the people have a new television

and c, if you look into the future

then they want will want the buttons

of their thing is broke.

Now next request

will we have time

yet? I mean I have to tell you.

I don't think there is much left

so we're starting to close.

Our next meeting will start well

or a little bit early

but our next meeting will start in thirty minutes.

In the meantime there's time for some individual actions.

Now next request

well that's good.

five minutes

right on schedule.

next meeting person will take a look at the new requirement specification.

The user interface designer will work out the technical function design.

And this was the interface designer?

or the interaction designer.

Interface Designer

First guess was right.

all take a look at the the working design.

The interaction designer will take a look at the working design

and the in usability interaction.

let's just use the acronym.

Of course specific instructions will be sent to you through your personal coach.

will these instructions will be in the email you will receive shortly.

and of course you have your own expertise.

will that was what I had to say.

Figure D.8: T53005a transcript in black, correctly extracted sentences in pink and important missing sentences in blue, part 8